SPREAD SPECTRUM CODE SYNCHRONIZATION USING SIGNAL TRANSITIONS

A Thesis

Submitted to the Faculty of Graduate Studies and Research

in Partial Fulfilment of the Requirements

for the Degree of

Master of Science

in the Department of Electrical Engineering

University of Saskatchewan

by

Floyd Larb Fore Lau

Saskatoon, Saskatchewan

AUGUST, 1991

Copyright (c) 1991 F. L. F. Lau

Copyright

The author has agreed that the Library, University of Saskatchewan, shall make this thesis freely available for inspection. Moreover, the author has agreed that permission for extensive copying of this thesis for scholarly purposes may be granted by the professor or professors who supervised the thesis work recorded herein or, in their absence, by the Head of the Department or Dean of the College in which this thesis work was done. It is understood that due recognition will be given to the author of this thesis and to the University of Saskatchewan in any use of the material in this thesis. Copying or publication or any other use of the thesis for financial gain without approval by the University of Saskatchewan and the author's written permission is prohibited.

Requests for permission to copy or to make other use of the material in this thesis in whole or in part should be addressed to:

Head of the Department of Electrical Engineering,
University of Saskatchewan,
SASKATOON, Canada.

Acknowledgements

The author wishes to thank Professor D. E. Dodds for his supervision, guidance and encouragement during the course of this work, and for suggesting the idea. The assistance of Professor A. Krause during the development of the computerized test setup is gratefully acknowledged. The author also thanks Dr. S. M. Pan for his valuable comments on spread spectrum acquisition.

Financial assistance was obtained through Natural Sciences and Engineering Research Council (NSERC) operating grant #OGP0007616 and this is gratefully acknowledged.

UNIVERSITY OF SASKATCHEWAN

Electrical Engineering Abstract 91A349

SPREAD SPECTRUM CODE SYNCHRONIZATION USING SIGNAL TRANSITIONS

Student: F. L. F. Lau

Supervisor: D. E. Dodds

M. Sc. Thesis Presented to the College of Graduate Studies

August 1991

Abstract

Direct Sequence (DS) Spread Spectrum (SS) techniques are finding more frequent use in today's commercial communication systems. Serial search is a widely used scheme for DS SS code phase acquisition. The problem of serial search is that it requires long acquisition time which results in a long set-up time before the communication link is useable.

In this thesis, a new acquisition scheme called Rapid Acquisition by Accumulation of Transition Sequence (RAATS) is introduced. The RAATS system detects and accumulates the transitions of the transmitted pseudonoise (PN) sequence. A code phase estimate is made by matching a periodic pattern in the accumulated transition sequence. The RAATS method is nearly insensitive to polarity reversals of the data and allows fast acquisition in presence of high noise.

Prototypes of both the serial search and the RAATS acquisition scheme were successfully implemented and measured for their acquisition time performance. Source data at 2400 bits/s was spread by a 1023 chips/cycle maximal length PN sequence. The chip rate was set at 32 times the data rate. Test results indicate that the RAATS system has a negative exponential acquisition time probability density function. Significant improvement in acquisition time is achieved by the RAATS system when compared with a serial search acquisition system.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AWGN	Additive White Gaussian Noise
BASIC	Beginner's All-purpose Symbolic Instruction Code
BPSK	Binary Phase Shift Keying
BW	Bandwidth
CCD	Charge Coupled Devices
DS	Direct Sequence
FET	Field Effect Transistor
FH	Frequency Hopping
LPF	Low Pass Filter
pdf	Probability Density Function
PN	Pseudonoise
PSD	Power Spectral Density
RAATS	Rapid Acquisition by Accumulation of Transition Sequence
RASE	Rapid Acquisition by Sequential Estimation
rms	Root-Mean-Square
SAW	Surface Acoustic Wave
SNR	Signal-to-Noise Ratio
SS	Spread Spectrum
SVL	Search/Verification/Lock
TH	Time Hopping
TTL	Transistor Transistor Logic
XOR	Exclusive OR

Chapter 1

INTRODUCTION

1.1. Spread Spectrum Overview

Almost all communication system modulation and coding techniques are designed to efficiently utilize the bandwidth limited signal energy and to achieve a low probability of error. In contrast, the Spread Spectrum (SS) technique spreads the transmitted signal power over a frequency band much wider than the minimum bandwidth required to transmit the original information. At the receiver, the SS signal is compressed back to its original narrow band while all unwanted signals are spread over the wide transmission bandwidth.

The initial application of the SS technique was in military guidance and communication systems. Due to the secrecy under which these systems were developed, the history of SS communications is not well documented. The concept of SS was first documented in the late 1940s and early 1950s by Mortimer Rogoff in terms of a noise wheels experiment [1, 2]. The experiments successfully demonstrated the possibility of conveying information hidden in noiselike signals. This idea was then applied into the development of SS communication systems around 1955. During the period of 1955 to 1960, experimental systems such as the HUSH-UP and BLADES and non-experimental systems such as the F9C-A and RAKE were completed and deployed [2].

For a communication system to be considered a SS system, it must fulfil the following three criteria [1]:

(1) The transmitted signal occupies a bandwidth much in excess of the minimum bandwidth necessary to send the information.

(2) Spreading is accomplished by means of a spreading signal, often called a code signal, which is independent of the data.

(3) At the receiver, despreading (recovering the original data) is accomplished by a synchronized replica of the code signal used to spread the information.

SS offers a number of advantages over other modulation/demodulation techniques. First, it provides a greater immunity from jamming, interference, and multipath distortion. Furthermore, the spreading is typically designed such that the SS signal is transparent to other users. Hence the probability of being detected (or intercepted) is extremely low. In addition, multiple SS signals can coexist in the same frequency band simultaneously. This is known as Code Division Multiple Access (CDMA). All these advantages make SS an interesting choice for communication in today's congested frequency spectrum.

There are three types of SS systems, Direct Sequence (DS), Frequency Hopping (FH), and Time Hopping (TH). There are also hybrid combinations of these three. Fig. 1.1 shows a basic block diagram of a DS SS system. Here the data is directly modulated (spread) by a high frequency pseudonoise (PN) or pseudorandom sequence before carrier modulated for transmission. After the spreading, multiples of the original data spectrum are created

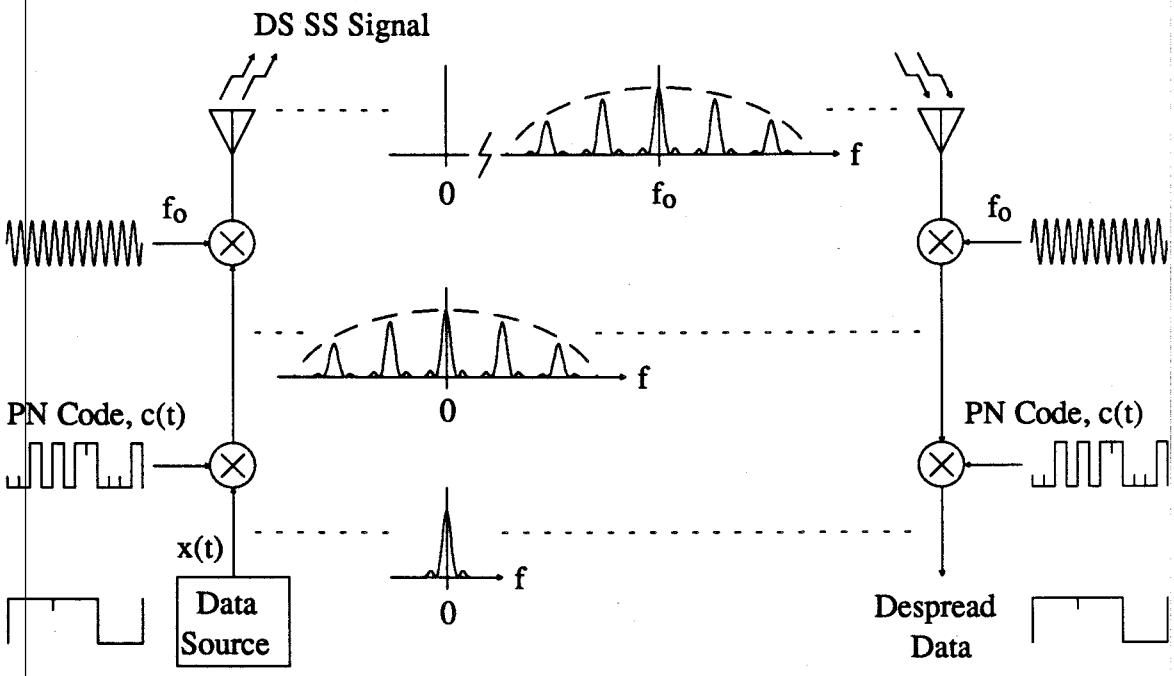


Fig. 1.1 Basic Block Diagram of a DS SS System

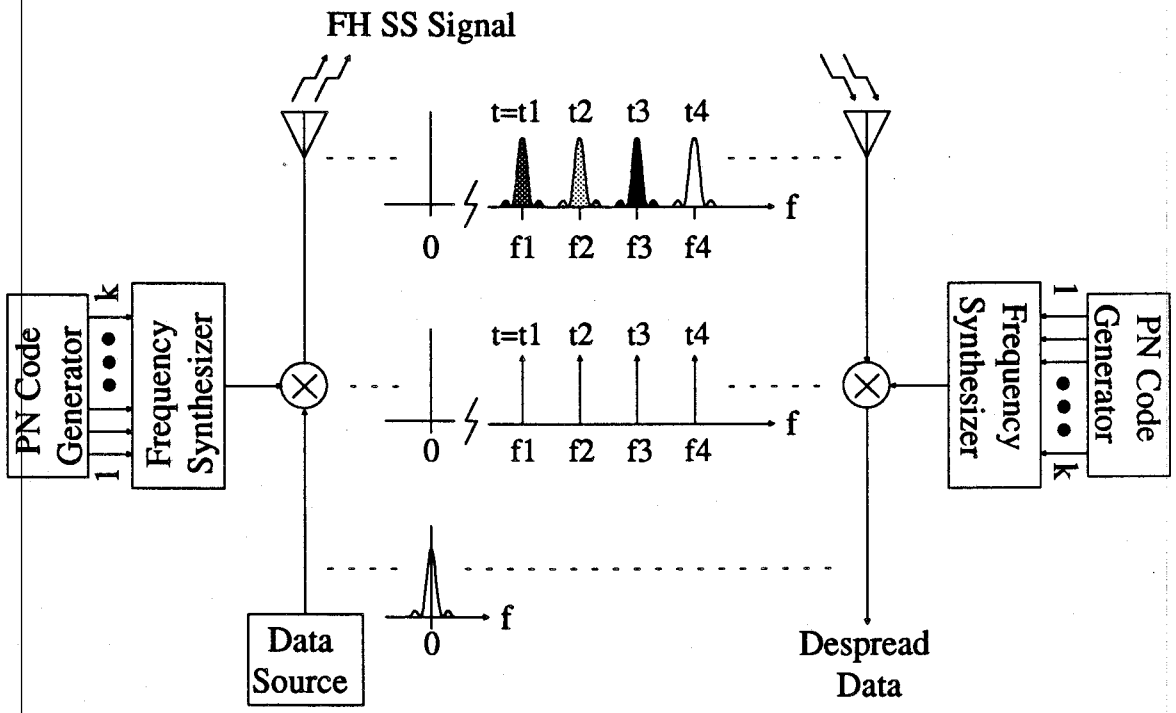


Fig. 1.2 Basic Block Diagram of a FH SS System

which occupy the entire wide band of the PN code sequence. The receiver which knows the spreading PN code sequence used by the transmitter can acquire the phase of the code sequence and demodulate (despread) the SS signal accordingly. Fig. 1.2 shows a general block diagram of a FH SS system. Here the carrier frequency is chosen from a set of 2^k frequencies. The PN sequence in this case is used to control the sequence or hop of the carrier frequencies. For example, at time t=t1, carrier frequency f1 is used, at time t=t2, carrier frequency f2 is used, and so on. Thus the data spectrum occupies an overall much wider bandwidth than conventional modulation scheme. In a TH SS system, transmission time is divided into frames of 2^k time slots as illustrated in Fig. 1.3. Source data is burst into a selected time slot within each frame. A PN sequence is used to assign the transmission time slots. No bandwidth expansion is achieved by the TH SS technique, but there is a time delay required to despread the signal. Compared with DS and FH SS which spread the signal in the frequency domain, TH SS spreads the signal in the time domain. Finally, hybrid SS

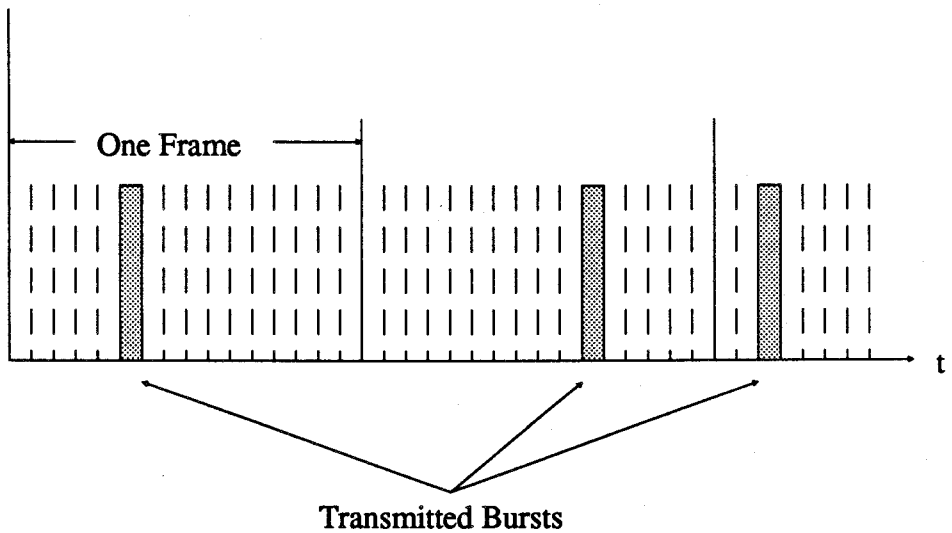


Fig. 1.3 Transmission Slots of a TH SS System

systems employ any possible combinations of the other three SS methods, such as DS/FH, FH/TH, or DS/FH/TH.

Whichever SS technique is used, the intended receiver must employ a synchronized replica of the PN code sequence in order to despread the received signal successfully. The process of synchronizing the locally generated PN code sequence with the received SS signal is usually accomplished in two steps. The first step, called acquisition, consists of bringing the two PN code sequences into coarse alignment with one another. Once the received SS signal has been acquired, the second step, called tracking, takes over and continually maintains the best possible code phase fine alignment by means of a tracking loop.

A PN code sequence is generated using a clocked shift register with specific feedback connections. The output sequence will repeat with a period (known as an epoch) which is 2^n-1 clock periods where n is the shift register length. The output symbol during each clock period is known as a chip. Thus the PN code sequence has 2^n-1 chips in each epoch.

The choice of SS techniques depends on the system specifications. The advantages and disadvantages [3] of DS, FH, and TH SS are summarized in Table 1.1. Note that hybrid SS is a special case in which some of the advantages of DS, FH, and TH SS systems are combined into a single system. One distinct feature of DS SS is its application to ranging systems. The phase of the synchronized code is proportional to the propagation delay of the PN sequence, hence the range from the transmitter to the receiver can be calculated.

Table 1.1 Advantages and Disadvantages of SS [3]

SS System	Advantages	Disadvantages
DS	Best noise and antijam performance. Most difficult to detect. Best discrimination against multipath.	Long acquisition time. Affected by near-far problem. Requires wideband channel with little phase distortion.
FH	Greatest amount of spreading. Can be programmed to avoid portions of the spectrum. Relatively short acquisition time. Less affected by near-far problem.	Complex frequency synthesizer. Not useful for range and range-rate measurement.
TH	High bandwidth efficiency. Implementation simpler than FH. Useful when transmitter is average power limited but not peak power limited.	Long acquisition time.

A number of SS systems of different sizes are in operation today. One important application of the DS SS system is the Navstar Global Positioning System (GPS) [4, 5, 6]. The GPS enables military as well as commercial users to determine their position on or above the earth's surface by measuring their distance to four GPS satellites which have accurately known position. The first Navstar satellite was launched in 1978. The scheduled completion date, with a total of 18 satellites operating in place, is set for 1993. Two PN code sequences are used to modulate a 50 bits/s navigation message. The first one is called the Clear/Acquisition (C/A) code. This code is a Gold code [4] of length 1023 chips. The code rate is 1.023 megachips/s. A different Gold code is used for each satellite so that a ground receiver can distinguish satellites. The second code sequence is called the Precise

(P) code. This code has a rate of 10.230 megachips/s and a period of 280 days. Each satellite is assigned a different phase of this long code. The satellites continuously broadcast the SS signals at two carrier frequencies, designated L1 (1575.42 MHz) and L2 (1227.60 MHz). A user set locks onto the codes by internally generating these codes and then aligning them with the received code.

Another well known DS SS system is the Tracking and Data Relay Satellite System (TDRSS) [4, 7, 8]. This system provides a two-way S-band and Ku-band communication links between the NASA Space Shuttle and the ground stations during the on-orbit period of the Shuttle flights. The S-band forward link (TDRS to Space Shuttle) operates at one of two carrier frequencies, 2106.406 or 2287.5 MHz. The PN code length and code rate is 2047 chips and 11.232 megachips/s, respectively. The data rate is either 32 kbits/s or 72 kbits/s and is convolutionally encoded with a rate 1/3, constraint length 7 coder. The Ku-band forward link, on the other hand, operates at 13.775 GHz. The PN code is a 1023 chips Gold code and the code rate is 3.028 megachips/s. The possible data rates are 32, 72, 96, or 216 kbits/s. The S-band despreader acquires PN code phase by using serial search with 1/2 code steps and multiple-dwell search/lock strategy while the Ku-band despreader acquires PN code phase by continuously slipping the local and the incoming PN codes past each other until a synchronism is achieved. Discussion on serial search and multiple-dwell search/lock strategy will be included in Chapter 2.

In 1985, the U. S. Federal Communication Commission (FCC) issued a ruling called Part 15.247 which allows for the unlicensed use of spread spectrum radios with up to one watt of output power in three shared frequency bands: 902 to 928 MHz, 2.400 to 2.4835

GHz, and 5.725 to 5.850 GHz [9]. This ruling has stimulated a wide range of commercial applications for spread spectrum. Personal Communications Network (PCN) [10, 11, 12], which is a network of pocket-sized radio telephones served by clusters of receiver-transmitter cells similar to current mobile cellular telephone systems, is one of such commercial products. Others like Local Area Wireless Network (LAWN), CDMA mobile communications network [13] are also blooming quickly.

1.2. Objective of Thesis

In order to demodulate SS signals, both the transmitter and the receiver PN code sequences must be synchronized. For all SS system designs, synchronization stands out as the most difficult task.

The investigation described in this thesis is focused on DS SS code phase acquisition. A few types of PN sequence acquisition scheme have been developed successfully in the past few decades. The common goal of all these schemes is to locate the correct PN code phase within a reasonably short period of time at low signal-to-noise ratio (SNR) and in many cases in the presence of jamming signals. Depending on the system specifications, different acquisition schemes could vary the degree of system complexity.

A widely used DS SS acquisition scheme is called serial search. This method performs a serial "trial and error" search through all potential PN code phases (or sometimes referred to as the code phase uncertainty region) until synchronization is achieved. A Search/Verification/Lock (SVL) strategy [14, 15, 16, 17] is often used as a logical procedure by which the process of code synchronization is controlled. With this SVL strategy, code

synchronization becomes a two step process with a search mode and a lock mode. In search mode, coarse synchronization is performed with the traditional serial search where the receiver code is adjusted in 1/2 code steps. In lock mode, fine adjustment is performed by using a code tracking loop. The transition between search mode and lock mode is controlled by the SVL strategy. A theoretical analysis of the factors affecting serial search performance with SVL strategy is given in [16].

Serial search is by far the simplest and most economic method to achieve PN code phase acquisition. However, the major drawback of serial search is its long acquisition time which results in long set up time before the communication link is usable. As a result, a faster acquisition scheme is always in demand.

A new approach of PN code phase acquisition, named Rapid Acquisition by Accumulation of Transition Sequence (RAATS), is proposed in this thesis. This technique linearly accumulates the transition information of the received SS signal. The accumulated transition signal is then tested for code phase acquisition. This technique allows acquisition to be achieved in much shorter time than the traditional serial search, thus allowing DS SS to be used in products which require fast channel establishment, such as cordless telephones, wireless modems, and cellular radio. RAATS will replace serial search in the process of code acquisition while the SVL strategy remains as a search/lock mode control.

The three main objectives of this thesis are:

(1) Experimentally verify the performance of the serial search acquisition technique with SVL strategy as analyzed by Dodds, Kumar and Pan [16].

(2) Experimentally examine the performance of the RAATS with SVL strategy.

(3) Compare the acquisition time performance of serial search and RAATS.

The following chapters will discuss the concepts, design, and test results of two DS SS acquisition systems which have been implemented.

Chapter 2

DIRECT SEQUENCE SPREAD SPECTRUM SYSTEMS

Since the acquisition methods being investigated in this thesis are for the synchronization of DS SS, a discussion about DS SS modulation and demodulation is necessary. Section 2.1 describes at the modulation process with emphasis on the PN code sequence while section 2.2 discusses the DS SS demodulation. Section 2.3 presents the existing acquisition techniques including serial search, parallel search, Rapid Acquisition by Sequential Estimation (RASE), and variable dwell time search.

Due to the nature of this project, detailed mathematical representations of DS SS modulation and demodulation are kept to minimum. A thorough theoretical analysis of SS systems can be found in [2, 4, 18, 19].

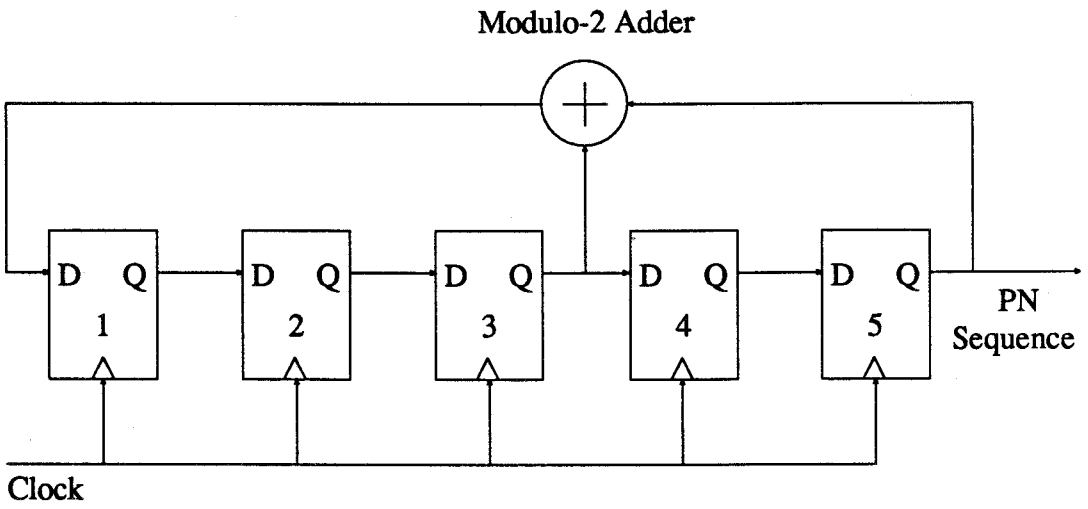
2.1. Direct Sequence Spread Spectrum Modulation

2.1.1. PN Sequence Generation

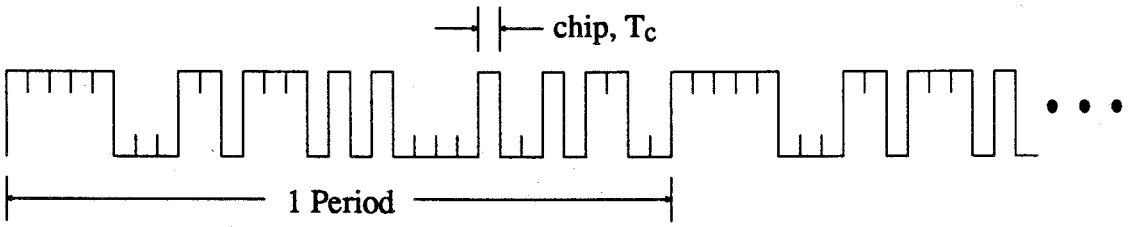
The key part of any spread spectrum system is the PN sequence generator which generates a binary pseudorandom sequence to spread the information. A PN sequence can be generated by using a feedback structure containing a modulo-2 adder in a n-stage shift register. The initial state of the shift register can have any value except the all zero state.

With appropriate feedback taps from the shift register, a maximal length sequence of length 2^n-1 can be generated.

Fig. 2.1 (a) is an example of a PN sequence generator which uses a 5-bit shift register. The generator shown is called a (3, 5) generator which means feedback taps are taken after the third and the fifth stages. The generator is driven by a stream of clock pulses which determine the rate of the PN code sequence. Each code symbol generated is given the name



(a) Pseudorandom Sequence Generator (n=5)



(b) Maximal Length Pseudorandom Sequence

Fig. 2.1 Maximal Length PN Sequence Generator

chip. Hence the chip interval or chip time, T_c, is equivalent to the clock period. Each time the generator is clocked, the content in each stage of the register shifts one stage to the right. Furthermore, the contents of stages 3 and 5 are modulo-2 added, and the result is fed back to the input of stage 1. The PN sequence, in this example, is taken from the output of the last stage and is shown in Fig. 2.1 (b). The sequence has a random appearance, but repeats after a period of 31 chips, and is thus a maximal length (2^5-1) sequence. For each sequence of maximal length, there is a corresponding complement of that sequence, the reverse of that sequence, and the complement of the reverse sequence. When the length of a PN sequence is less than 2^n-1, the sequence is classified as a non-maximal length sequence.

A reverse sequence can be generated by reversing the direction of the shift register and altering the connections of each modulo-2 adder. For the (3, 5) generator shown, a reverse sequence generator will have feedback taps taken after 2 and 5 delays. In other words, a (2, 5) generator will gives a reverse sequence of a (3, 5) generator. A set of maximal length PN sequence generators is tabulated in Table 2.1 [20]. A complete set of tap connections are listed up to shift register length 8. Note that reverse sequence generators are not listed in the table.

2.1.2. PN Sequence Properties

There are five basic randomness properties that can be found in any maximal length PN sequences [4]. Properties 1, 4 and 5, in particular, are very useful in their application to SS systems.

Table 2.1 Maximal Length PN Generators [20]

n	Generator
2	(1, 2)
3	(1, 3)
4	(1, 4)
5	(2, 5) (2, 3, 4, 5) (1, 2, 4, 5)
6	(1, 6) (1, 2, 5, 6) (2, 3, 5, 6)
7	(3, 7) (1, 2, 3, 7) (1, 2, 4, 5, 6, 7) (2, 3, 4, 7) (1, 2, 3, 4, 5, 7) (2, 4, 6, 7) (1, 3, 6, 7) (2, 5, 6, 7)
8	(2, 3, 4, 8) (3, 5, 6, 8) (1, 2, 5, 6, 7, 8) (1, 3, 5, 8) (2, 5, 6, 8) (1, 5, 6, 8) (1, 2, 3, 4, 6, 8) (1, 6, 7, 8)
9	(4, 9) (3, 4, 6, 9) (4, 5, 8, 9) (1, 4, 8, 9) (2, 3, 5, 9) (1, 2, 4, 5, 6, 9) (5, 6, 8, 9) (2, 7, 8, 9) (1, 3, 4, 6, 7, 9)
10	(3, 10) (2, 3, 8, 10) (3, 4, 5, 6, 7, 8, 9, 10) (1, 2, 3, 4, 5, 10) (2, 3, 6, 8, 9, 10) (1, 3, 4, 5, 6, 7, 8, 10)
11	(2, 11) (2, 5, 8, 10) (2, 3, 7, 11) (2, 3, 5, 11) (2, 3, 10, 11) (1, 3, 8, 9, 10, 11)
12	(1, 4, 6, 12) (1, 2, 5, 7, 8, 9, 11, 12) (1, 3, 4, 6, 8, 10, 11, 12) (1, 2, 5, 10, 11, 12) (2, 3, 9, 12) (1, 2, 4, 6, 11, 12)
13	(1, 3, 4, 13) (4, 5, 7, 9, 10, 13) (1, 4, 7, 8, 11, 13) (1, 2, 3, 6, 8, 9, 10, 13) (5, 6, 7, 8, 12, 13) (1, 5, 7, 8, 9, 13)
14	(1, 6, 10, 14) (1, 3, 4, 6, 7, 9, 10, 14) (4, 5, 6, 7, 8, 9, 12, 14) (1, 6, 8, 14) (5, 6, 9, 10, 11, 12, 13, 14) (1, 2, 3, 4, 5, 7, 8, 10, 13, 14)
15	(1, 15) (1, 5, 10, 15) (1, 3, 12, 15) (1, 2, 4, 5, 10, 15) (1, 2, 6, 7, 11, 15) (1, 2, 3, 6, 7, 15)
16	(1, 3, 12, 16) (1, 3, 6, 7, 11, 12, 13, 16) (1, 2, 4, 6, 8, 9, 10, 11, 15, 16) (1, 2, 3, 5, 6, 7, 10, 15, 16) (2, 3, 4, 6, 7, 8, 9, 16) (7, 10, 12, 13, 14, 16)
17	(3, 17) (1, 2, 3, 17) (3, 4, 8, 17)
18	(7, 18) (5, 7, 10, 18) (7, 8, 9, 10, 15, 16, 17, 18)
19	(1, 2, 5, 19) (3, 4, 5, 8, 13, 19) (3, 7, 9, 10, 12, 19)
20	(3, 20) (3, 5, 9, 20) (2, 3, 6, 8, 11, 20)
21	(2, 21) (2, 7, 14, 21) (2, 5, 13, 21)
22	(1, 22) (1, 5, 9, 22) (1, 4, 7, 10, 13, 16, 19, 22)
23	(5, 23) (5, 11, 17, 23)
24	(1, 2, 7, 24) (4, 5, 7, 8, 9, 11, 14, 16, 18, 20, 22, 24) (1, 4, 5, 9, 10, 13, 14, 15, 16, 17, 18, 19, 21, 24)
25	(3, 25) (1, 2, 3, 25) (3, 4, 12, 25)

Property 1 - Balance property. A maximal length sequence contains one more binary digit one than zero. The number of ones in the sequence is $\frac{1}{2}(N+1)$ where N is the length of the sequence.

Property 2 - Shift-and-add property. 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.

Property 3 - Sliding window property. Consider the PN sequence generator shown in Fig. 2.1, the PN sequence passes through all n=5 stages of the shift register generator. If a window of width n is slid along the sequence, each n-tuple is simply a state of the shift register. Since the shift register passes through all non-zero states exactly once, each n-tuple will appear exactly once.

Property 4 - Run property. A run is defined as a subsequence of identical symbols within the sequence. The length of this subsequence is the length of the run. Then, for any maximal length sequence, there is 1 run of ones of length n, 1 run of zeros of length n-1, 1 run of ones and 1 run of zeros of length n-2, 2 runs of ones and 2 runs of zeros of length n-3, 4 runs of ones and 4 runs of zeros of length n-4, ..., and 2n-3 runs of ones and 2n-3 runs of zeros of length 1.

Property 5 - Correlation property. The discrete period autocorrelation function of a maximal length sequence, R(k), has only two values and is given by

$$R(k) = \frac{1}{N} \sum_{n=0}^{N-1} C_n C_{n+k} \tag{2.1}$$

where $C_n = \pm 1$ is the chip value. Hence

$$R(k) = \begin{cases} 1 & , k = iN,\ i = ..., -2, -1, 0, 1, 2, \ldots \\ -\frac{1}{N} & , k \neq iN,\ i = ..., -2, -1, 0, 1, 2, \ldots \end{cases} \tag{2.2}$$

where i is any integer. The continuous autocorrelation function of the waveform, $R(\tau)$, is given by

$$R(\tau) = \left(1 - \frac{\tau}{T_c}\right) R(k) + \frac{\tau}{T_c} R(k+1) \tag{2.3}$$

in [4]. For $0 \leq \tau \leq T_c$ and k=iN for any integer i, $R(\tau)$ becomes

$$R(\tau) = \left(1 - \frac{\tau}{T_c}\right) + \left(\frac{\tau}{T_c}\right)\left(-\frac{1}{N}\right)$$

$$= 1 - \frac{\tau}{T_c}\left(1 + \frac{1}{N}\right) \tag{2.4}$$

For $T_c \leq \tau \leq (N-1)T_c$, $k \neq iN$ and $(k+1) \neq iN$ for any integer i, $R(\tau)$ becomes

$$R(\tau) = \left(1 - \frac{\tau}{T_c}\right)\left(-\frac{1}{N}\right) + \left(\frac{\tau}{T_c}\right)\left(-\frac{1}{N}\right)$$

$$= -\frac{1}{N} \tag{2.5}$$

Since R(k) is symmetrical about k=0, $R(\tau)$ is symmetrical about τ=0, i.e.,

$$R(\tau) = R(-\tau) \tag{2.6}$$

According to Equations (2.4), (2.5) and (2.6), the shape of $R(\tau)$ is shown in Fig. 2.2 (a). Since R(k) is periodic, $R(\tau)$ is also periodic and has a period of NT_c.

The power spectral density (PSD) of a PN sequence is found by taking the Fourier transform of its autocorrelation function. Since $R(\tau)$ is periodic, the power spectral density is a line spectrum.

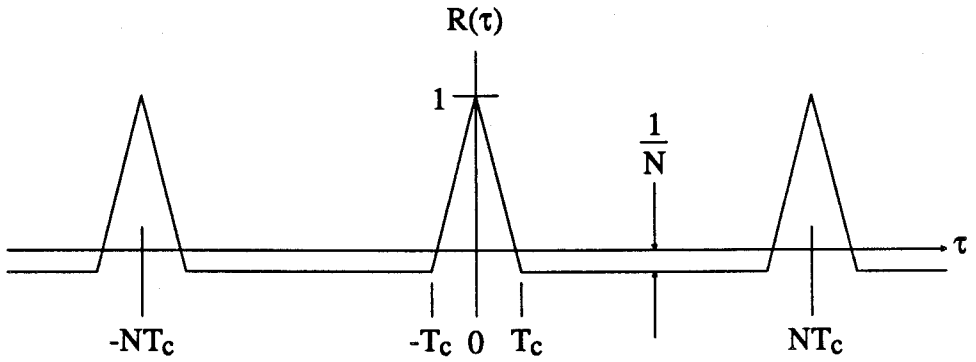
$$S(f) = \mathcal{F}[\, R(\tau)\,] \tag{2.7}$$

$$= \sum_{m=-\infty}^{\infty} r_m \, \delta(f - mf_o) \tag{2.8}$$

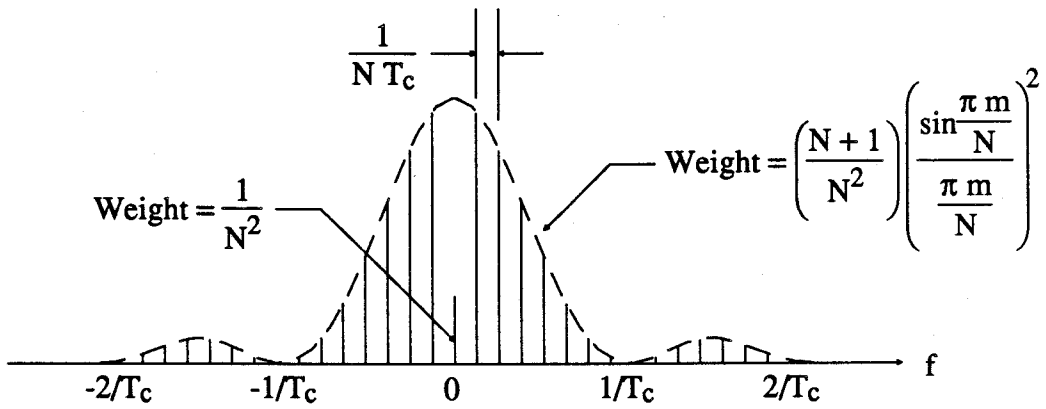
where $f_o = 1/(NT_c)$ and $\{r_m\}$ is a set of Fourier series coefficients evaluated to be

$$r_m = \begin{cases} \dfrac{1}{N^2} & ,m = 0 \\ \left(\dfrac{N+1}{N^2}\right)\left(\dfrac{\sin\frac{\pi m}{N}}{\frac{\pi m}{N}}\right)^2 & ,m \neq 0 \end{cases} \tag{2.9}$$

This power spectral density is illustrated in Fig. 2.2 (b).



(a) Autocorrelation Function of a PN Sequence



(b) Power Spectral Density of a PN Binary Sequence.

Fig. 2.2 Autocorrelation and PSD of PN Sequence

2.1.3. Properties of Spread Signal

A basic model of a DS SS system was shown earlier in Fig. 1.1. At the modulator, the data, x(t), of rate f_b bits/s is directly modulated by a PN sequence, c(t), of chip rate f_c chips/s. Both signals are only ±1 in value. Since multiplication in the time domain transforms to convolution in the frequency domain,

$$x(t)\, c(t) \longleftrightarrow X(f) * C(f) \tag{2.10}$$

and the power spectral density of random data x(t) is

$$S_x(f) = T_b \left(\frac{\sin \pi f T_b}{\pi f T_b} \right)^2 \tag{2.11}$$

where $T_b = 1/f_b$. Hence after the modulation each discrete spectral line of the PN sequence shown in Fig. 2.2 (b) becomes a sinc function in the case when f_b is less than $\frac{1}{2N T_c}$. However, for typical commercial mobile radio systems where T_b equal one PN epoch, i.e., $f_b = \frac{1}{N T_c}$, the resulting spectrum will merge into a continuous smooth spectrum approximately the same as the envelope of the PN sequence PSD. This spreading process is illustrated in Fig. 2.3 with the time waveform shown on the left and the relative PSD shown on the right.

As indicated in Fig. 2.3, the power of the data is spread to the entire bandwidth of the PN sequence. In many cases, the SS signal power is spread to a level below the noise level of the transmission channel, making it difficult to detect by any eavesdroppers.

2.2. Direct Sequence Spread Spectrum Demodulation

DS SS demodulation is accomplished by multiplying the received signal with a synchronized replica of the same PN sequence used to spread the original information. Consider that the received SS signal is coupled with a single-tone jammer and that Additive White Gaussian Noise (AWGN) is presented as illustrated in Fig. 2.4 (a). With the receiver

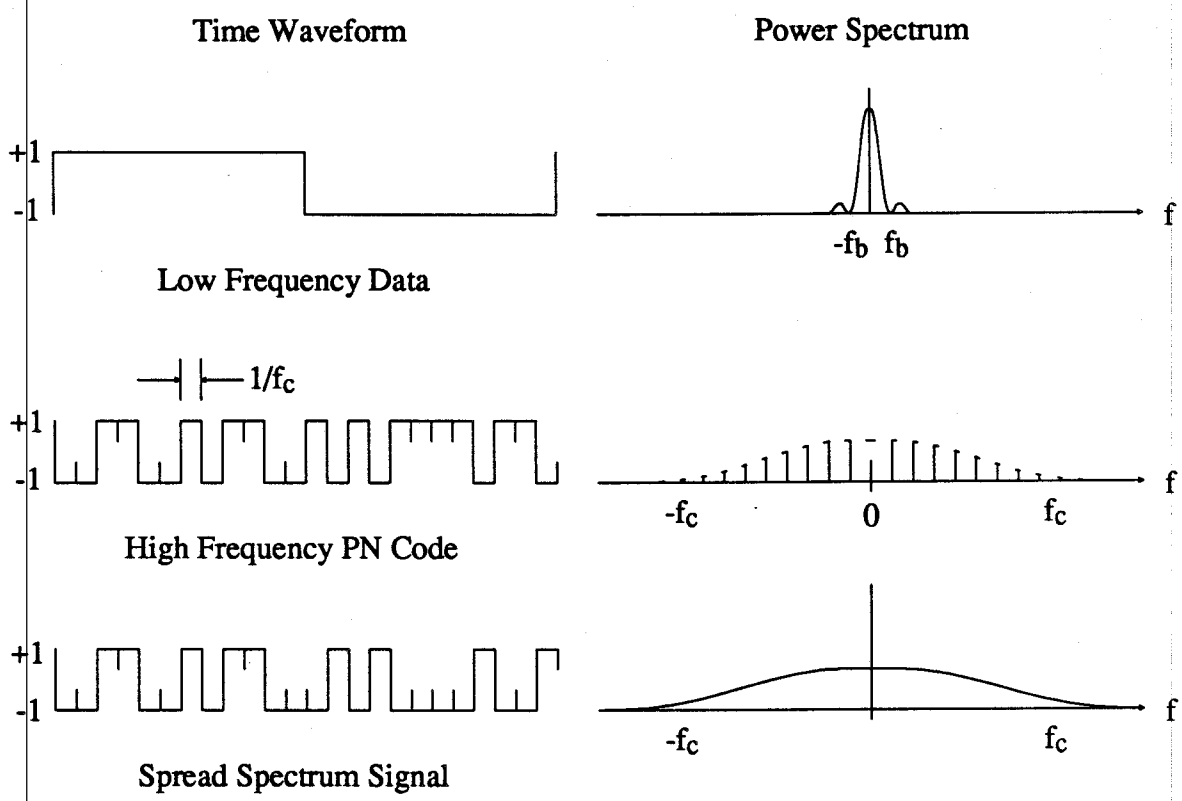


Fig. 2.3 DS SS Modulation

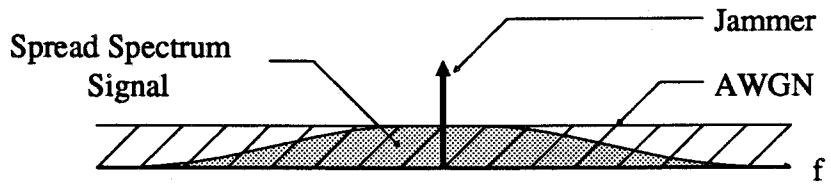
PN code sequence correctly synchronized with the transmitter, the SS signal will be despread to the original data bandwidth, while the jammer is spread over the entire PN sequence bandwidth as shown in Fig. 2.4 (c). By filtering the despread signal through a low pass filter of bandwidth equal to the data rate, most of the signal power is captured by the filter, while a large fraction of the spread jammer and noise power is rejected.

As indicated in Fig. 2.4, the antijamming and interference rejection performance of a SS system depends entirely on how wide the data spectrum is spread. The amount of performance improvement that is achieved through the use of SS is defined as the processing

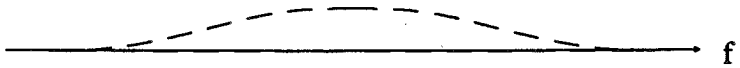
gain of the SS system. An often used approximation for processing gain is the ratio of the spread bandwidth to the data rate, that is

$$G_p = \frac{f_c}{f_b} \tag{2.12}$$

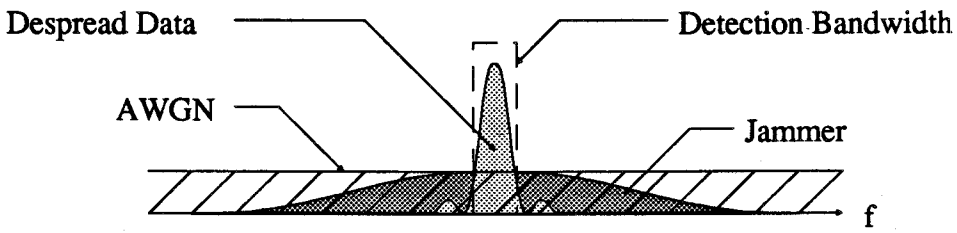
The higher the processing gain of a SS system, the better the antijamming and interference rejection performance can be achieved.



(a) Received Spread Spectrum Signal with AWGN and Jammer



(b) Receiver PN Code Spectrum



(c) Despread Result

Fig. 2.4 DS SS Demodulation

2.3. Direct Sequence Spread Spectrum Acquisition

Acquisition is an initial requirement for SS demodulation. At the receiver, despreading occurs once the local PN code phase is located to within ±1/2 chip from the true phase. The acquisition process is then disabled and a tracking loop is started to fine adjust the code phase and maintain the synchronization.

Acquisition usually involves cross-correlating the received sequence with the locally generated PN sequence for a period known as the dwell time, T_d. The correlation result is then compared with a pre-determined detection threshold. A correct PN code phase is declared if the correlation obtained exceeds the threshold. A number of acquisition schemes have been developed and reported in the literature [1, 17, 21, 22, 23, 24, 25]. Depending on the system specification, these acquisition schemes can change the system complexity and the acquisition time performance significantly.

The first acquisition scheme presented here is called serial search. At the beginning of the search, the entire PN code period or epoch is divided into a finite number of elements called cells. Sometimes only a portion of the epoch is searched and this "uncertainty region" is similarly divided into cells. Each of these cells is a discrete step, typically 1/2 of the PN code chip. The search is performed by sweeping through all possible cells, one at a time, and testing for the correct cell. The test involves correlating the received signal with a phase cell of the PN code for a dwell time period. The result is then compared with a pre-determined decision threshold. If the correlation result is less than the threshold, the cell is rejected. Subsequently, the local PN code generator steps to the next cell, and the

correlation test repeats. When the correlation result is greater than the detection threshold, the cell is accepted as a correct PN code phase and the acquisition process ends.

A number of search strategies fall into the serial search category, they are the so-called "straight line", dual search, Z, and Expanding Window (EW) serial search strategies [17, 21, 22, 23].

The structure of straight line serial search is illustrated in Fig. 2.5. Here the code phase uncertainty region is the entire epoch. Since the cell tested after the last one in the uncertainty region is exactly the same as the first one from which the search started, the search direction extends in a straight line as the time goes on. The search continues until the correct code phase is located. This search strategy is the simplest among all four serial search schemes. It requires the least hardware and thus is the most commonly used acquisition scheme. However, this method requires the longest acquisition time. Consider a PN code sequence of length 2^n-1, and the synchronizer steps 1/2 chip each time a new

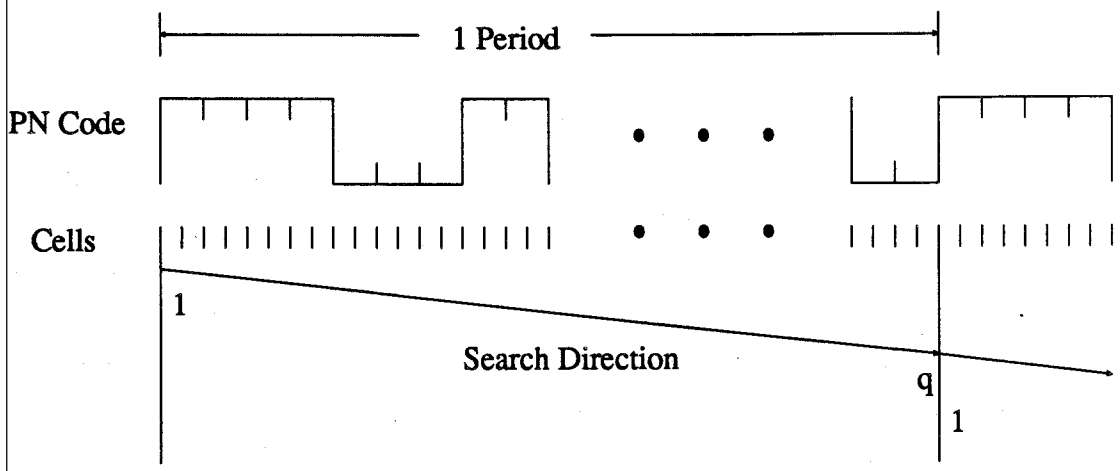
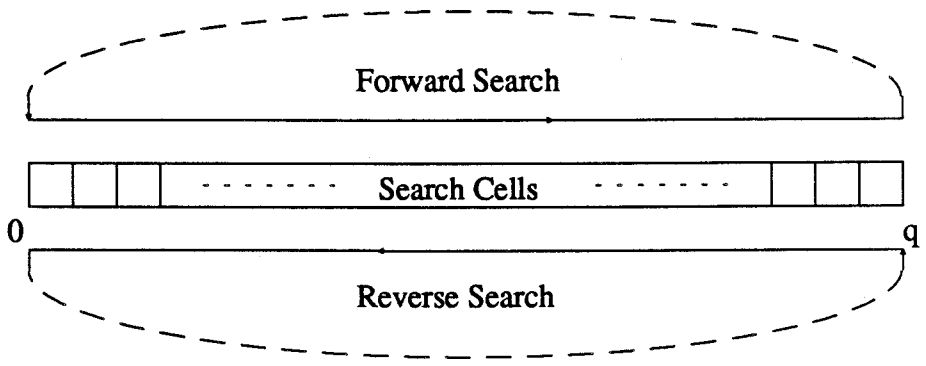


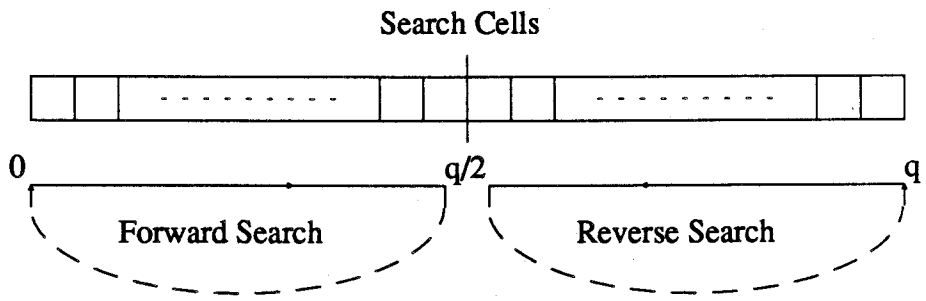
Fig. 2.5 Structure of Straight Line Serial Search

search begins. The maximum time required for a full search of the uncertainty region is $2(2^n-1)T_d$.

The idea of straight line serial search was extended to a dual search acquisition scheme [17]. In dual search acquisition, the search is carried out in two paths called the forward search path and the reverse search path. Coarse synchronization is declared when either path results in a positive code phase correction. There are two possible scenarios for conducting the dual search. In the first, shown in Fig. 2.6 (a), the dual search starts at the



(a) Full Area Dual Search



(b) Exclusive Area Dual Search

Fig. 2.6 Dual Serial Search Acquisition Schemes [17]

two ends in opposite directions and both search the entire code phase uncertainty region of q cells. In this scenario, if the correct cell is missed, the two searches will meet probably near the center of the uncertainty region, cross over and continue searching. When either path reaches the end, it returns to its starting cell and then search again in the same direction. Hence each path is a unidirectional search path. In the other scenario, shown in Fig. 2.6 (b), the uncertainty region is split into two equal mutually-exclusive search areas called the forward search area and the reverse search area. Both searches start at the two ends in opposite directions toward each other. When either search path reaches the center of the uncertainty region, it returns to its starting cell and then search again. The probability of finding the correct cell in each area is equal to half. It was shown that there is little difference in performance between the full area dual search and the exclusive area dual search. Compared to straight line serial search, dual search schemes effectively reduce the mean acquisition time by half, but the required hardware is doubled.

Both straight line and dual serial search is appropriate for acquisition without a priori information about the approximate position of the correct cell in the uncertainty region. However, if such information is available to the receiver, Z and EW search can be used to optimize the acquisition time. Fig. 2.7 illustrates various types of Z and EW search strategies.

In both Z and EW search, the uncertainty region can be considered to have a center and two edge boundaries. The center represents the most probable code phase position while the two edges represent the least probable position. The distance between the two edges corresponds to length of the uncertainty region, i.e., the total number of cells which is

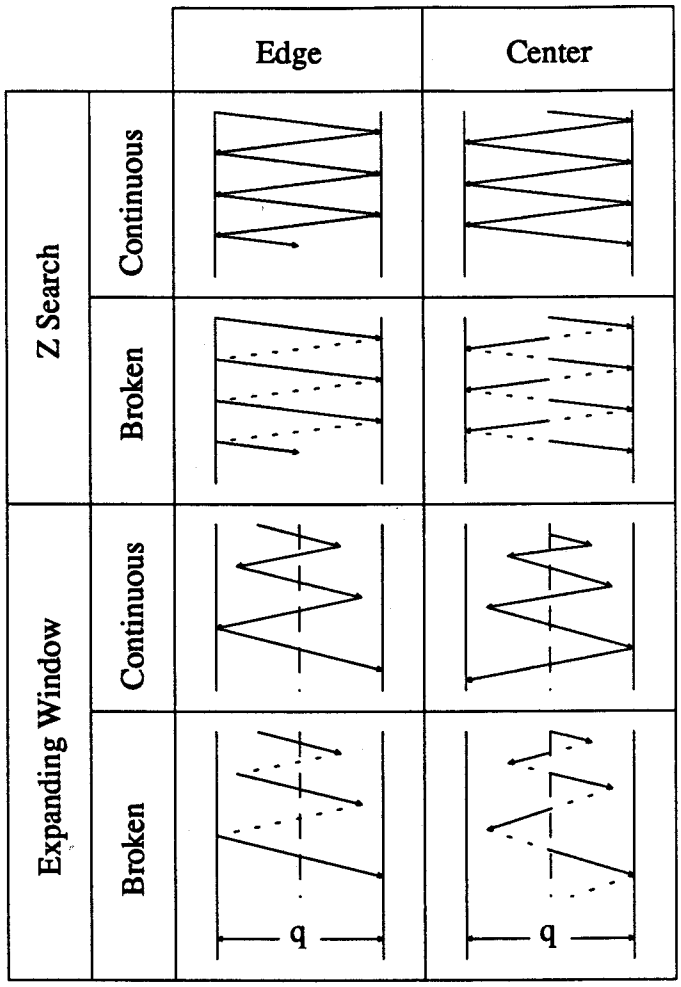
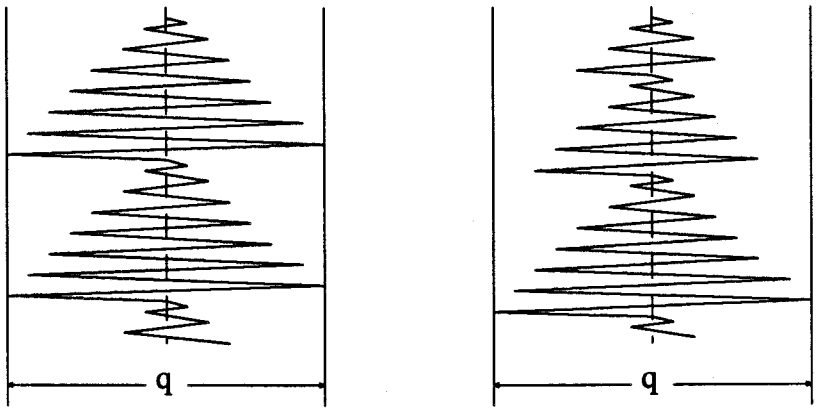


Fig. 2.7 Structure of Z and Expanding Window Search [23]

denoted as q. Referring to Fig. 2.7, all Z search methods have been defined with respect to the center and the edges. The search may be started at the center or either edge. In the case of continuous Z search, the search direction is reversed each time an edge is reached. In the case of broken Z search, however, the code phase is "rewound" to the starting position and then the search direction is reversed. In the special case of broken edge Z search where the uncertainty region extends throughout the epoch, two reversals occur in succession and a unidirectional straight line search results.

EW search is very similar to Z search. The only difference is that EW search starts with an initial sweep window which is only a fraction of the entire uncertainty region. Every time the search direction is reversed, the sweep window is extended. The window will continue expanding until it covers the entire uncertainty region. The sweep window stays unchanged once it reaches both edges. Jovanovic [23] had proposed and analyzed two modified EW strategies called uniformly expanding alternate (UEA) and nonuniformly expanding alternate (NUEA) search strategy, shown in Fig. 2.8, which outperforms all known serial search strategies.



Uniformly Expanding Alternate Serial Search Strategy (UEA)

Nonuniformly Expanding Alternate Serial Search Strategy (NUEA)

Fig. 2.8 UEA and NUEA Search [23]

Another acquisition scheme is called parallel search. This method uses a number of correlators in parallel. Each correlator measures the correlation of the received signal with a fixed cell. At the end of each dwell time, the result from the correlators are compared with each other. The cell resulting in the highest correlation is considered to be the correct code phase. Compared to serial search, this method increases the amount of hardware

substantially. Consider a PN code length of 2^n-1 and a cell step of 1/2 chip, as many as $2(2^n-1)$ correlators are required if a complete search of the entire phase uncertainty region is to be accomplished in a single search time [1]. However, the acquisition time can be as short as one dwell time under no noise condition. As an alternative, a mixed serial and parallel search can be used to take the hardware advantage of serial search and short acquisition time advantage of parallel search.

Yet another acquisition method, called Rapid Acquisition by Sequential Estimation (RASE), was first presented and analyzed by Ward [24]. The RASE system makes its best estimate of the first n received chips, presets the starting state of the local PN sequence generator with this estimate, and attempts to despread starting from this condition. Similar to serial search, the correlation between the received sequence and the local PN sequence is compared with a threshold at the end of each dwell time. If the correlation exceeds the threshold, the receiver has estimated the correct cell, and no further action is taken. However, if the correlation falls below the threshold, an incorrect estimate was made. A new estimate of another n received chips is then made, loaded in the PN code generator, and despreading is attempted again. This process continues until the correct estimate is obtained. At high signal-to-noise ratios (SNR), the correct code phase is likely to be estimated in the first attempt, so that the initial condition of the PN sequence generator is correctly loaded. This results in an acquisition time of nT_c. At reduced SNR, however, the generator may not be correctly loaded and additional loadings will have to be performed. As a result, RASE system is fast, but performance degrades as SNR decreases.

The last acquisition method is called variable dwell time scheme [25]. This scheme is designed to reduce the average test time in a serial search receiver. Most cells that are examined are not the correct code phase and yet many such cells must be examined. Therefore, instead of examining each cell for a fixed dwell time, the correlation result is continually compared to a varying decision threshold voltage during the decision period, and the cell is discarded if the voltage falls below the threshold anywhere in the time interval as shown in Fig. 2.9. In this way, many cells can be discarded in less than one fixed dwell time with a net decrease in search time. Synchronization is declared when the correlator output exceeds the final threshold value at the end of the decision period.

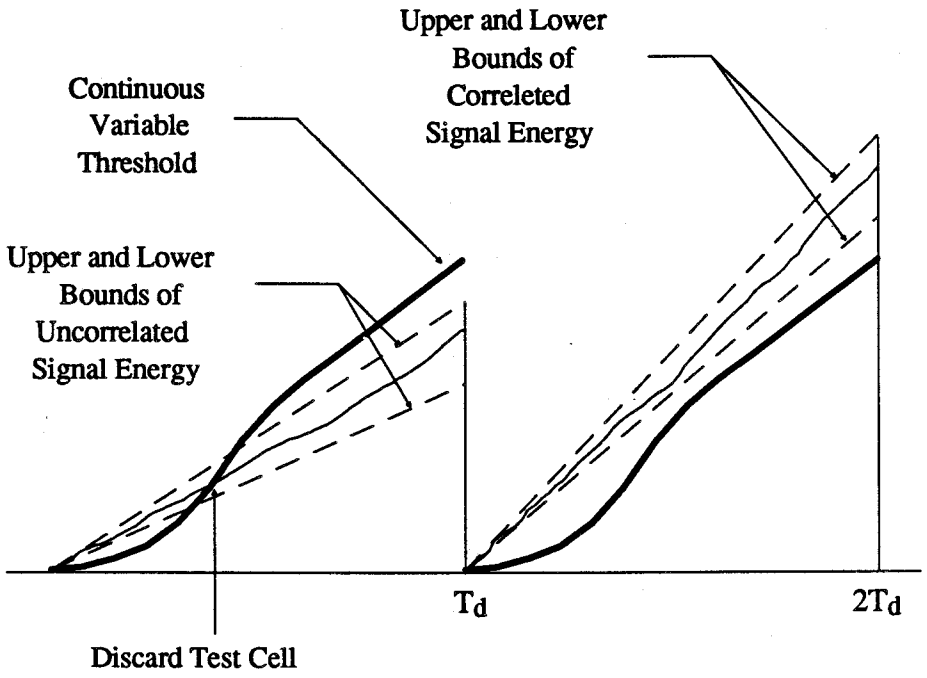


Fig. 2.9 Example of Variable Dwell Time Scheme

Chapter 3

DIRECT SEQUENCE SPREAD SPECTRUM MODULATOR

This chapter provides a description of the baseband DS SS modulator implemented for this project. Section 3.1 discusses the modulator specifications while section 3.2 presents the hardware implementation.

3.1. Modulator Specifications

Due to the maximum frequency limitation of some of the integrated circuits used, the data rate and chip rate were limited. The processing gain of the system was chosen to be 32. With a maximum sampling frequency of 1.6 MHz, the Charge Coupled Devices (CCD) used in the RAATS acquisition system limited the maximum data rate to 2400 bps. Lower data rates can be used if a higher processing gain is desired.

The choice of PN code length and code generator were somewhat arbitrary. However, the code length was selected such that it is comparable to that used in real systems. The PN code length used in four satellite systems is shown in Table 3.1. Based on these systems, the PN code length used in this project was chosen to be 1023 chips/cycle generated by a 10-bit shift register. For simplicity, the maximal length code generator was chosen to be (3, 10) from Table 2.1.

Table 3.1 PN Code Length of Four Satellite Systems [5, 8, 26, 27]

Satellite Systems	PN Code Length (chips/cycle)
Navstar Global Positioning System (GPS)	1023
Tracking and Data Relay Satellite System (TDRSS)	1023 (Ku-band), 2047 (S-band)
Sakura (Japanese communication satellite), CDMA	2047
VSAT Network, Ku-band DS CDMA	1023

Since acquisition time performance of the serial search and the RAATS scheme at various SNR is the focus in this project, AWGN or jamming signal must be able to add to the SS signal during the transmission. In order to obtain data for a wide range of SNR, an adjustable amplitude is desirable for the baseband SS signal.

3.2. Modulator Implementation

The PN code generator consists of a 10-bit shift register and a 2-input XOR gate. Stages 3 and 10 of the shift register are tapped, XORed, and fed back to the input of the first stage. The modulator is clocked by a 921.6 kHz square wave. This clock is divided by twelve and the resulting 76.8 kHz signal is then used to clock the PN code generator. Hence the chip rate is 76.8 kchips/second.

The basic block diagram of the baseband DS SS modulator is illustrated in Fig. 3.1. The design is very straight forward. An XOR gate forms a multiplier to perform the modulation. A potentiometer is placed at the output of the XOR gate to give the signal an adjustable amplitude. Before entering an inverting summing amplifier, the signal is ac

coupled so that a bipolar signal is created. AWGN and jammer can be added via the inverting summing amplifier. The complete schematic of the modulator is included in Appendix A. Note that carrier modulation such as BPSK could be performed at point X if desired.

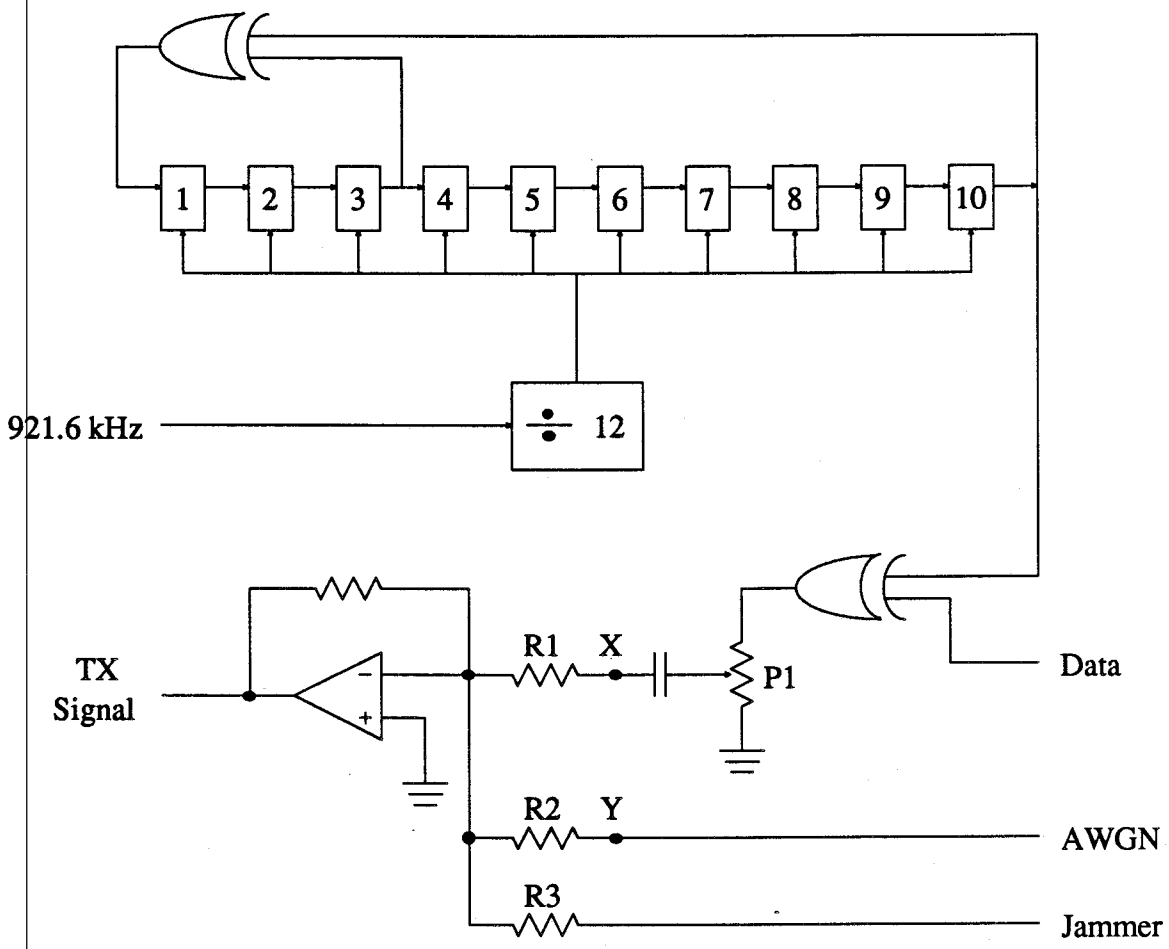


Fig. 3.1 Block Diagram of the DS SS Modulator Implemented

Referring to Fig. 3.1, different SNR values can be achieved by either adjusting potentiometer P1 or the noise input. Transmitted SNR is calculated as

$$SNR = 10 \log \frac{P_{signal}}{P_{noise}} \tag{3.1}$$

where P_{signal} is the signal power and P_{noise} is the noise power before the summation. Since R1, R2, and R3 are of the same value and the negative terminal of the summing amplifier is at a virtual ground, SNR is calculated as

$$SNR = 10 \log \frac{\frac{(V_{signal})^2}{R_1}}{\frac{(V_{noise})^2}{R_2}}$$

$$= 20 \log \frac{V_{signal}}{V_{noise}} \tag{3.2}$$

where V_{signal} is the true rms voltage of the signal at point X, and V_{noise} is the true rms voltage of the noise at point Y. A Fluke 8920A True RMS Voltmeter was used to measure the signal and noise voltages required for different SNR.

Chapter 4

SERIAL SEARCH ACQUISITION

This chapter gives a description of the serial search acquisition system which was used as a basis of comparison with the RAATS system. Section 4.1 provides an overview while section 4.2 describes the search/verification/lock strategy, and section 4.3 provides the details of the hardware implementation.

4.1. System Overview

The serial search acquisition system investigated in this project employs a Search/Verification/Lock (SVL) strategy with two lock states and double dwell times. The system block diagram is shown in Fig. 4.1. In order to achieve code phase synchronization, the received SS signal is first multiplied with a locally generated PN code sequence which starts with an arbitrary code phase. When a cell is within ±1/2 chip from synchronization, the signal spectrum after multiplication is compressed back to the original data bandwidth. The product is then filtered by a low pass filter. By setting the low pass filter bandwidth equal to the data bandwidth, most of the energy contained in the data can be captured. On the other hand, if the cell is more than 1/2 chip apart from the correct phase, the received signal will remain spread. As a result, only a very small portion of energy is contained in the data bandwidth, hence the output of the low pass filter is very small. The filter output

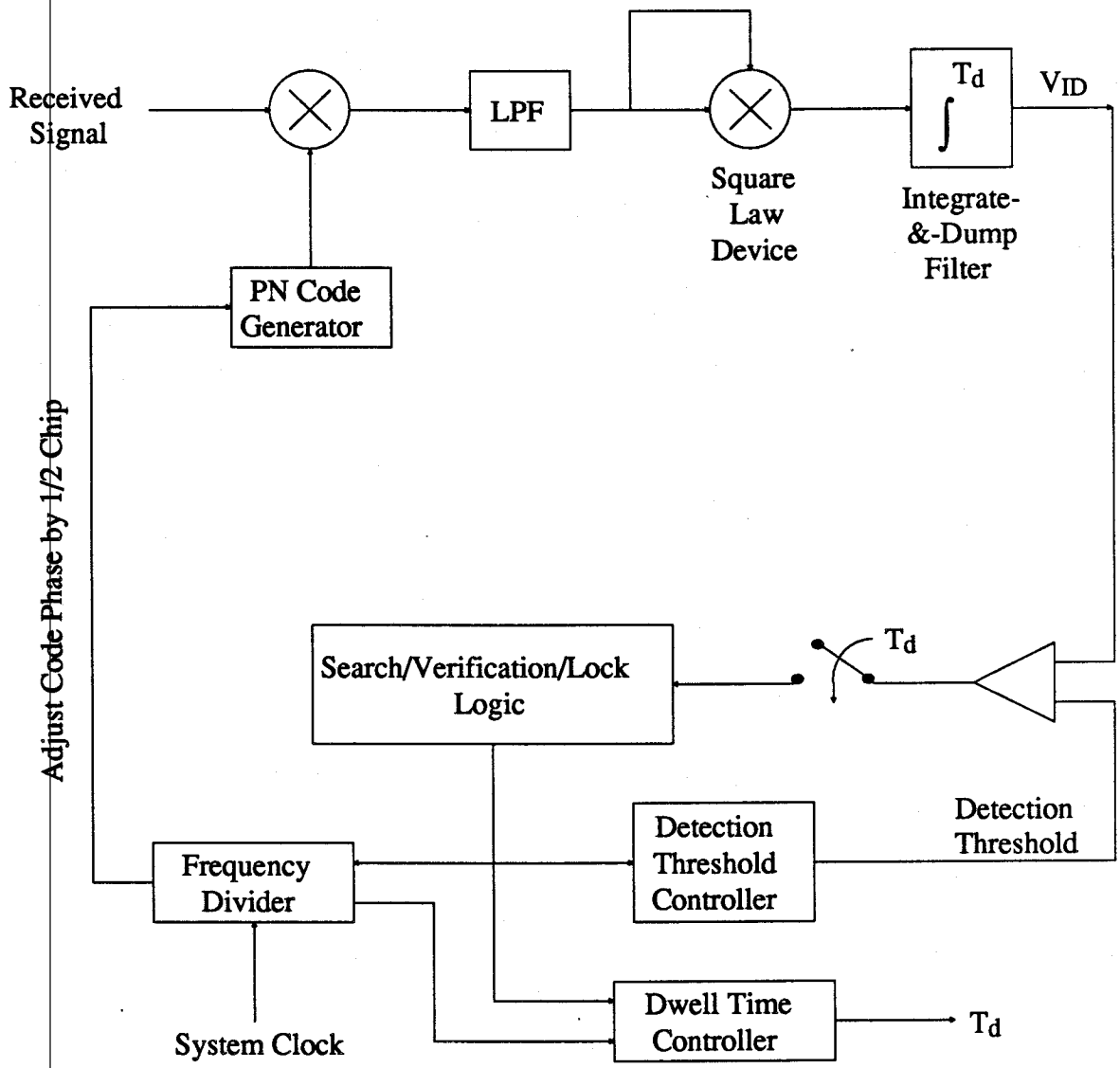


Fig. 4.1 Block Diagram of Serial Search System

is brought into an energy detector which consists of a square law device and an integrate-and-dump filter. The integration time is known as the dwell time of the SVL strategy. The result from the integration V_{ID} is then compared with a predetermined detection threshold at the end of each dwell time. A different threshold is used for different dwell time interval. If V_{ID} is less than the threshold, the SVL strategy will reject the current test cell and signal the PN code generator to step to the next cell. On the other hand, if V_{ID}

is greater than the threshold, the SVL can either re-examine the cell again to confirm the result or immediately declare synchronization. The choice of operation is controlled entirely by the SVL logic. In addition, the SVL logic controls the switches of dwell times and detection thresholds.

4.2. Search/Verification/Lock Strategy

Search/Lock Strategy (SLS) [14] is a well known algorithm used for PN code acquisition. Pan et al. had theoretically analyzed factors affecting the acquisition performance of serial search synchronization with SLS, but renamed it as Search/Verification/Lock (SVL) strategy [16]. The investigation showed that acquisition performance is highly sensitive to detector thresholds.

With SVL strategy, serial search code phase synchronization becomes a two step process with a search mode and a lock mode. In search mode, coarse synchronization is performed in a serial "trial and error" search where all possible code phases will be examined. In lock mode, fine adjustment of code phase is performed by using a code tracking loop. The transition between the search mode and the lock mode is governed by the SVL strategy used by the SS synchronizer.

The SVL strategy used in this project is illustrated in Fig. 4.2. It has one search state, one verification state and two lock states. When a cell is tested in search (S) mode and the integrate-and-dump filter output is less than the predetermined detection threshold at the end of dwell time T_{d1}, the cell is rejected and the next cell is tested. On the other hand, the cell is accepted if the filter output exceeds the detection threshold. Upon acceptance of a

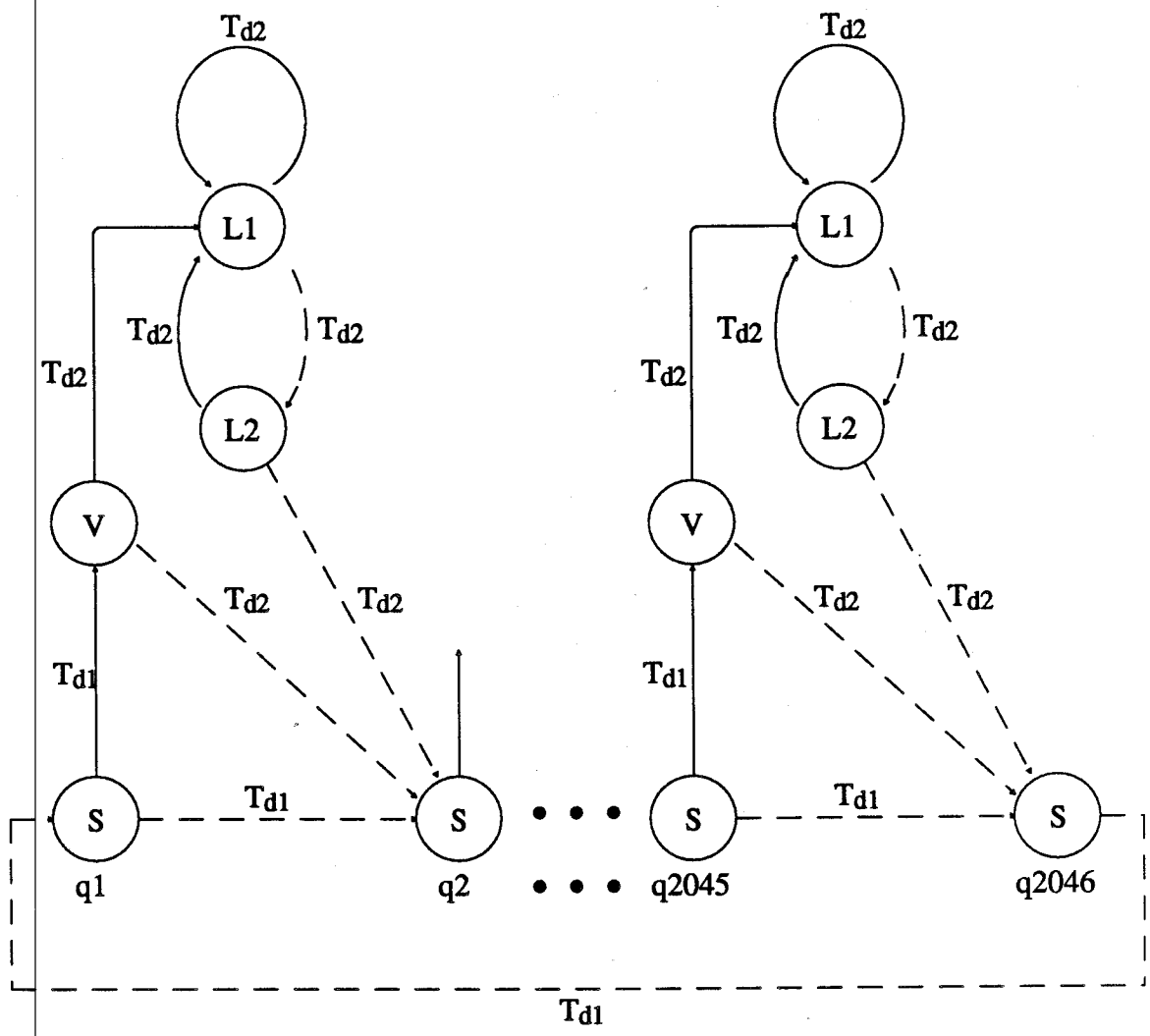


Fig. 4.2 2-Lock-State Search/Verification/Lock (SVL) Strategy

cell, the algorithm enters its verification (V) mode and the cell is re-tested with dwell time T_{d2}. If the test fails at the end of T_{d2}, the cell is dismissed as being incorrect and the algorithm re-enters the search mode with a new cell. The transition from search mode to verification mode and back to search mode is known as a detour. If the cell passes the test at verification mode, the algorithm enters the first lock (L1) mode where the cell, dwell time, and detection threshold remain unchanged and synchronization is declared. The algorithm continues performing tests at L1 mode and the system maintains the alignment as long as the tests are successful. If a test is failed at any time, the algorithm enters the second lock (L2) mode where the cell is given one more test before dismissal. If the cell passes the test at L2 mode, the system re-enters L1 mode. However, if the cell fails the test at L2 mode, the system returns to search mode with a new cell. The transition from search mode to verification mode to lock mode and back to search mode is known as a false alarm. To obtain rapid acquisition and robust true lock, dwell time T_{d1} is made shorter than dwell time T_{d2}. In this project, T_{d2} is chosen to be five times T_{d1}.

4.3. Hardware Implementation

In the following sections, the five subsystems in the serial search acquisition system, namely the correlator, dwell time controller, detection threshold controller, SVL controller, and PN code phase controller are presented. The complete set of schematics are included in Appendix B.

4.3.1. Correlator

One key element in SS acquisition systems is the device which performs the cross-correlation between the received sequence and the local PN code sequence. There are three classes of devices which can perform this function [25]. The first class is the PN matched filter (or transversal filter). This is a passive device which has an impulse response equivalent to the time reverse of the PN sequence. When a received signal enters a PN matched filter, it is multiplied by appropriate weighting coefficients and summed. The weights take on the equivalent value of the PN sequence, i.e., ±1, and serves as a stored reference code. The output is the correlation of the PN sequence and a finite segment of the incoming signal. Examples of such devices are Surface Acoustic Wave (SAW) matched filters [28] and Charge Coupled Device (CCD) matched filters [29, 30, 31]. The second class is the convolver. This is an active device which multiplies and integrates the incoming sequence by a time reversed replica of the PN sequence while both signals counter-propagate across each other in a delay line. An example of such a device is a SAW monolithic convolver. The last class is the correlator. This is an active device which performs a chip-by-chip multiplication and integration over a particular dwell time. An example of this device is an integrate-and-dump filter. Because of its low cost and simple structure, the correlator is the most commonly used detection device, and is the type used in this project.

A corrector was implemented as illustrated in Fig. 4.3. The received baseband DS SS signal is first multiplied with the local PN code sequence with an analog multiplier (AD532). When despreading occurs, a large amount of signal energy is concentrated within the data bandwidth which in this case was 2400 Hz. A first-order low pass filter (R_1 C_1) of

bandwidth 1200 Hz is used to capture the main lobe of the despread data spectrum. The 1200 Hz filter bandwidth is chosen such that the roll off of the filter matches approximately the shape of the energy spectrum of a 2400 bps binary data. The output of the low pass filter is then fed into an energy detector which consists of a square law device and an integrate-and-dump filter. The time period of the integrate-and-dump filter is T_{d1}. A Field-Effect-Transistor (FET) is used to perform the "dump" operation. At the end of each dwell time interval, a narrow pulse is sent to the gate of the FET via the TIME_OUT line. These pulses short the charging capacitor (C_2) of the integrator, thus causing it to discharge through the FET to ground potential. Since the resistor of the FET is very small when it is turned on, the discharge time period is extremely short compared to the dwell time. As a result, the integrator is reset to zero potential and integration resumed when the FET is turned off.

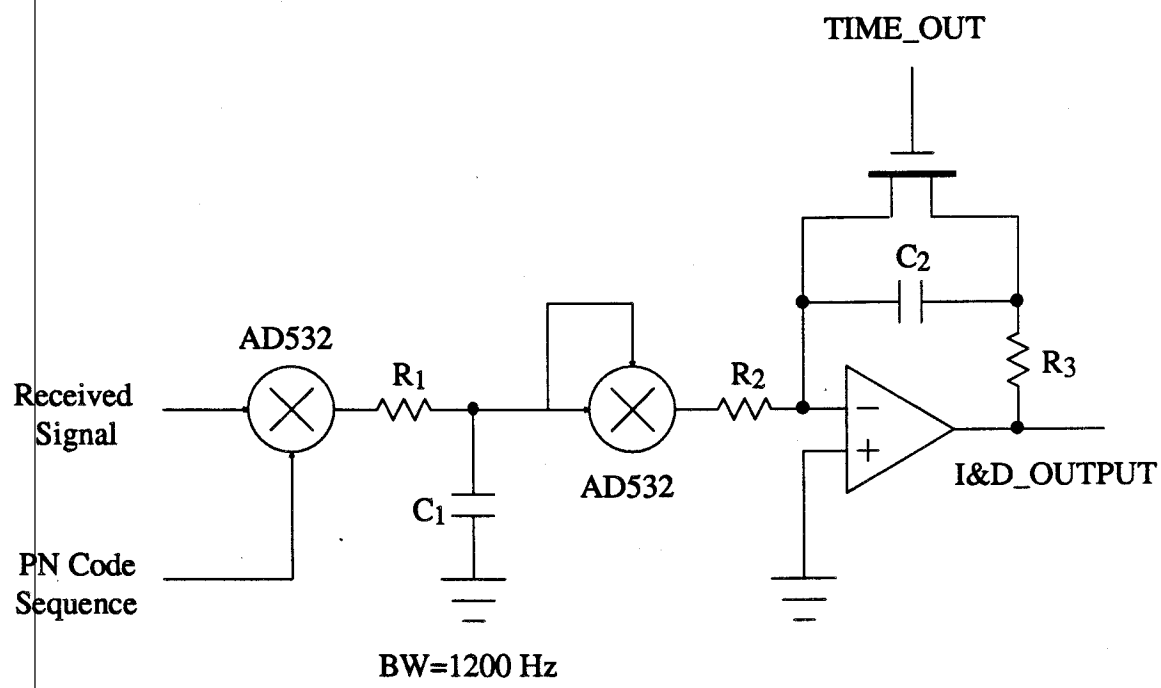


Fig. 4.3 Correlator

4.3.2. Dwell Time Controller

As indicated in Fig. 4.2, the SVL strategy has two dwell times. Dwell time T_{d1} is used only in the search mode while dwell time T_{d2} is used in either the verification or the lock mode. Since rapid acquisition is desired, the search mode dwell time at each cell should be as short as is practical. Short dwell time results in a fast search through the entire code phase uncertainty region, but a low probability of detecting the correct synchronization when it occurs. Conversely, long dwell time results in a slow search but a high probability of detecting the correct cell. As a result, to obtain rapid acquisition and robust true phase lock, T_{d1} was made shorter than T_{d2}. With the available hardware, the most convenient choice for T_{d1} is 8 data bit times or 256 chip times long (3.33 ms) and T_{d2} equal five times T_{d1}.

Fig. 4.4 is a block diagram of the dwell time controller. The PN code generator clock signal PN_CLK is brought into a divide-by-256 counter which produces a narrow pulse every 256 chips. This output signal, labelled as TD1_TIME_OUT, gives an indication of the end of dwell time T_{d1}. Similarly, the divide-by-5 counter followed produces a narrow pulse every five TD1_TIME_OUT pulses, thus creating a T_{d2}-to-T_{d1} ratio of five. Both TD1_TIME_OUT and TD2_TIME_OUT are connected to a 2-to-1 multiplexer. The signal TD_SEL selects the appropriate dwell time according to the state of the SVL strategy. In search mode, TD_SEL is asserted low and T_{d1} is selected. In verification or lock mode, TD_SEL is asserted high and T_{d2} is selected. The selected dwell time appears on the signal line denoted TIME_OUT.

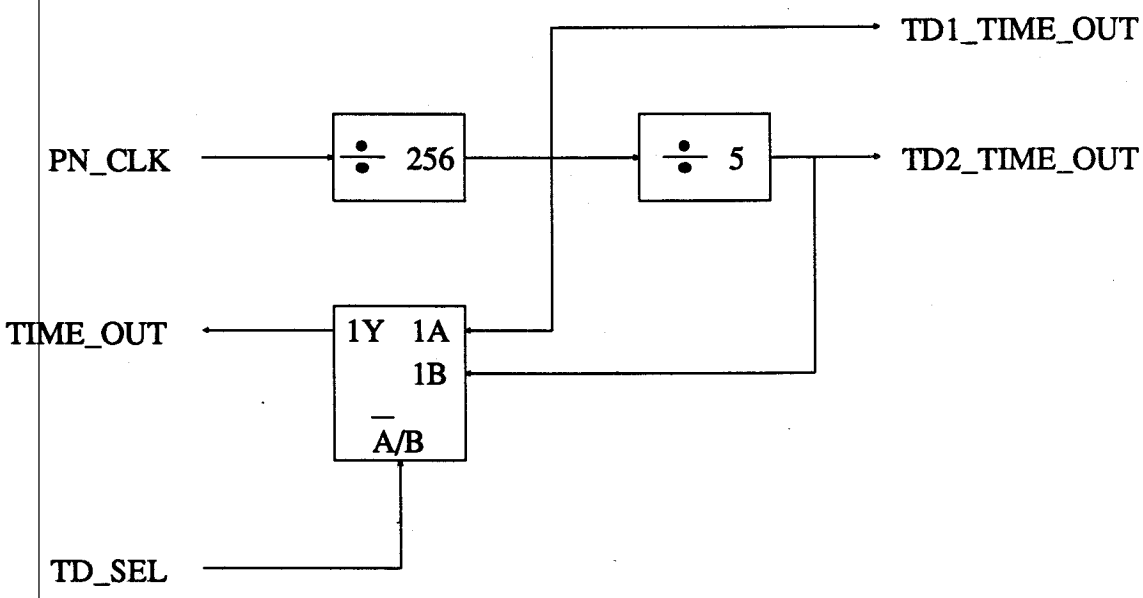


Fig. 4.4 Dwell Time Controller

4.3.3 Detection Threshold Controller

Different detection threshold is required for different dwell time. These thresholds have significant effects on the mean acquisition time of the acquisition system. The detection threshold in search mode V_{th1} affects the occurrence of detour while the detection threshold in verification or lock mode V_{th2} affects the occurrence of false alarm. If these two threshold voltages are set too high, the synchronizer might never lock with the incoming SS signal. If these two threshold voltages are set too low, however, false lock might result.

Since a integrate-and-dump filter is used and T_{d2} is 5 times T_{d1}, V_{th2} should be approximately 5 times larger than V_{th1}. In order to determine V_{th1} and V_{th2}, four signal envelopes at the output of the integrate-and-dump filter have to be measured. For convenience, these envelopes were called A, B, C, and D. Envelope A referred to as the uncorrelated signal plus noise integration envelope with dwell time T_{d1}. Envelope B

referred to as the correlated despread (within ±1/2 chip alignment) signal plus noise integration envelope with dwell time T_{d1}. Envelope C referred to as the uncorrelated signal plus noise integration envelope with dwell time T_{d2}. Envelope D referred to as the correlated despread (within ±1/2 chip alignment) signal plus noise integration envelope with dwell time T_{d2}. The relationship between envelopes A and B, and C and D are illustrated in Fig. 4.5.

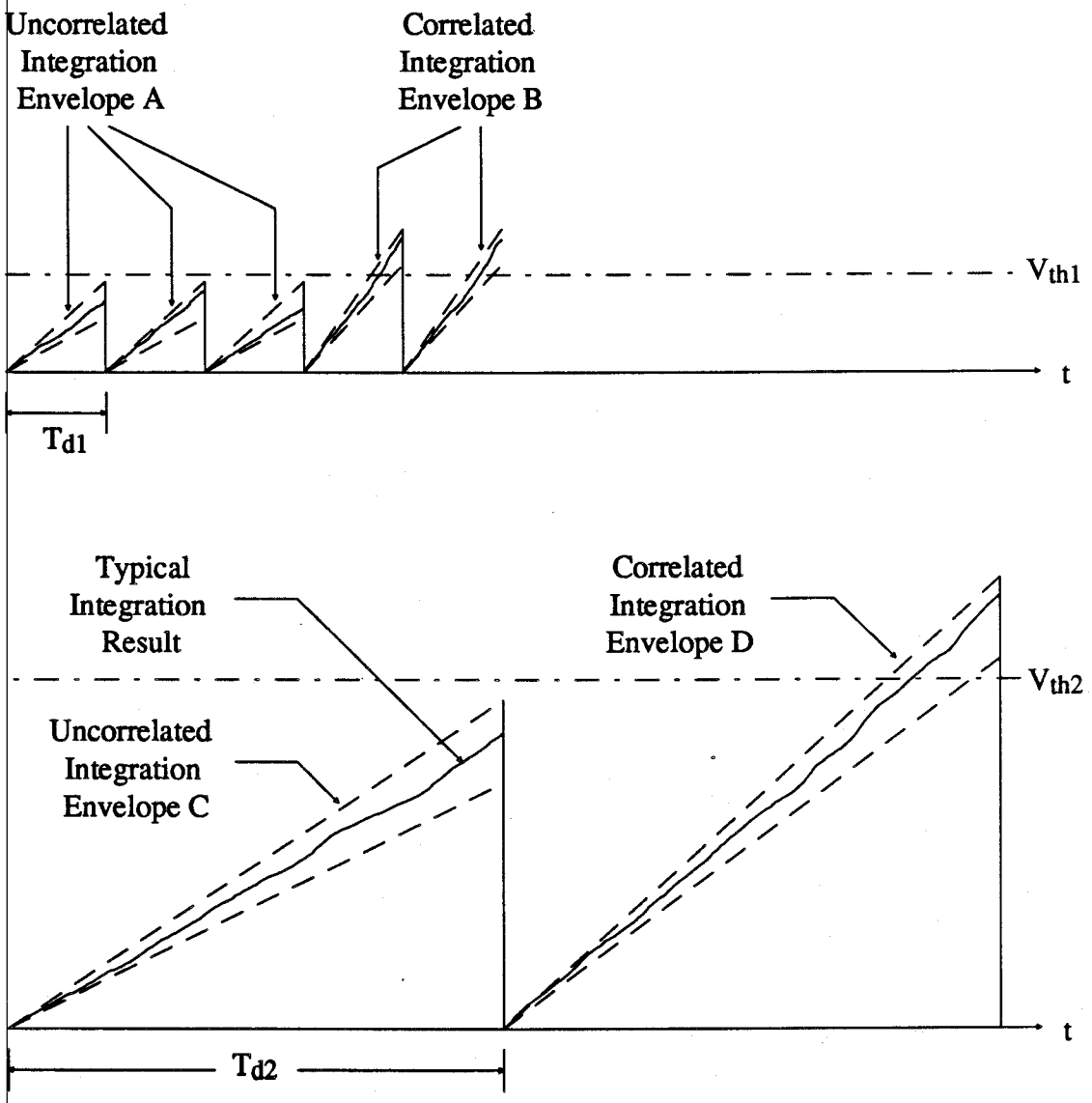


Fig. 4.5 Integration-and-Dump Filter Output Envelope

The schematic of the detection threshold controller is illustrated in Fig. 4.6. An analog multiplexer (MC4053) is used to switch between the thresholds V_{th1} and V_{th2} which can be varied from 0 to +12 volts. In search mode, TD_SEL is asserted low and V_{th1} is selected to compare with the integrate-and-dump filter output while in verification or lock mode, TD_SEL is asserted high and V_{th2} is selected. Since the analog multiplexer is a CMOS device and TD_SEL is a TTL signal, a transistor interface is required. The result from the comparison is then sent to the SVL controller.

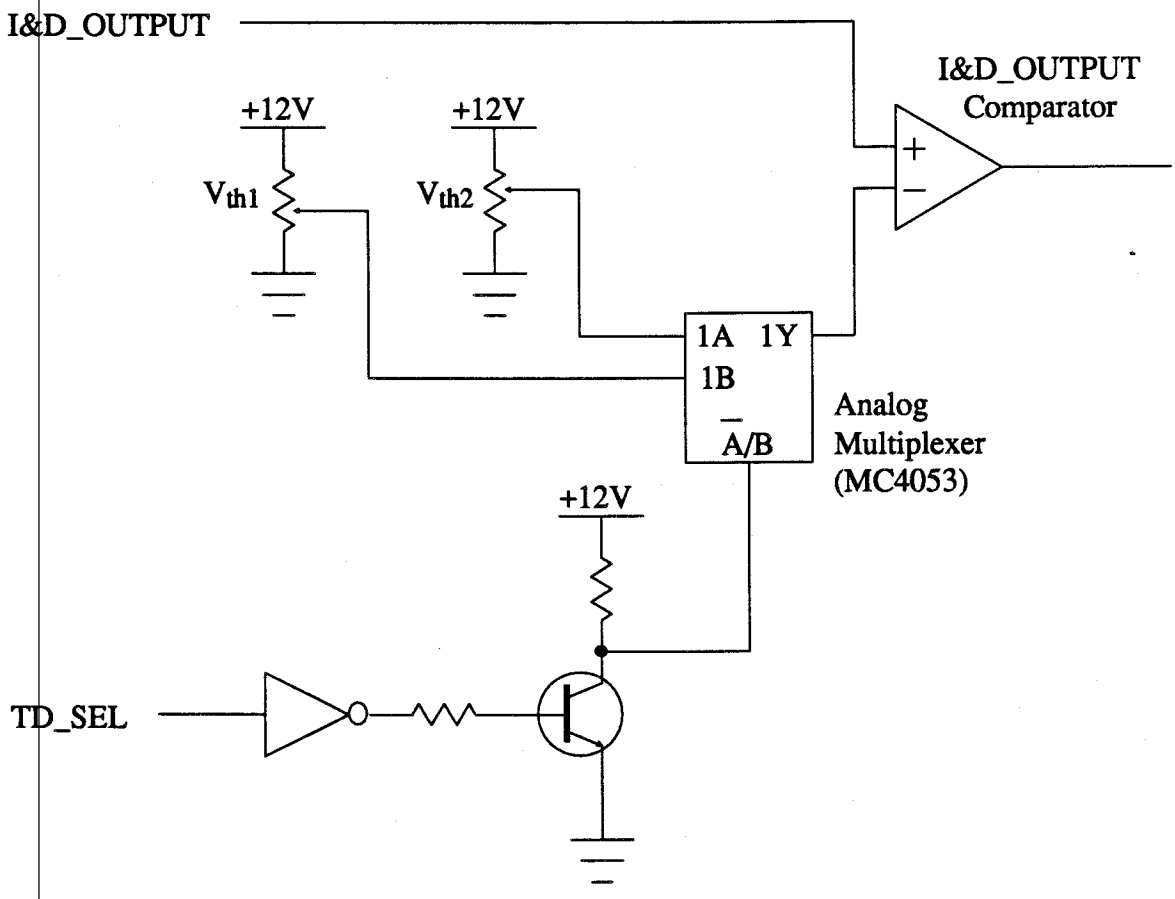


Fig. 4.6 Detection Threshold Controller

4.3.4. SVL Controller

A section of the SVL state flow diagram shown previously in Fig. 4.2 is extracted and redrawn in Fig. 4.7. There are only two possible transition paths from each operation mode, and a total of eight possible transition paths in each section of the SVL state flow diagram. At the end of each dwell time, a comparison is made between the integrate-and-dump filter output and the detection threshold. A 3-bit register is used to store

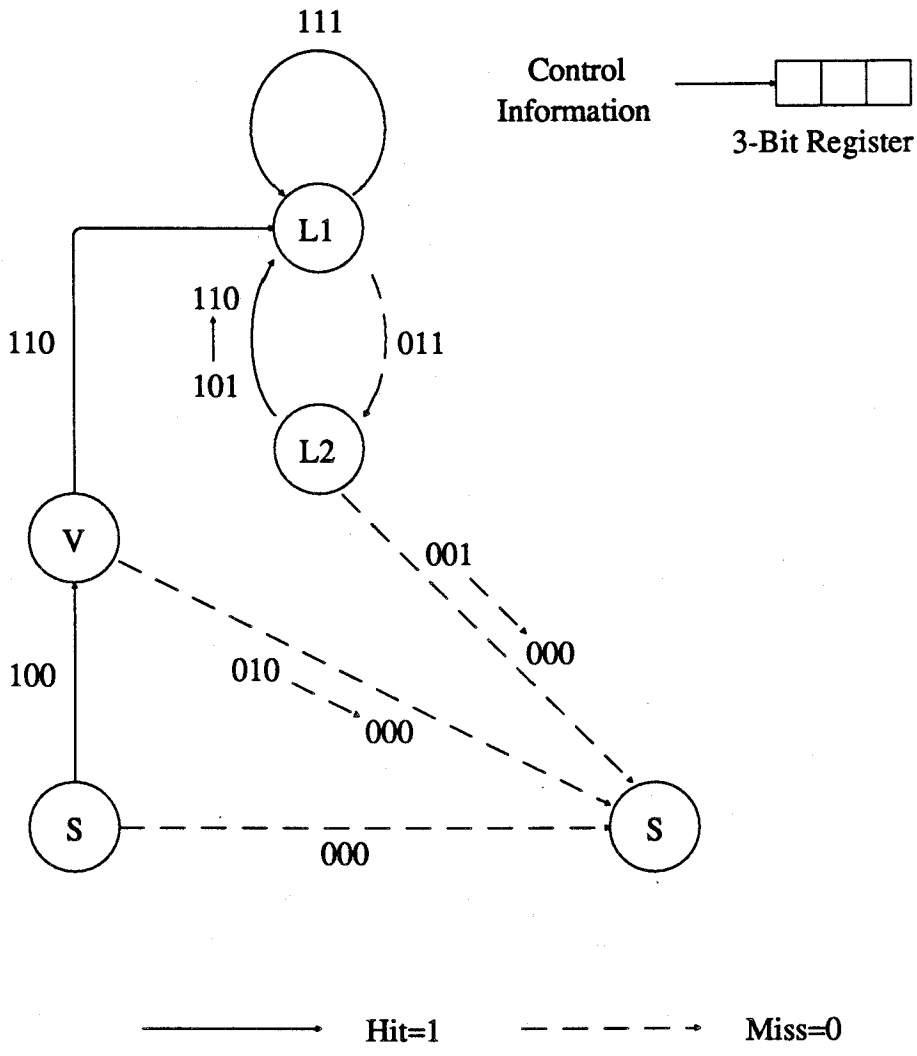


Fig. 4.7 SVL Controller Transition Conditions

the three most recent results from the comparison (hits and misses). Referring to Fig. 4.7, the 3-bit binary number labelled along each of the transition path indicates the required register content that causes the transition. A "1" represents a hit while a "0" represents a miss. A 3-to-8 line decoder is used to decode the condition of the register. Hence the state of the SVL strategy is continuously monitored and appropriate actions are carried out accordingly.

Note that the register content must be all zero at the beginning of every new search. In order to obtain correct transition of the SVL strategy, two 2-to-1 multiplexers are used to control the input to the second and the third stage of the register as shown in Fig. 4.8. The register content of 100 and 011 are detected by the 3-to-8 line decoder and a NAND gate.

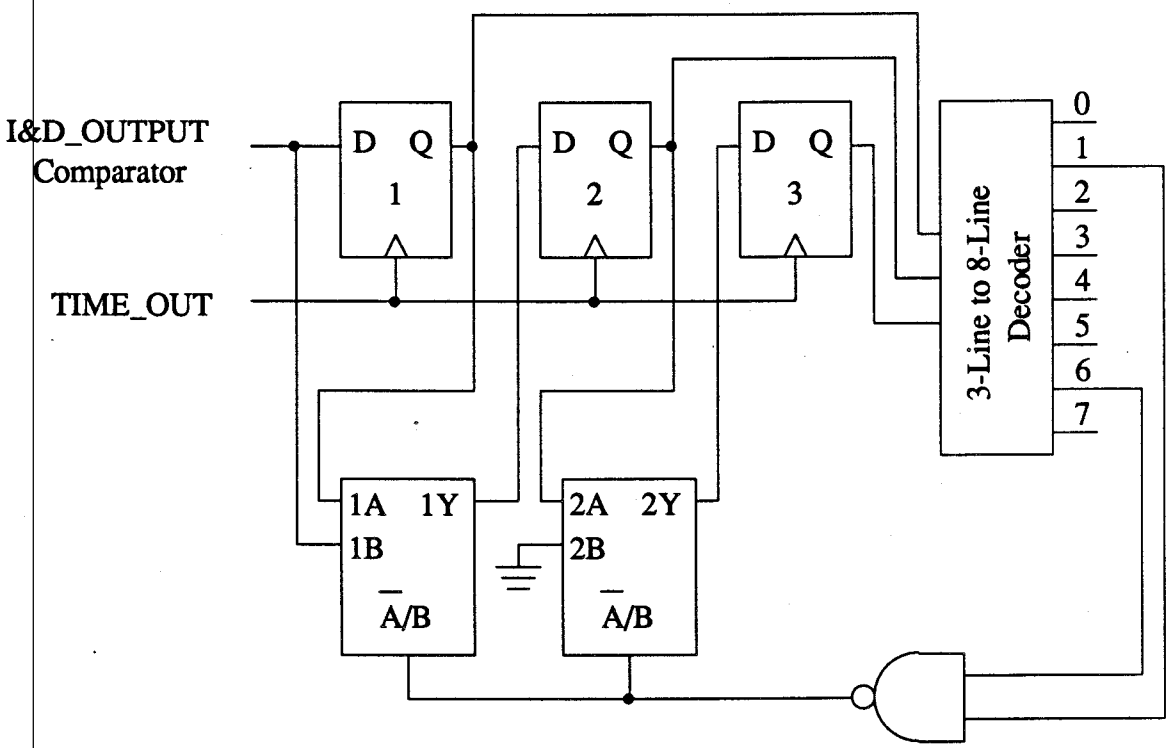


Fig. 4.8 SVL Controller

When neither one is detected, the input to the second and the third stage of the register is connected to the output of the first and the second stage, respectively. When either 100 or 011 is detected, however, the incoming data is routed to stages 1 and 2, and the input of stage 3 is grounded.

4.3.5. PN Code Phase Controller

As shown in Fig. 4.2, every time the correlation test fails in search mode, verification mode or lock 2 mode, the PN code phase steps 1/2 chip. Fig. 4.9 illustrates how this was achieved in the design of the PN code phase controller.

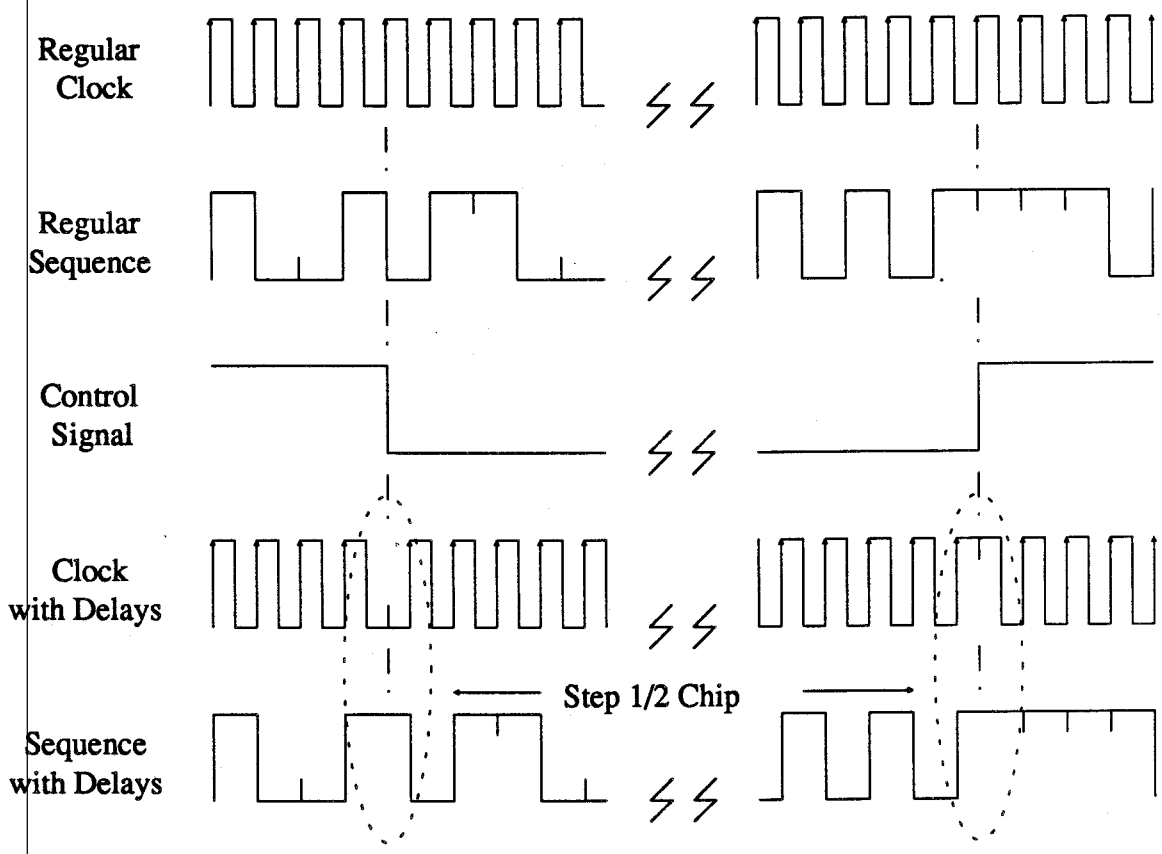


Fig. 4.9 Generation of New PN Code Phases

Referring to Fig. 4.9, the top two waveforms are the PN code generator clock signal and the PN code sequence at normal situation. A control signal is introduced which inverts its logic level whenever 000 is detected in the SVL controller at the end of a dwell time interval. By XORing this control signal with the PN code generator clock, the clock signal is flipped each time the control signal is inverted. As a result, the PN code generator clock signal is delayed by half cycle each time a new search is required. Hence the PN code sequence is delayed by 1/2 chip.

Chapter 5

RAPID ACQUISITION BY ACCUMULATION OF TRANSITION SEQUENCE

This chapter presents a new acquisition scheme called Rapid Acquisition by Accumulation of Transition Sequence (RAATS). Section 5.1 provides an introduction to the RAATS while Section 5.2 explains the structure in detail. Section 5.3 presents the hardware implementation of the system. Actual waveforms were recorded to illustrate the operation.

5.1. Acquisition Scheme

RAATS is an acquisition scheme which has been developed from the Rapid Acquisition by Sequential Estimation (RASE) described in Section 2.3. In RASE, a code phase estimate of the PN sequence is made after carrier demodulation, however, this estimate can be corrupted by polarity reversals of the modulating data signal. In contrast, RAATS detects transitions in the transmitted signal and is nearly insensitive to polarity reversals of the data (or of the "carrier" if of low frequency). Transition information is stored in a cyclic accumulator with the number of storage cells equal to the code length (or an integer multiple of the code length) and a period equal to the PN code epoch. The accumulation provides an averaging effect and allows the estimate of the transition sequence to build up gradually

with time and the effects of noise to be reduced by averaging. This permits acquisition to be performed in presence of high noise.

A PN code phase estimate is made by matching the pattern of transitions in the expected PN sequence with the averaged received transition sequence. An optimum detection would be a complete pattern match with the transition sequence. For the purpose of this investigation, the match was only tested for a small portion which corresponded to the longest run in the PN sequence. The longest run resulted in the longest period with no transition and was thus easy to detect with simple logic.

5.2. System Overview

Fig. 5.1 is a block diagram of the RAATS system. The transmitted SS signal received by the receiver is first delayed by exactly one chip time. This delayed version of the SS signal is then multiplied with the received signal. Without added noise, the received sequence has only ±1 value, a +1 in the product indicates no transition in the sequence, and a -1 indicates a transition in the sequence. Hence the product provides the location of the transition in the received signal. In presence of noise, the polarity of the multiplier output will, in most cases, still indicate transitions in the received sequence. To reduce the effect of noise by averaging, this transition signal is brought into an analog delay line which delays the transition signal by exactly one PN code period. The output from the delay line is summed with the new transition sequence and fed into the analog delay line again. This feedback structure linearly accumulates the periodic transition signal on a per PN cycle basis.

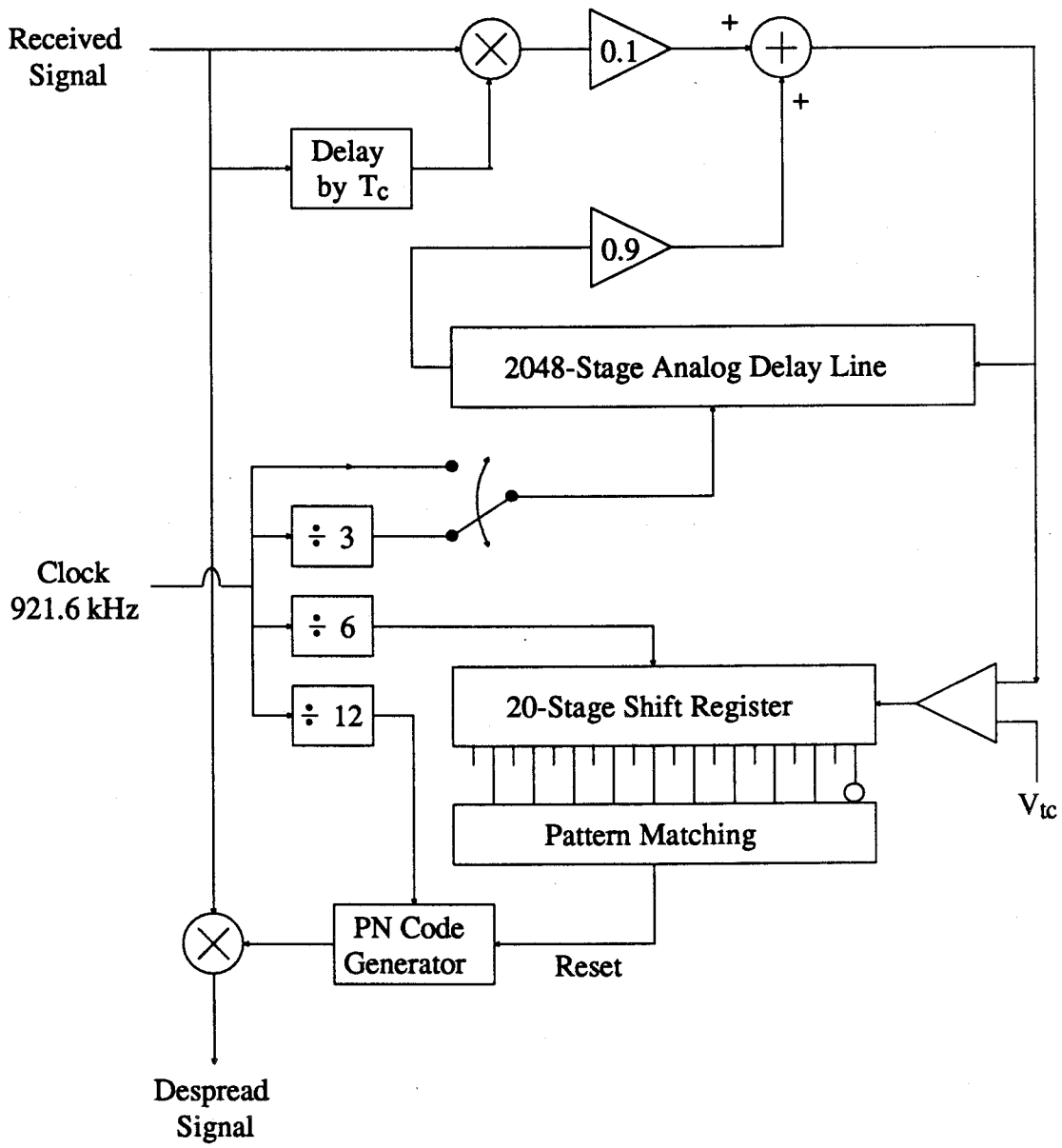


Fig. 5.1 Block Diagram of RAATS Code Phase Estimator

The feedback coefficients were set to provide an accumulation time constant of approximately 10 epoch. The results from the accumulation are used to estimate code phase.

Code phase estimation is performed by detecting a specific pattern in the transition sequence. Since the PN code sequence used is a maximal length sequence generated by a

10-bit shift register, there will be a run of ten ones (or ten zeros) in each period of the sequence. Hence there will be no transition for precisely nine chips in the accumulated transition signal. This pattern will repeat with a period of one epoch. The RAATS implemented here uses this pattern as an indication of the start of the PN epoch. This test pattern can be further expanded by realizing that a zero will be presented in front of and after the nine consecutive ones. Hence a more accurate estimate can be made by extending the test pattern. For hardware simplicity, the test pattern 0111111111 was selected with the zero being the most recent data.

To detect the test pattern, the accumulated transition signal is compared with a threshold, V_{tc}, which is set at approximately the peak of the accumulated transition signal. This hard-decision result is shifted into a shift register where the test pattern is searched. Since a typical tracking loop is capable of tracking the correct code phase within $\pm 1/2$ chip of alignment, the length of the shift register is twice that of the test pattern and the clock rate of the shift register is twice that of the PN chip rate. Hence each chip of the estimated transition sequence is sampled twice as it propagates through the shift register as shown in Fig. 5.2. As indicated in Fig. 5.2 (b) and (c), the test pattern can be detected twice which correspond to +1/2 chip alignment and exact synchronization, respectively. This arrangement provides an estimation of the PN code transition sequence to one half chip accuracy. When the test pattern is detected, the local PN code generator is reset to the state immediately after its longest run then allowed to progress through its sequence.

After the PN code generator is reset, the received SS signal and the local PN code sequence are multiplied and low pass filtered to a bandwidth equal to the original data

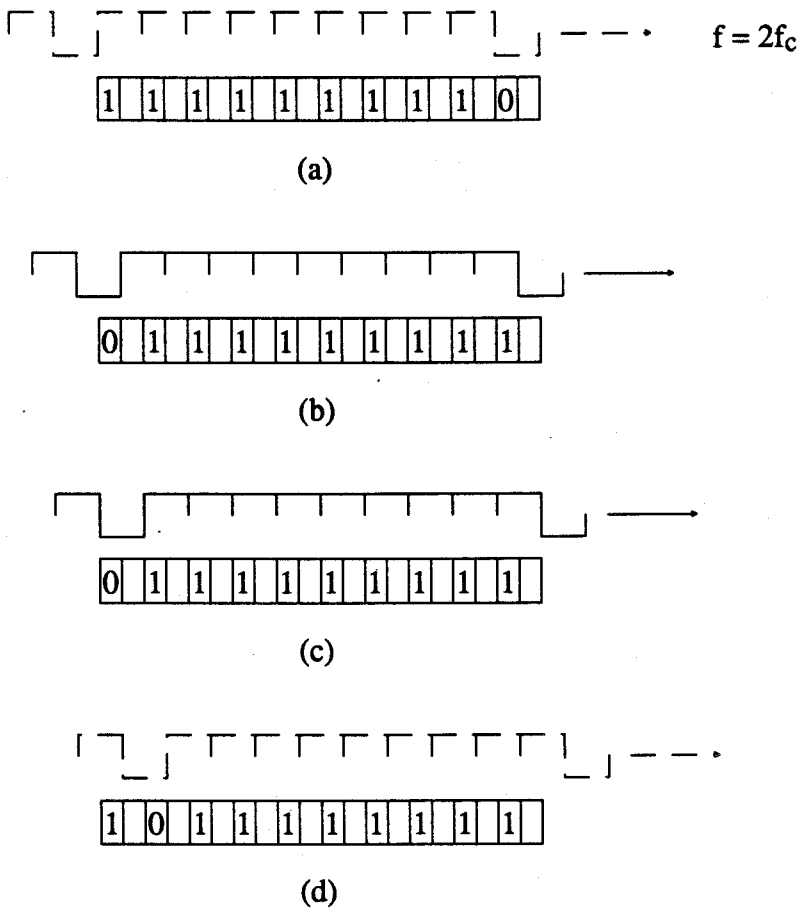


Fig. 5.2 Pattern Searching with 20-Bit Shift Register.

bandwidth. The result is brought into an energy detector which consists of a square law device and an integrate-and-dump filter. The filter output is compared with a decision threshold, sampled, and cleared at a regular dwell time interval.

The serial search scheme shown in Fig. 4.2 is now replaced by the RAATS shown in Fig. 5.3. Since the search mode is substituted by the transition sequence estimation (TSE), dwell time T_{d1} is eliminated. The entire SVL strategy is then simplified to have just one stage instead of q stages. However, the basic components that form the SVL strategy remain unchanged. When the test pattern is identified, the local PN code generator is reset and the

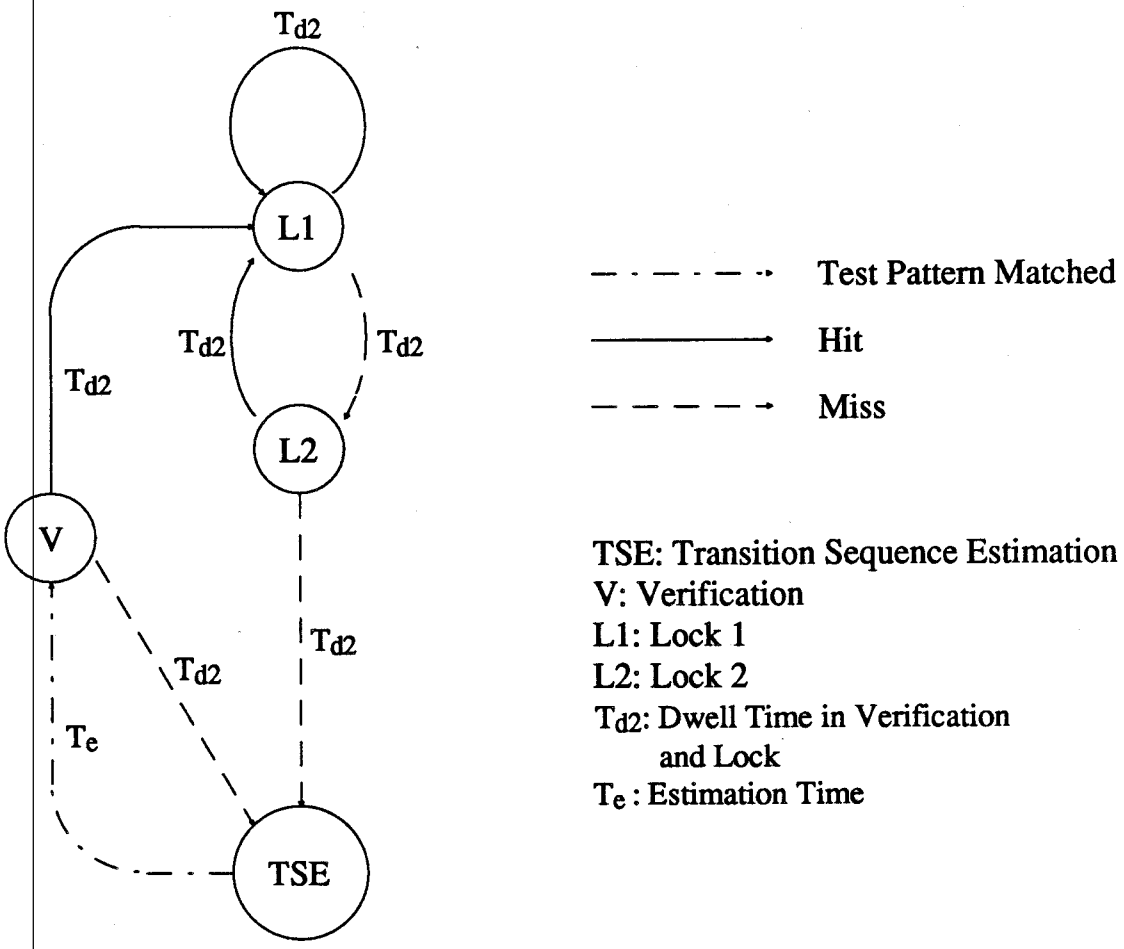


Fig. 5.3 SVL Strategy in RAATS Method

SVL strategy enters the verification mode. The estimation time T_e varies with the received SNR and is at least 10 or more epoch times. After a dwell time of T_{d2}, the integrate-and-dump filter output is compared with the detection threshold V_{th2}. If the output exceeds the threshold, the SVL strategy enters the lock 1 mode. On the other hand, if the integrate-and-dump filter output falls below V_{th2}, the SVL strategy clears the accumulator and restarts the accumulation of the transition sequence and waits for another code phase estimate. Upon entering the L1 mode, the operation of the SVL strategy remains the same

as in the serial search scheme. Two successive misses will put the SVL strategy back to search mode again where another estimate of code phase will be obtained.

5.3. Hardware Implementation

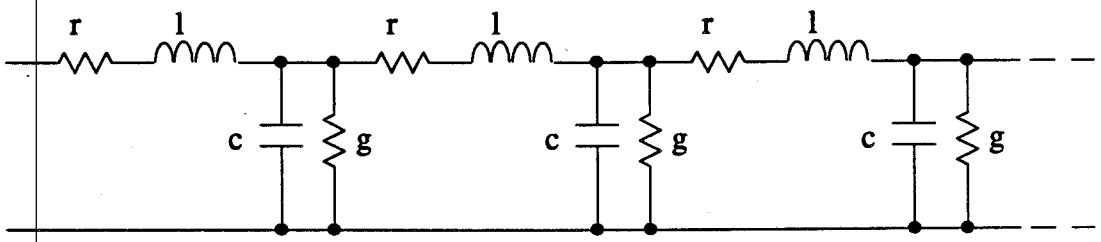
5.3.1. Transition Estimator

The transition estimator consists of two components, a one-chip-time delay line and an analog multiplier. A number of approaches were considered for delaying the received SS signal by exactly one chip time (13 μs). A series of Charge Coupled Devices (CCD) were first considered. However, with the length and maximum sampling frequency limitations of the CCD available, the delay time could not be made short enough with the available devices. Coaxial transmission line and fibre optic cable were considered but these ideas were not used because of the long physical length required. A lumped element transmission line model was suggested and finally used. A model is available which represented 26 gauge twisted pair telephone cable. Sections of the transmission line model are illustrated in Fig. 5.4(a). Since attenuation is not desirable, the series resistance r in all sections are shorted and the final line model used is shown in Fig. 5.4(b). The characteristic impedance of the line model is calculated as

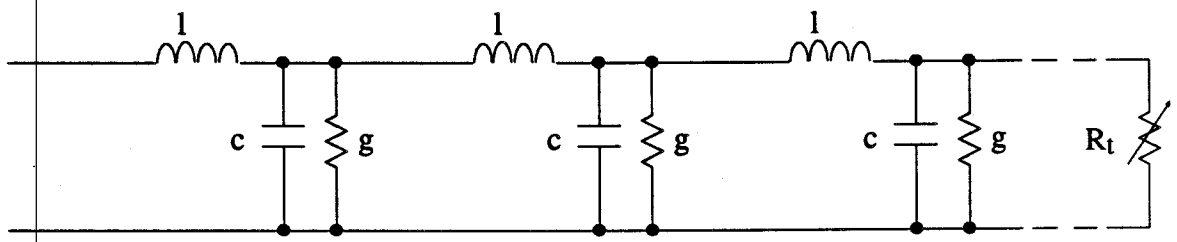
$$Z_0 = \sqrt{\frac{r + j\omega l}{g + j\omega c}} \qquad \Omega \qquad (5.1)$$

where r is the series resistance, l is the series inductance, g is the shunt conductance and c is the shunt capacitance. By setting r=0 Ω/unit length, l=100 μH/unit length g=1/54 x 10^6

S/unit length and c=83 nF/unit length, Z_0 becomes 107 Ω. The line model is terminated with a potentiometer which is adjusted to match the characteristic impedance and to prevent reflection. With a propagation delay of approximately 5 ns/meter, 2.6 km (or 1.56 miles) of equivalent cable is required.



(a)



(b)

r = Series Resistance = 18 Ω/unit length
l = Series Inductance = 100 mH/unit length
c = Shunt Capacitance = 83 nF/unit length
g = Shunt Conductance = $1/54 \times 10^6$ s/unit length
R_t = Termination Impedance

Fig. 5.4 Transmission Line (Delay Line) Model

A block diagram of the transition estimator is shown in Fig. 5.5. In order to remove the unnecessary high frequency noise, the received signal is low pass filtered with a bandwidth equal to the chip rate. Since the lumped element transmission line model also band limits the signal, low pass filtering is not applied to the delayed version of the received signal. An analog multiplier (AD532) is used to multiply the received SS signal and its delayed version. Since the multiplier has a gain of 1/10, both input signals are amplified to the level of ±2.25 V before the multiplication in order to maintain sufficient output signal level to operate the CCD. The product is the estimation of the transition of the received sequence and is then sent to the transition sequence accumulator.

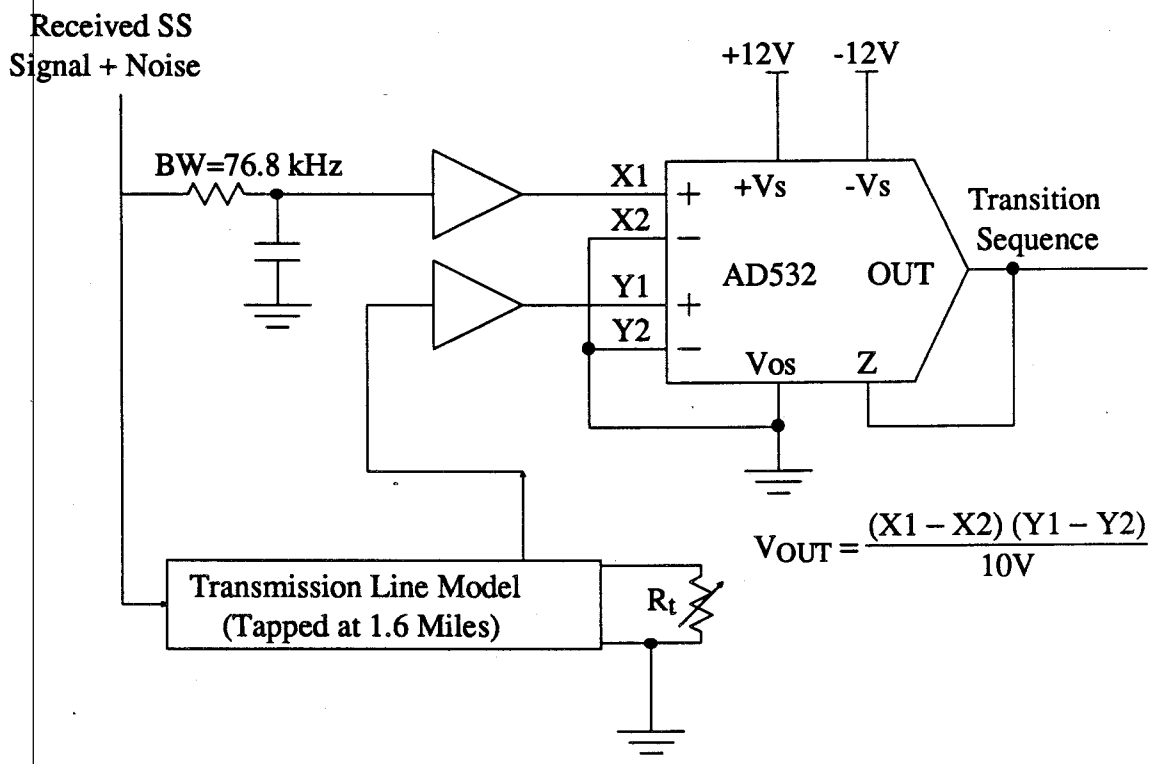


Fig. 5.5 Block Diagram of Transition Estimator

Actual waveforms from the transition estimator are plotted in Fig. 5.6 for the no noise condition. The chip rate is 76.8 kchips/s. Trace 1 is the received sequence. Trace 2 is the delayed received sequence. Trace 3 is the output of the analog multiplier, i.e., the instantaneous transition sequence. Trace 4 is the local PN code generator reset signal resulted from the test pattern detection. The widest pulse in traces 1 and 2 represents the longest run of the PN sequence (10 chips). The product of these segments is marked on trace 3. Note that the pulse on trace 4 was resulted from a match with the pattern 0111111111.

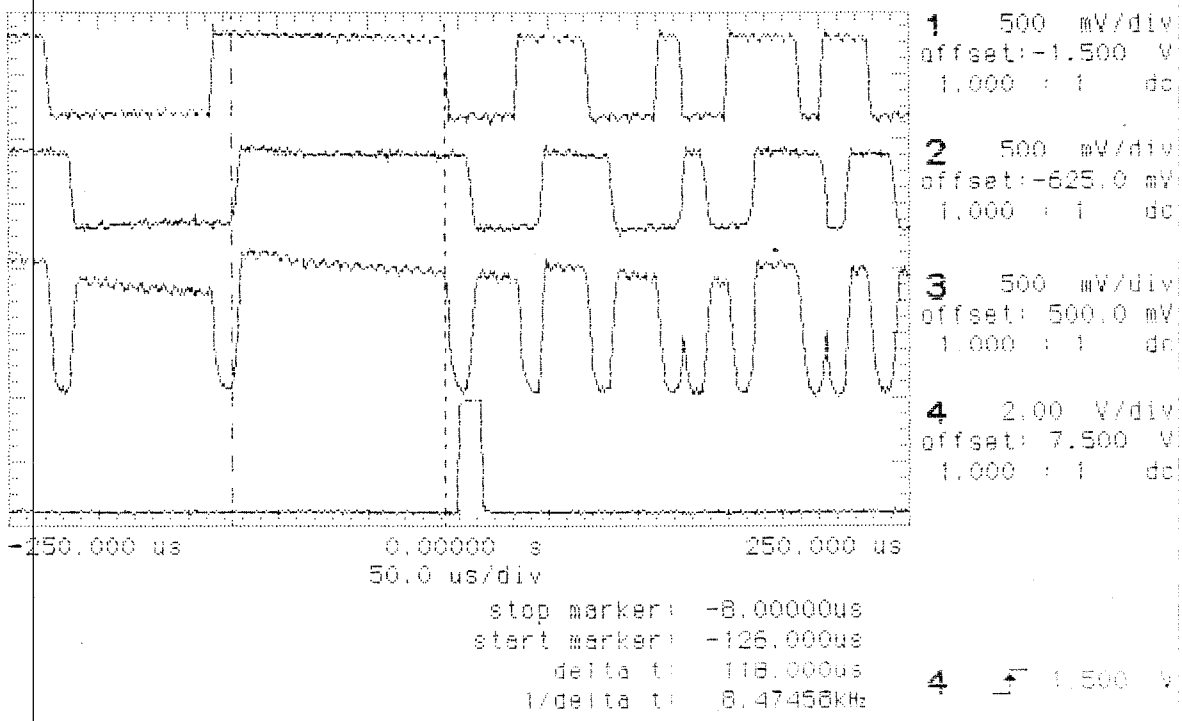


Fig. 5.6 Plot of Transition Sequence

5.3.2. Transition Sequence Accumulator

After the transition information of the received SS signal is obtained, it is brought into a transition sequence accumulator. This "leaky" accumulator adds the new transition information to the previous transition information which has been slightly attenuated ("leakage"). The structure of the accumulator is shown in Fig. 5.7. To obtain sufficient averaging effect, the forward input gain, G_{fw}, is set to approximately 0.1, and the total loop gain, G_l, is set to approximately 0.9. The loop gain must be less than 1.0 to prevent oscillation in the loop. The accumulator consists of an adder which adds the incoming signal with the feedback signal. A series of Charge Coupled Devices (CCD) are used to form an analog delay line. Hence G_l is the net gain of the CCD and the feedback gain, G_{fb}. Since ±1/2 chip alignment or better is required for synchronization, two samples are taken from

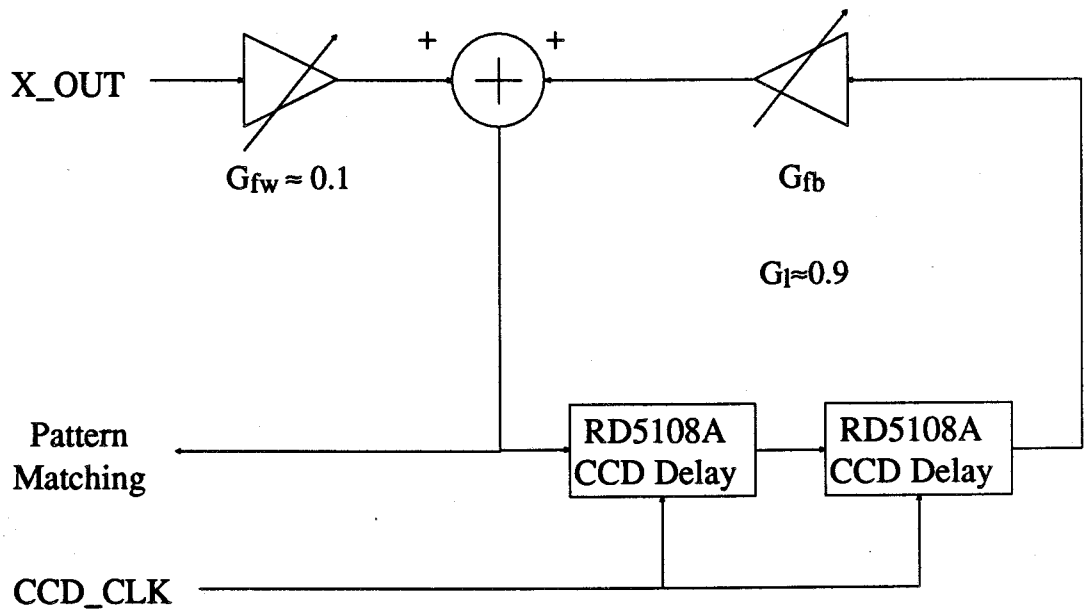


Fig. 5.7 Transition Accumulator Structure

each chip. This is implemented by making the number of cells in the analog delay line equal to twice the PN code length.

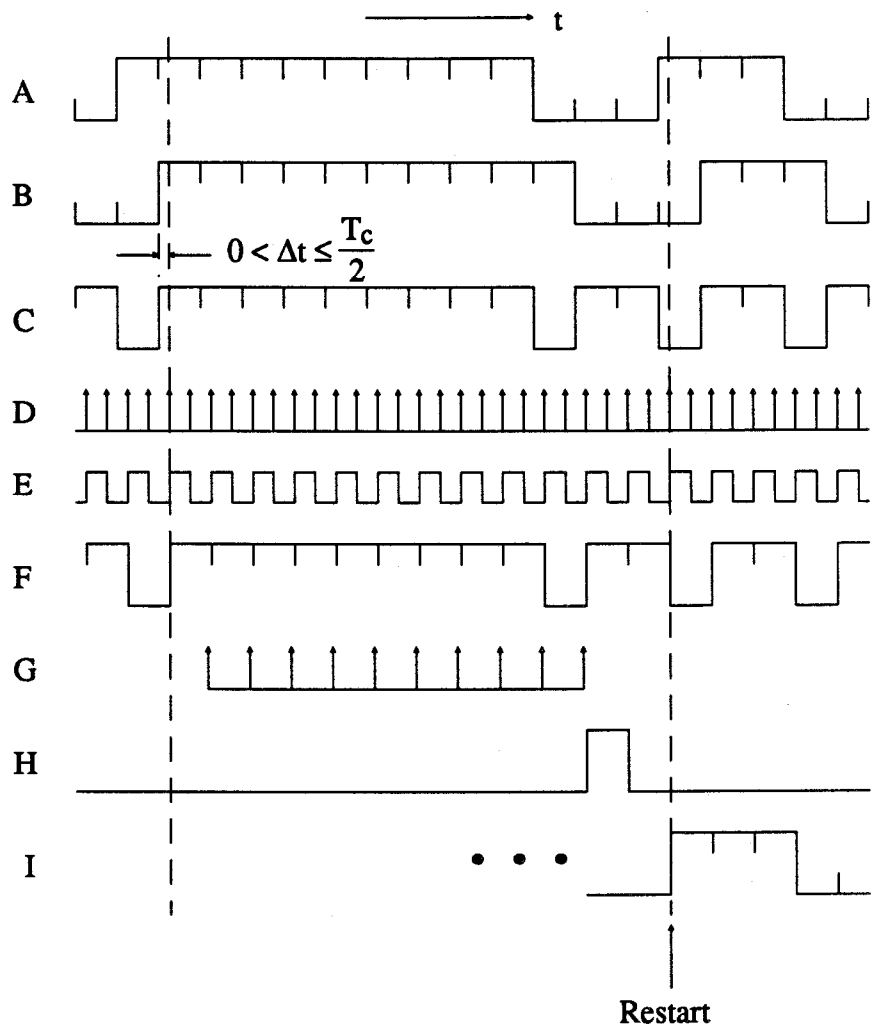
For the experimental hardware, the EG & G Reticon RD5108A [32] is selected to provide a 1024 sample analog delay line. By cascading two packages in series, a 2048 sample delay line is created. Since the CCD has a gain greater than one, G_{fb} has to be less than 0.9 in order to maintain the desired loop gain. For a 1023 chip PN sequence, a 2046 sample delay line is required. Since there are two extra cells in the line, two "dummy" samples are stuffed to fill the spaces. In order to maintain a cycling rate of one PN code period, these two samples are taken with a fast clock signal. Details on the sampling clock is addressed in the next section.

As mentioned in Section 5.2, any detour or false alarm will cause the synchronizer to clear the transition accumulator and restart the accumulation again. With an accumulation time constant of approximately 10 epoch, re-accumulation might increase the acquisition time significantly, especially at low SNR where detour or false alarm occurs more frequently. It was then decided not to reset the transition accumulator. Whenever the local PN sequence fails to correlate with the received sequence, a new code phase estimate will be made on the accumulating transition sequence until the correct phase is found. The disadvantage of this suboptimal implementation is that errors could be accumulated in the sequence and false detection could occur again.

The 20-bit shift register used for the pattern matching is shortened to 10-bit in the implementation and is clocked by the PN code generator clock signal at 76.8 kHz. Since

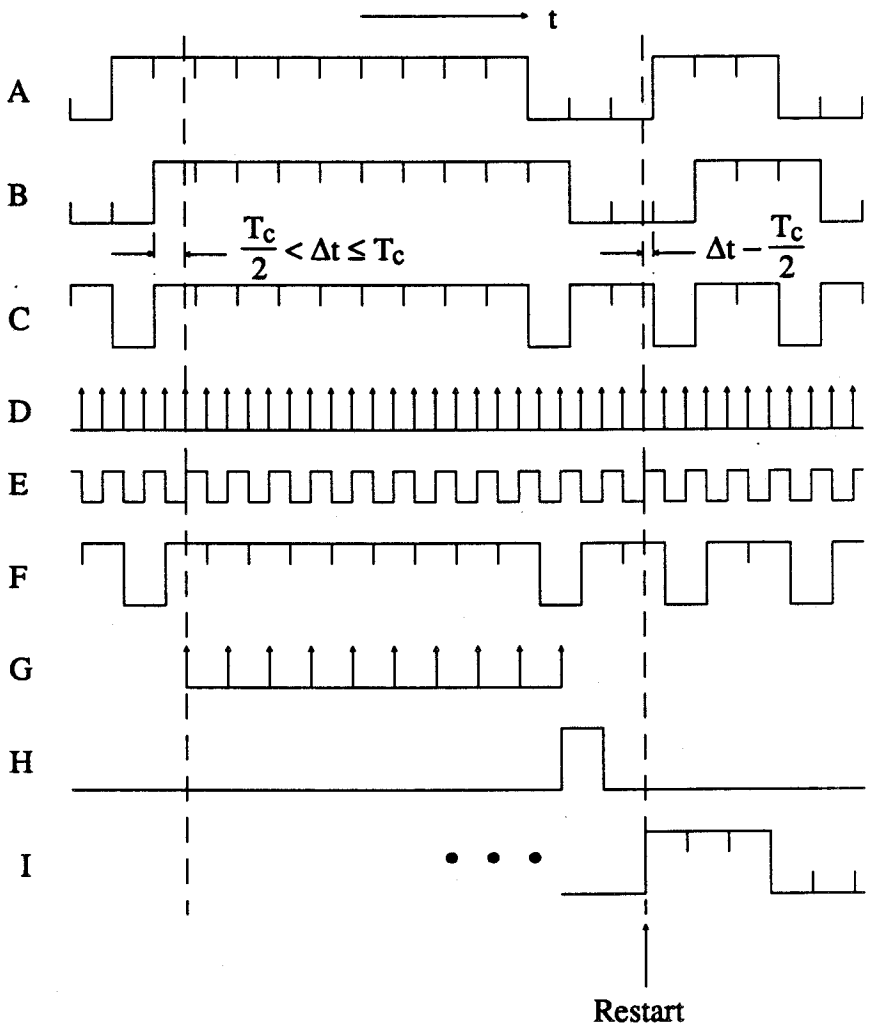
the transmitter clock generator is started randomly during tests, the sampling of the CCD accumulator will not likely be phase synchronized with the received sequence. Since two samples are taken from each chip of the received transition sequence, the first one can be sampled anywhere within the first half of a chip while the second one will be sampled at exactly $\frac{T_c}{2}$ later in the second half of the chip as illustrated in Fig. 5.8. The time delay between the received transition sequence and the receiver PN code generator clock is denoted as Δt. When $0 < \Delta t \leq \frac{T_c}{2}$, the accumulated transition sequence will be delayed by Δt as shown in Fig. 5.8 (a). Since pattern matching occurs in synchronism with the PN code generator clock, the receiver PN sequence after being reset will lag the transmitter PN sequence by Δt. When $\frac{T_c}{2} < \Delta t \leq T_c$, the accumulated transition sequence will be delayed by $\Delta t - \frac{T_c}{2}$ as shown in Fig. 5.8 (b). The receiver PN sequence after being reset will lead the transmitter PN sequence by $\Delta t - \frac{T_c}{2}$. Either case, the resulting PN sequence is within $\pm 1/2$ chip alignment with the transmitter PN sequence. Since the reset signal has duration of T_c and the PN code generator will not restart until the pulse is dismissed, the receiver PN sequence is reset to precisely two chips after the longest run and then allows to progress through its sequence.

In reality, the received and the delayed SS signal are bandlimited, hence the high frequency components in the received transition sequence will be attenuated and the waveform will be rolled off. The accumulated transition sequence will thus have step



A: Received SS Signal
B: Delayed SS Signal
C: Received Transition Sequence
D: Transition Accumulator Sampling Location
E: Receiver PN Code Generator Clock
F: Accumulated Transition Sequence (Clocked)
G: Pattern Matching Reference
H: PN Code Generator Reset Signal
I: Receiver PN Sequence

Fig. 5.8 (a) Idealized Pattern Matching Using 10-Bit Shift Register



A: Received SS Signal
B: Delayed SS Signal
C: Received Transition Sequence
D: Transition Accumulator Sampling Location
E: Receiver PN Code Generator Clock
F: Accumulated Transition Sequence (Clocked)
G: Pattern Matching Reference
H: PN Code Generator Reset Signal
I: Receiver PN Sequence

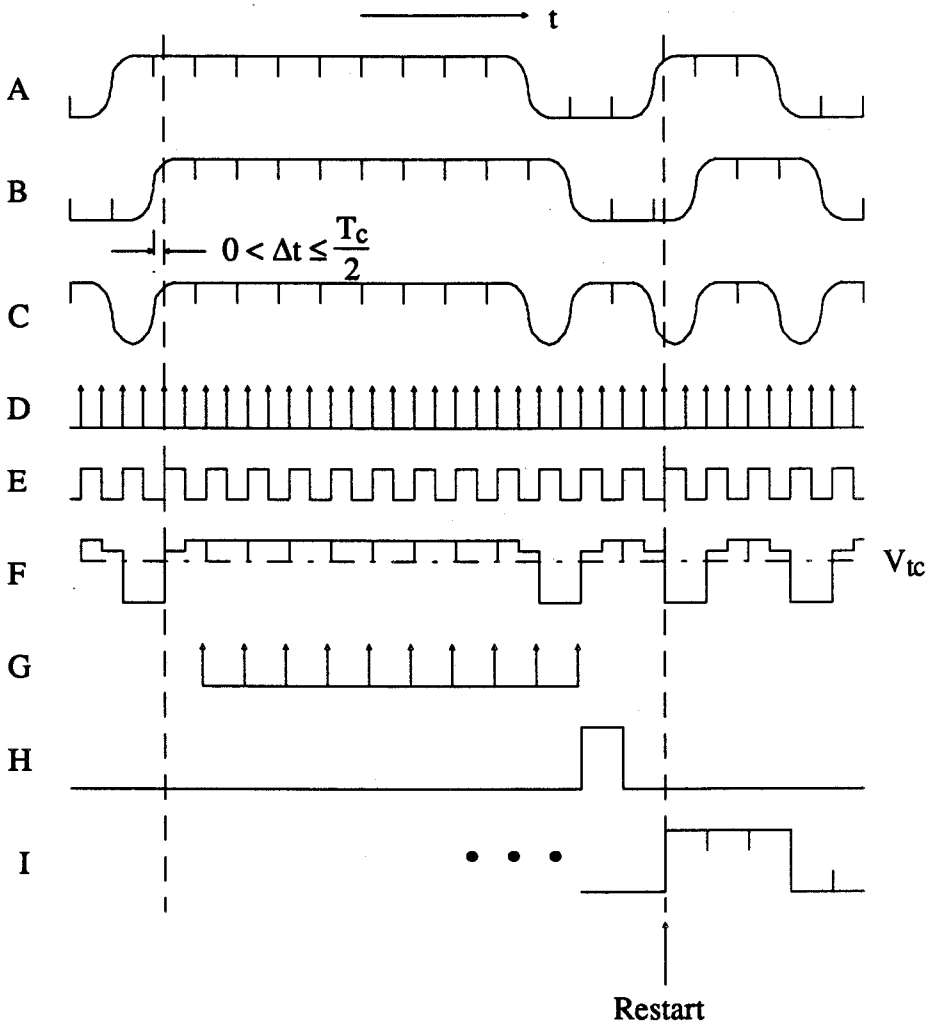
Fig. 5.8 (b) Idealized Pattern Matching Using 10-Bit Shift Register

voltages in a form approximately following the shape of the received transition sequence as shown in Fig. 5.9. The location of the PN code reset signal now depends on the detection threshold V_{tc} which in the case shown in Fig. 5.9 should be set near the peak voltage of the accumulated transition sequence.

5.3.3. Timing

As mentioned in Section 5.3.2, the transition sequence accumulator has 2048 CCD cells, but the number of transition samples required is 2046. In order to achieve a cycling rate of one PN code cycle or epoch, three CCD cells are used to store one of the samples. The additional two cells are referred to as "stuffed" samples. The stuffing process is illustrated in Fig. 5.10. The transition sequence is numbered from 1 to 1023 and repeats periodically. Each of these transition chips is sampled twice and the samples are numbered from 1 to 2046. Stuffing occurs after the 2046^{th} sample is taken. To maintain the alignment between the transition sequence and the samples, sampling rate for samples 2046 to 2048 are tripled. As shown in Fig. 5.10, samples 1 to 2045 has a sampling rate of $\frac{T_c}{2}$ whereas samples 2046 to 2048 has a sampling rate of $\frac{T_c/2}{3}$. Only samples 1 to 2046 contains transition information which will be used in the test pattern matching.

The CCD delay line requires two clock cycles per sample. Hence to sample, hold and shift 2048 samples through 2048 CCD cells, 4096 clock pulses are required. As shown



A: Received SS Signal
B: Delayed SS Signal
C: Received Transition Sequence
D: Transition Accumulator Sampling Location
E: Receiver PN Code Generator Clock
F: Accumulated Transition Sequence (Clocked)
G: Pattern Matching Reference
H: PN Code Generator Reset Signal
I: Receiver PN Sequence

Fig. 5.9 Bandlimited Pattern Matching Using 10-Bit Shift Register

in Fig. 5.10, the clock period for samples 1 to 2045 is $\frac{T_c}{4}$ whereas the clock period for samples 2046 to 2048 is $\frac{T_c}{12}$.

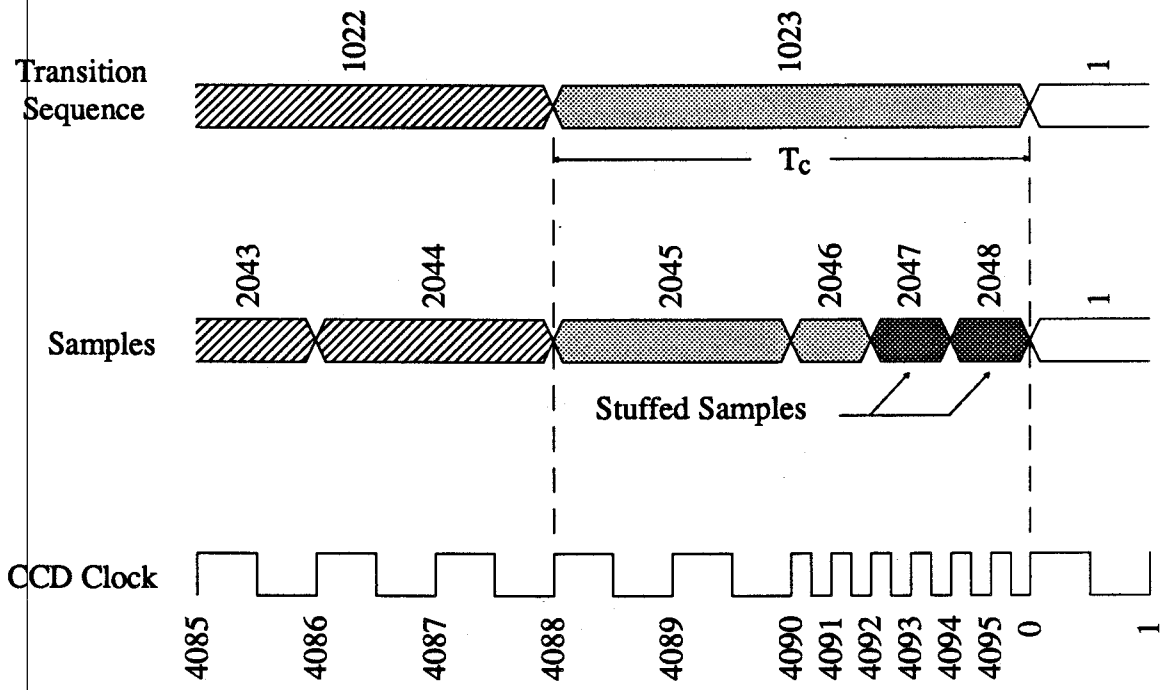


Fig. 5.10 Accumulator Clocking and Sample Stuffing

According to manufacturer specifications [32], the maximum clock rate of the CCD is 1.6 MHz, thus the maximum chip rate which can be used is $\frac{1.6\text{ MHz}}{12} = 133.33$ kchips/s. Since the processing gain was set to 32, the maximum data rate would be $\frac{133.33\text{ kHz}}{32} = 4.167$ kbits/s. For testing purposes, the data rate was set to the standard rate of 2400 bps. Hence T_c became 13.02 μs, the sampling rate for samples 1 to 2045 became 307.22 kHz, and the sampling rate for samples 2046 to 2048 became 921.6 kHz.

The block diagram for the RAATS timing circuitry is shown in Fig. 5.11. The system clock labelled SYS_CLK is set to 921.6 kHz. The divide-by-3 counter generates the lower sampling rate for samples 1 to 2045. It is multiplexed with the system clock to form the CCD clock. The CCD clock is brought into a 12-bit binary counter. When the count is below 4090 (see Fig. 5.10), the decoder selects the low sampling rate. When the count is higher than or equal to 4090, the decoder selects the high sampling rate. Furthermore, the divide-by-4 counter generates the 76.8 kHz clock for the PN code generator.

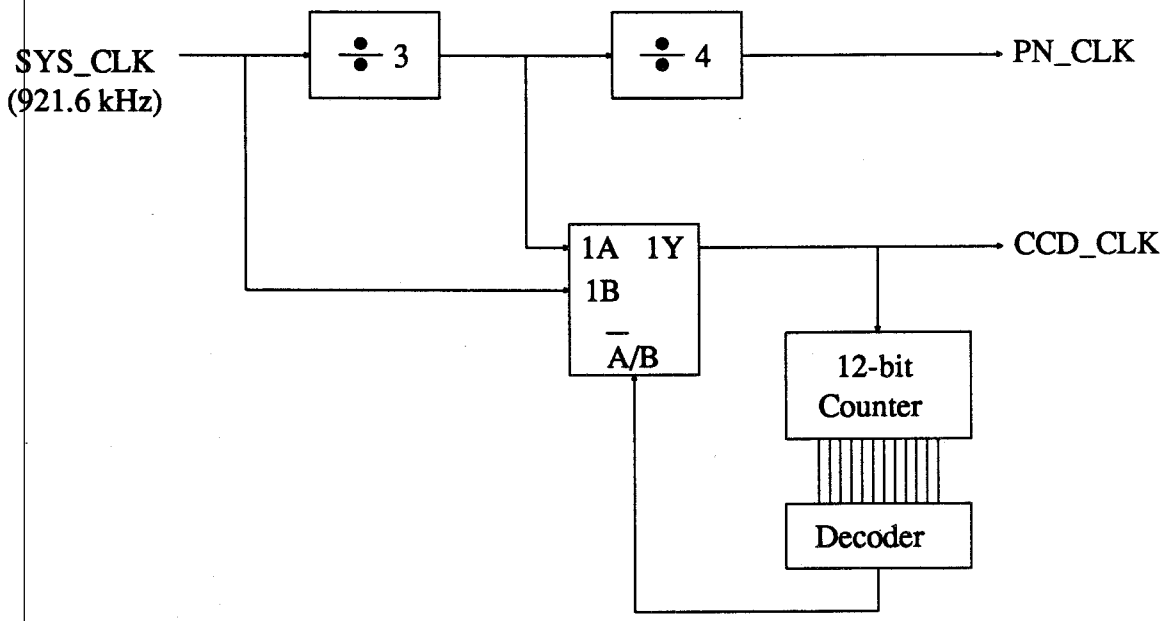


Fig. 5.11 Block Diagram of RAATS Timing Control

5.3.4. SVL Controller

Since the SVL strategy used in the RAATS system is similar to the one used in serial search, only slight modification is required in order to make use of the available hardware from the serial search system. The state flow diagram of the SVL shown in Fig. 5.3 has

been implemented using a 2-bit shift register. The state of the SVL is determined by decoding the 2-bit number.

Fig. 5.12 illustrates the modified serial search SVL controller which becomes the RAATS SVL controller. Only two shift register stages are required. The shift register clock is wired directly to TD2_TIME_OUT. The decoder output has only four values which corresponds to the four possible states in the SVL strategy. Circuits for the correlator (integrate-and-dump filter and energy detector), dwell time controller and detection threshold controller are also taken from the serial search system. Only T_{d2} and V_{th2} are used; T_{d1} is not required.

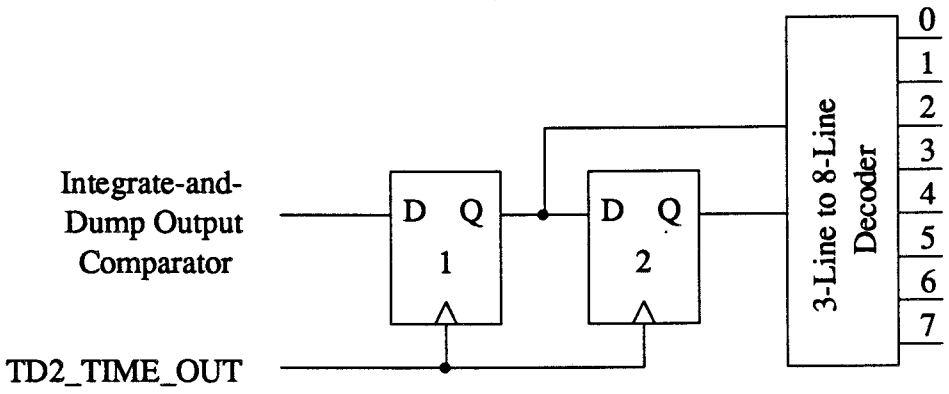


Fig. 5.12 RAATS SVL Controller

Chapter 6

TEST RESULTS

Two methods were considered for measuring the acquisition time performance of the serial search and the RAATS acquisition systems. The first method was to count and record the number of clock pulses from the arrival of the first chip to the point where the SVL strategy enters lock mode. The second method was to use a microcomputer to control the start and reset operations of the spread spectrum system, calculate the acquisition time, and record the results. The second method was chosen because it is faster and the maintenance of the lock signal can be tested by proper algorithm.

Several candidates were considered for the microcomputer controller and data acquisition. An IBM (or compatible) personal computer was considered where the spread spectrum system could be controlled via a RS-232 interface of the computer. A Hewlett Packard (HP) 9000 series computer was also considered. In this case, the circuit boards could be controlled via a General Purpose Interface Bus (GPIB) interface. Both of these methods require interface circuits and possible software development. It was finally decided to use a Commodore CBM microcomputer, Model 8032, for the following reasons. First, the CBM microcomputers have an internal real-time clock timer which can be programmed easily by BASIC. Second, they are equipped with a parallel user port which

for output purposes. The most significant bit (PB7) was configured for input only. Port PB0 and PB1 were used to start the demodulator and the modulator, respectively, while port

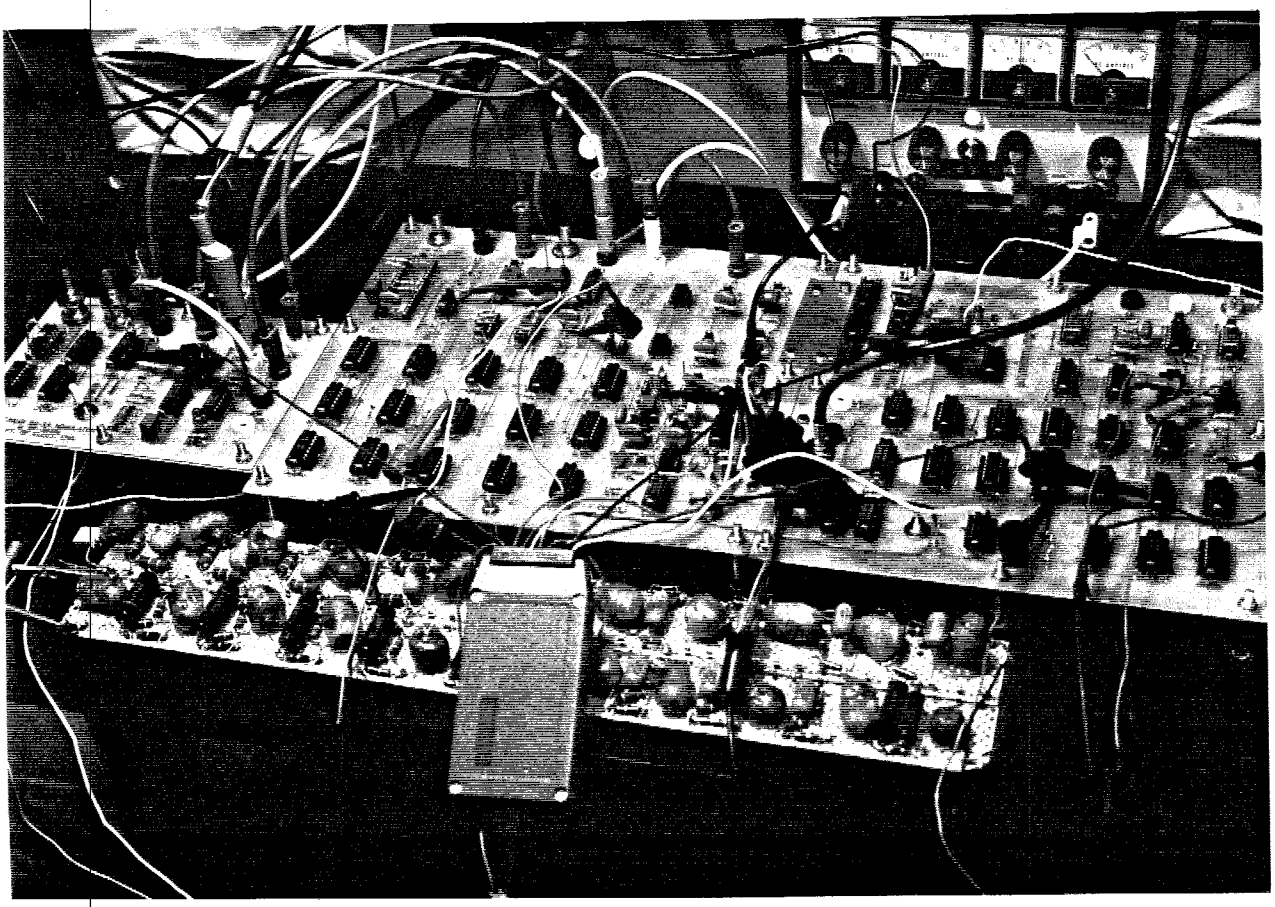


Fig. 6.3 Photograph of Circuit Board Prototypes

Fig. 6.4 Protograph of Test Setup

correlator, detection threshold controller, dwell time controller and SVL controller from the serial search were connected to the RAATS instead. Thus the two boards were interconnected.

6.2. Test Software

Two BASIC programs were written for this project. The first program, called ACQT, performs the timing and recording of acquisition time. ACQT makes use of the real-time clock of the CBM microcomputer. The computer keeps track of time in "jiffies" which is equivalent to 1/60 second [34]. TIME, or TI, is a reserved numeric variable which is automatically incremented every 1/60 second. TIME is initialized to zero on power up and is reset back to zero after 5,183,999 jiffies, i.e., 24 hours. Thus acquisition time can be calculated as follows:

(1) Assign TIME (or TI) to a reserved variable at the beginning of each acquisition time measurement.

(2) Subtract the stored TIME variable from the time the SVL enters lock mode. This will give the amount of jiffy time it took to locate the correct code phase. Time in terms of hours, minutes, and seconds can then be converted from jiffies accordingly.

ACQT has two user selected test modes: (1) threshold test, and (2) acquisition test. The flow chart of ACQT is shown in Fig. 6.5 while the source code is listed in Appendix D. When the program is first executed, a few questions are posed regarding the type of test, signal and noise levels, detection threshold voltages, and the automatic data saving rate. If

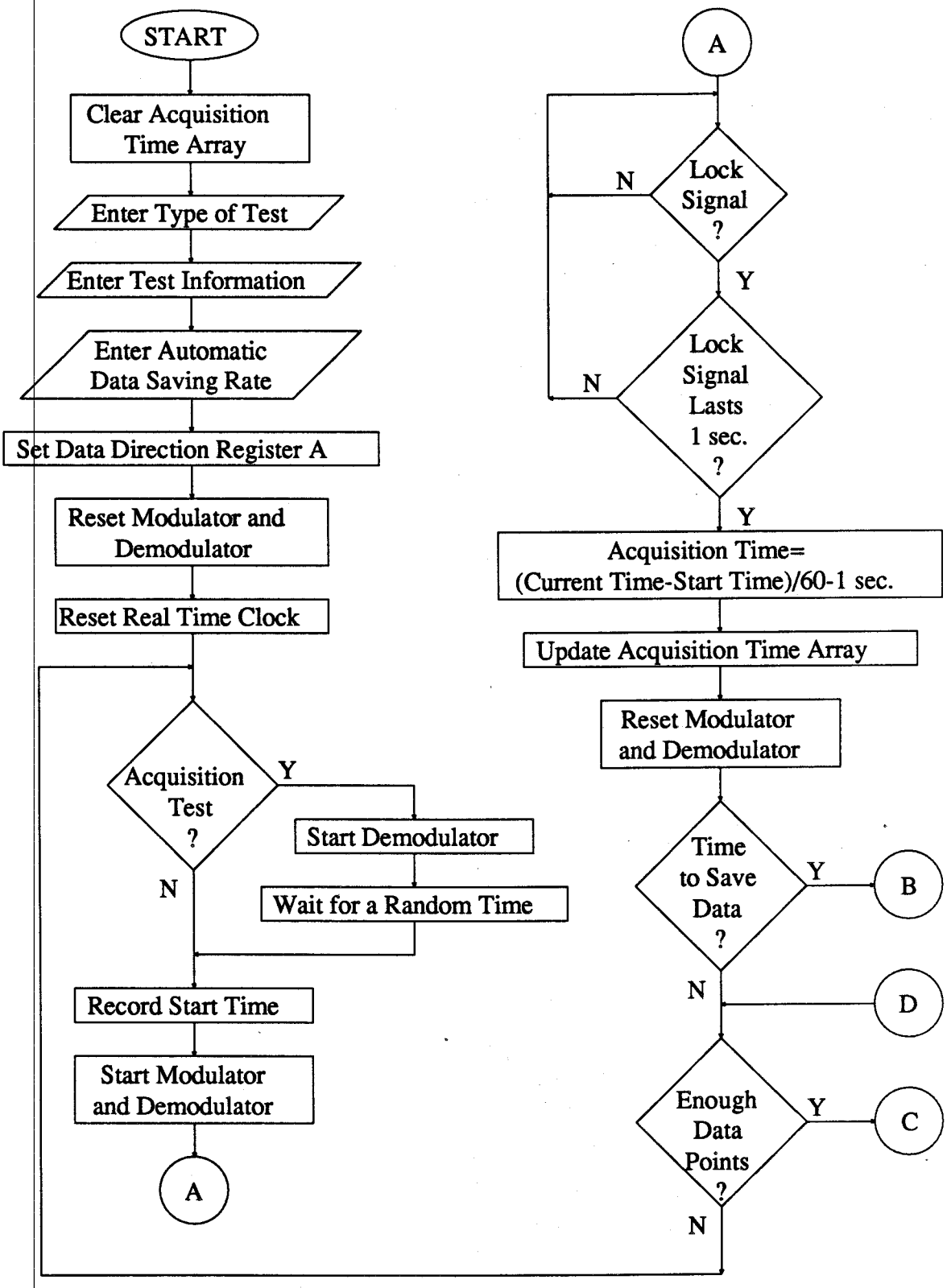


Fig. 6.5(a) Flow Chart of ACQT

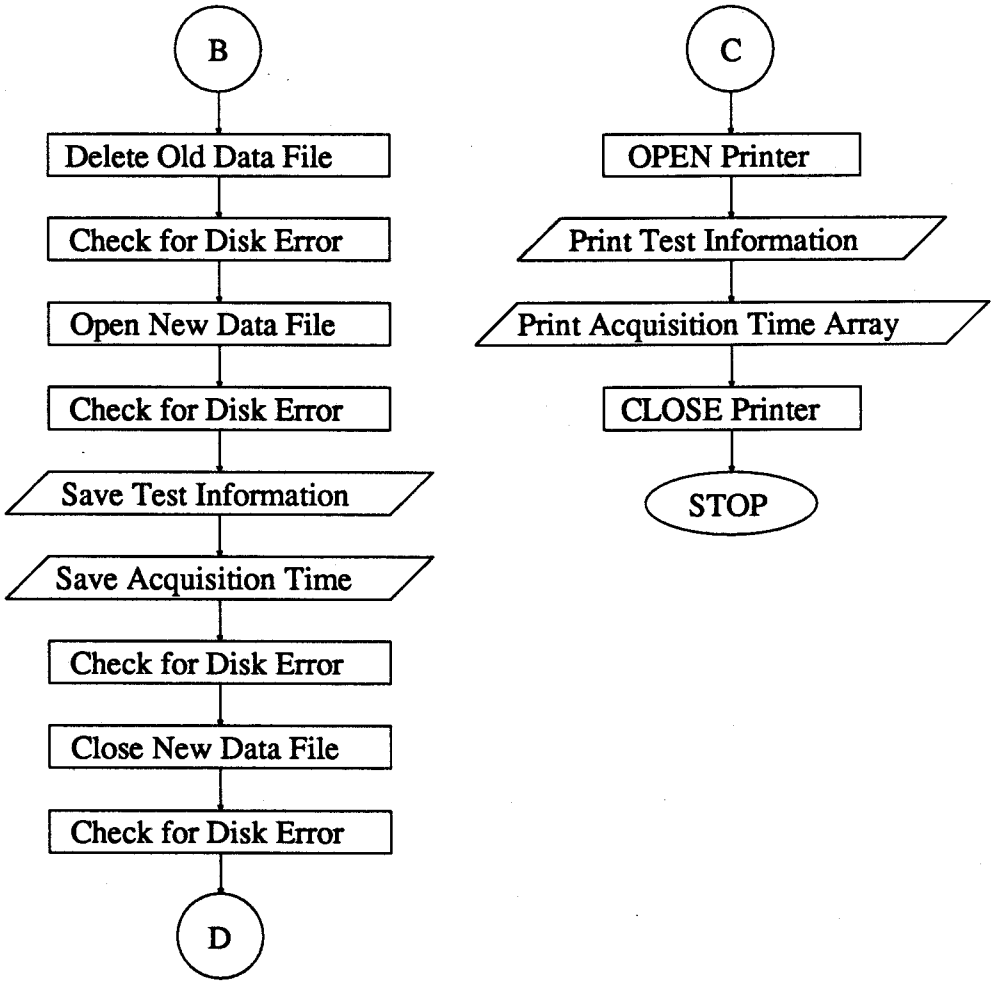


Fig. 6.5(b) Flow Chart of ACQT (continued)

threshold test is selected, both the modulator and the demodulator will be started simultaneously. If acquisition test is selected, on the other hand, the demodulator will be started first and the modulator will be started after a random time interval. This is used to simulate a random transmission time delay. Hence the correct code phase will be randomly located in the code phase uncertainly region. The lock signal is continuously monitored once the modulator has been started. When a lock signal is received on PB7, the program will immediately examine the maintenance time of the signal. If the lock signal maintains

for one second, correct code phase is considered to be found and acquisition time is calculated and displayed on screen. However, if the lock signal disappears within one second, a false alarm is assumed and the program will look again for the next lock signal.

Since the SVL strategy performs 60 tests/s in lock mode, the microcomputer should sample the lock signal at least 60 times during the one second of maintenance check in order to detect any false alarm. Unfortunately, experiments indicated that the computer can only process 29 samples/s. This is acceptable since it takes typically more than $\frac{1}{29}$ second to re-enter lock mode once lost lock occurred, thus any temporary break in the lock signal can be detected by the computer with a slight delay. In cases where the SVL strategy re-enters lock mode within $\frac{1}{29}$ second, the measured acquisition time will have an error of at most $\frac{1}{29}$ second which is negligible.

The measured acquisition times are separated with a bin size of 0.25 second. The results are accumulated and periodically stored in a sequential data file called TESTDATA. The automatic data saving rate is selected by the user at the beginning of the program. At the end, all test results are printed.

The second program, called RECALL, is a simple program which retrieves the acquisition time data stored on floppy disk by ACQT. The flow chart for RECALL is shown in Fig. 6.6 while the source code is listed in Appendix E. This program can be executed when the test is interrupted due to hardware problems or power failure and test results stored

before the termination are wanted. The program recalls the test information and test results from TESTDATA and display them on screen. Hardcopy can also be obtained.

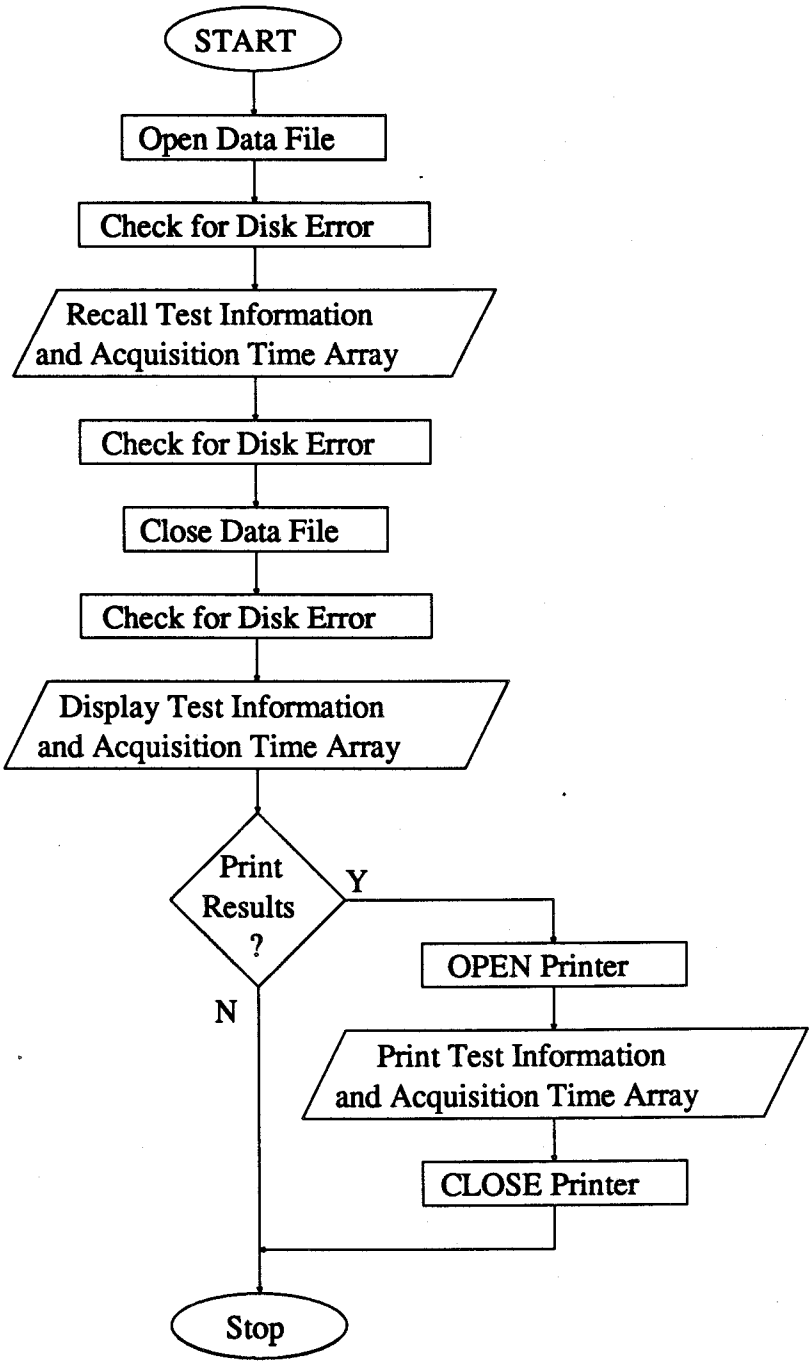


Fig. 6.6 Flow Chart of RECALL

6.3. Serial Search Test Results

6.3.1. Threshold Determination

As indicated in [16], serial search acquisition time performance is very sensitive to detection thresholds. Two detection threshold voltages have to be considered here: V_{th1} for dwell time T_{d1}, and V_{th2} for dwell time T_{d2}. In order to investigate the effect of threshold voltages and select an appropriate set for long term tests, threshold tests were conducted first. In these tests, both modulator and demodulator were started simultaneously. However, the modulator PN code generator was preset such that the sequence started at exactly one chip ahead of the demodulator PN sequence. This gave the worst case code phase separation (2044 cells apart). Hence a maximum average acquisition time would result.

A Hewlett Packard Digitizing Oscilloscope HP54501A was used to observe the integrate-and- dump filter output at different dwell time. With the display mode of the oscilloscope set at "env", the display reflects the minimum and maximum voltage in each horizontal position. Hence the amplitude variation of integration voltage can be captured as an envelope. Four such envelopes at -20 dB SNR are shown in Fig. 6.7 to 6.10. Each of the cases A, B, C, and D has a mean voltage with positive and negative deviations as recorded by the oscilloscope. This is described as voltage envelope in each case. Fig. 6.7 shows an integrate-and-dump filter output voltage envelope in search mode. This envelope is referred to as envelope A. The integration period is T_{d1} which is defined to be eight data bit times, i.e., $8 \times \frac{1}{2400} = 3.33$ ms. The voltage deviation of this envelope measured at 3.33 ms is

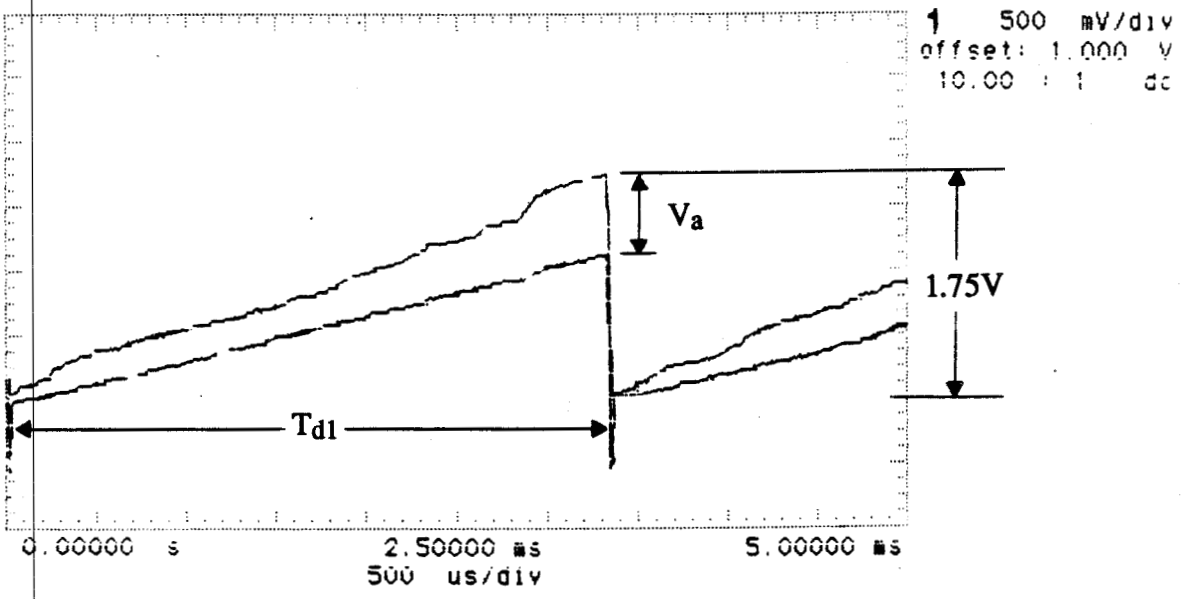


Fig. 6.7 Envelope A

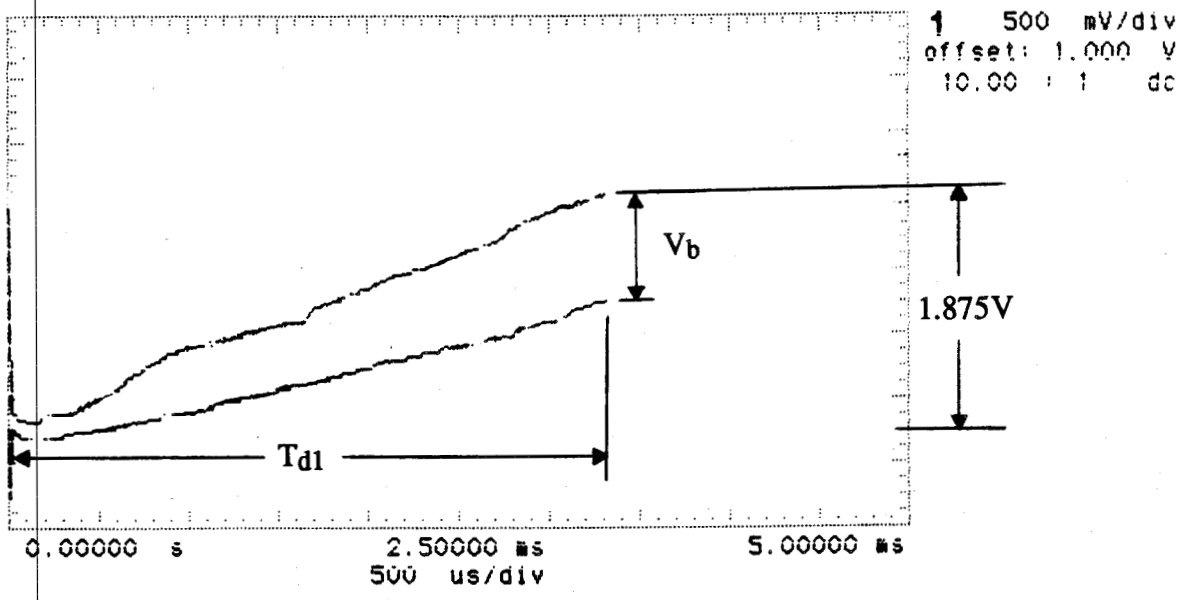


Fig. 6.8 Envelope B

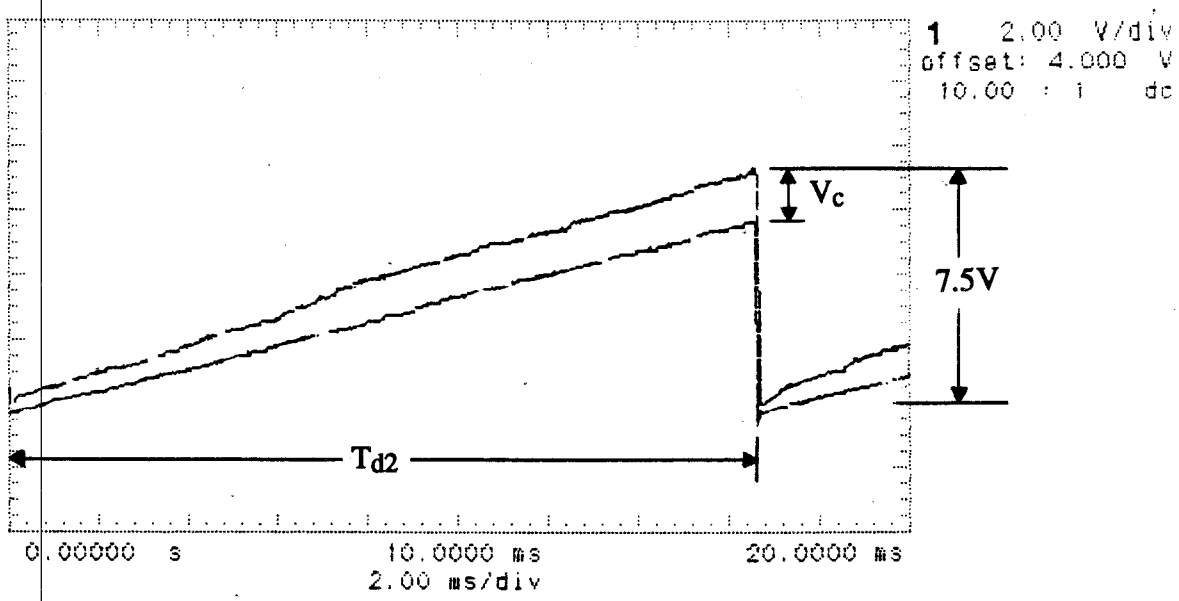


Fig. 6.9 Envelope C

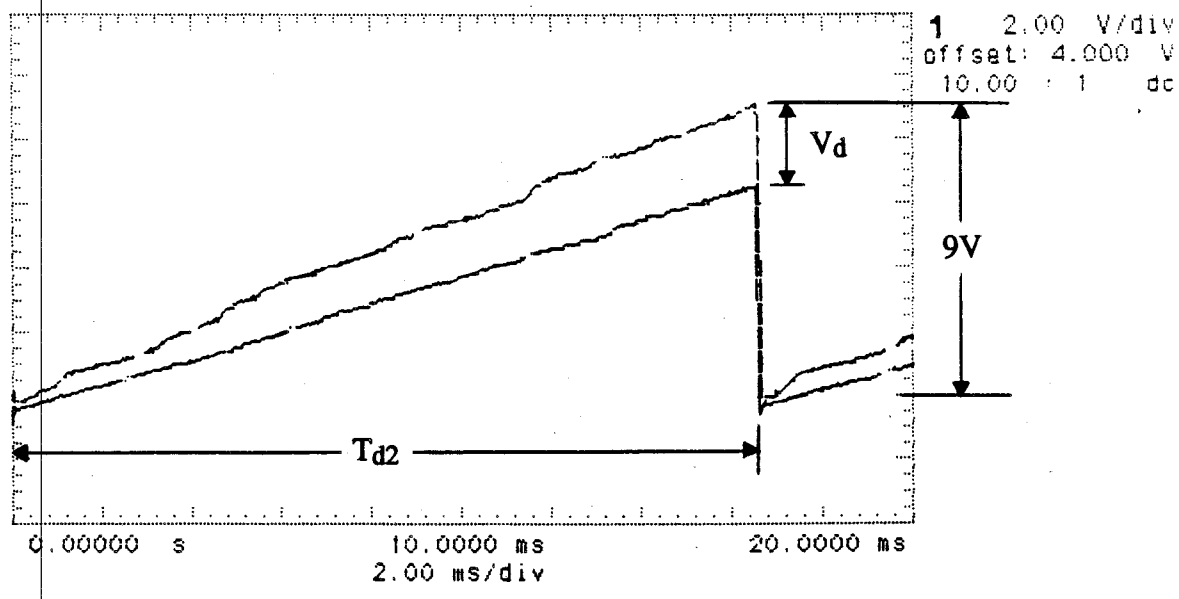


Fig. 6.10 Envelope D

referred to as V_a. Fig. 6.8 shows a truncated segment of a voltage envelope at which the SVL strategy is in lock mode and the code phase is within ±1/2 chip of the exact code phase. This envelope is referred to as envelope B and the voltage deviation measured at 3.33 ms is referred to as V_b. Fig. 6.9 is a voltage envelope at false lock which was created by reducing V_{th2}. This envelope is referred to as envelope C and the voltage deviation of this envelope measured at 16.65 ms is referred to as V_c. Note that the integration period is 16.65 ms which is equal to five times T_{d1}. Fig. 6.10 is a voltage envelope at which the SVL strategy is in lock mode and the code phase is within ±1/2 chip of the exact code phase. This envelope is referred to as envelope D and the voltage deviation of this envelope measured at 16.65 ms is referred to as V_d. Recall from Section 4.3.3, V_{th1} is set between V_a and V_b, and V_{th2} is set between V_c and V_d. Before starting the threshold tests, all four voltage envelopes were measured. Ten measurements were taken for each of the four envelopes and their mean voltage were calculated. These measurements are tabulated in Appendix F. For comparison purpose, a voltage envelope at which exact code phase is located and locked by the SVL strategy is shown in Fig. 6.11. Note that the amplitude of the envelope is higher than envelope D shown in Fig. 6.10. However, envelope D was used instead because of the ±1/2 chip alignment characteristic of a typical tracking loop.

For purpose of discussion, Gaussian noise was assumed and the voltage envelopes as recorded by the oscilloscope were assumed to be at deviations of $\pm 4\sigma$ about the mean. Hence the relationship between the probability density function (pdf) of V_a and the pdf of V_b can be illustrated as Fig. 6.12. Similar relationship can also be found between the pdf of V_c and the pdf of V_d. Detection threshold V_{th1} can be set at any point between the mean

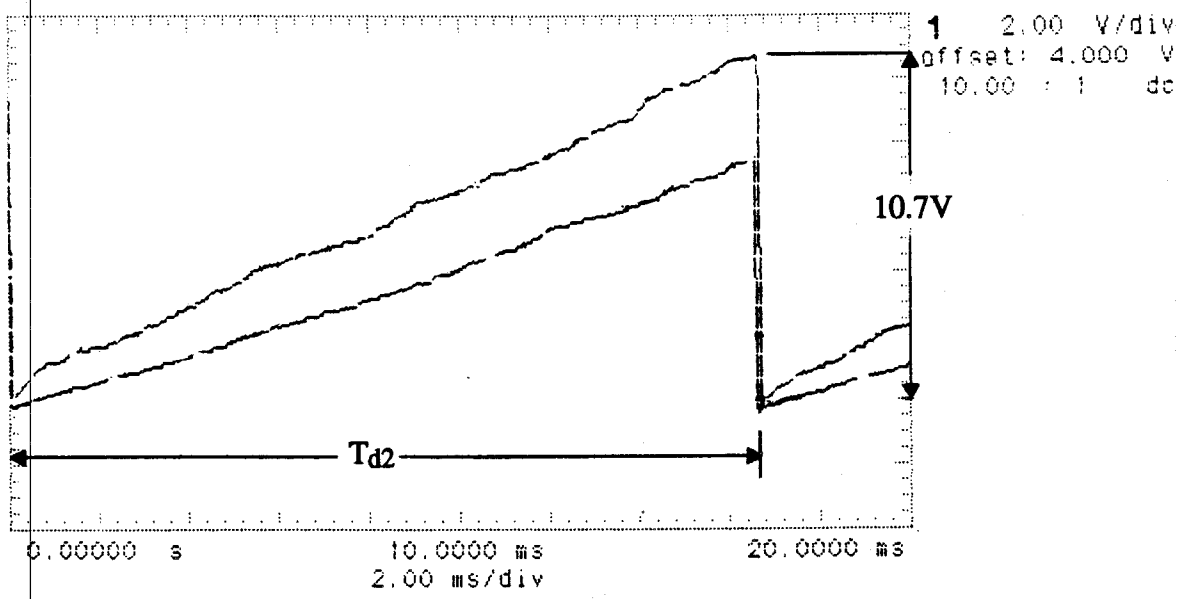
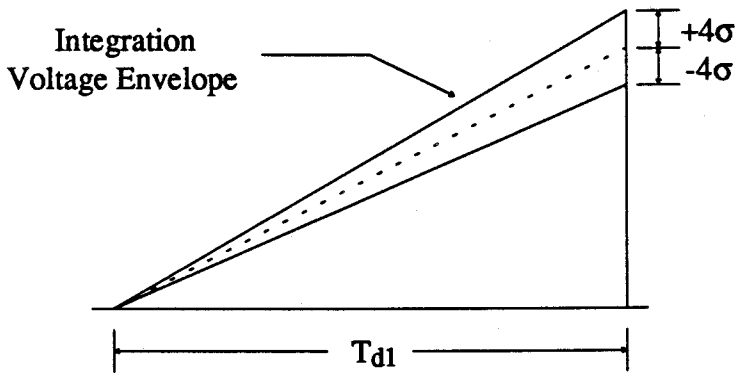


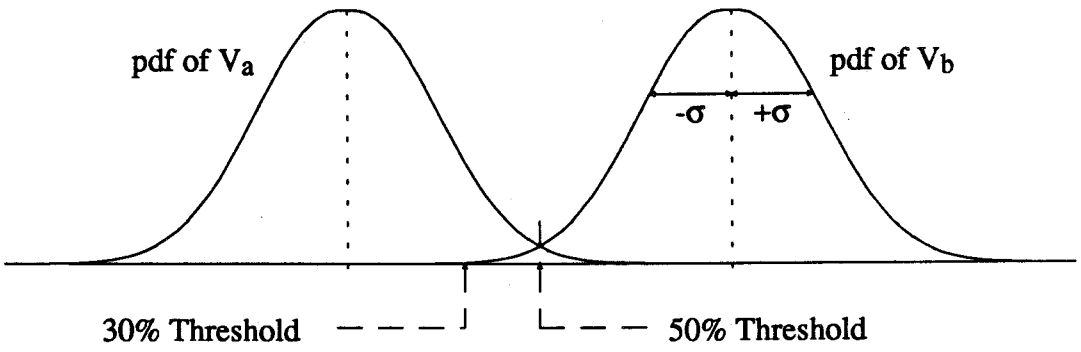
Fig. 6.11 Exact Synchronization Voltage Envelope

of the pdf of V_a and the mean of the pdf of V_b. Similarly, detection threshold V_{th2} can be set at any point between the mean of the pdf of V_c and the mean of the pdf of V_d. "x% threshold" was defined here as the voltage at x% between the mean of two pdf. In other words, the 50% threshold is the voltage at the midpoint between the mean of the two pdf and is typically considered as the optimum detection threshold.

Based on the measurements showed in Appendix F, different percentage thresholds were calculated. 30%, 40%, and 50% thresholds were tested in separate threshold tests. 1000 measurements were obtained for each set of thresholds. The results are shown in Appendix G. These results indicate that acquisition time distribution spreads wider as thresholds decrease. This is caused by the increase in the number of detours and false alarms.



(a)



(b)

Fig. 6.12 Detection Envelopes and Threshold

6.3.2. Acquisition Time Test Results

Acquisition tests were performed with 30% and 50% thresholds. These thresholds has shown a significant change in acquisition time in the threshold tests. A total of 10,000 measurements were taken for each set of thresholds. Test results are shown in Fig. 6.13 and 6.14 for 30% and 50% threshold, respectively. A dashed line is superimposed on top of each acquisition time distribution curve to emphasize the approximated shape of the curve.

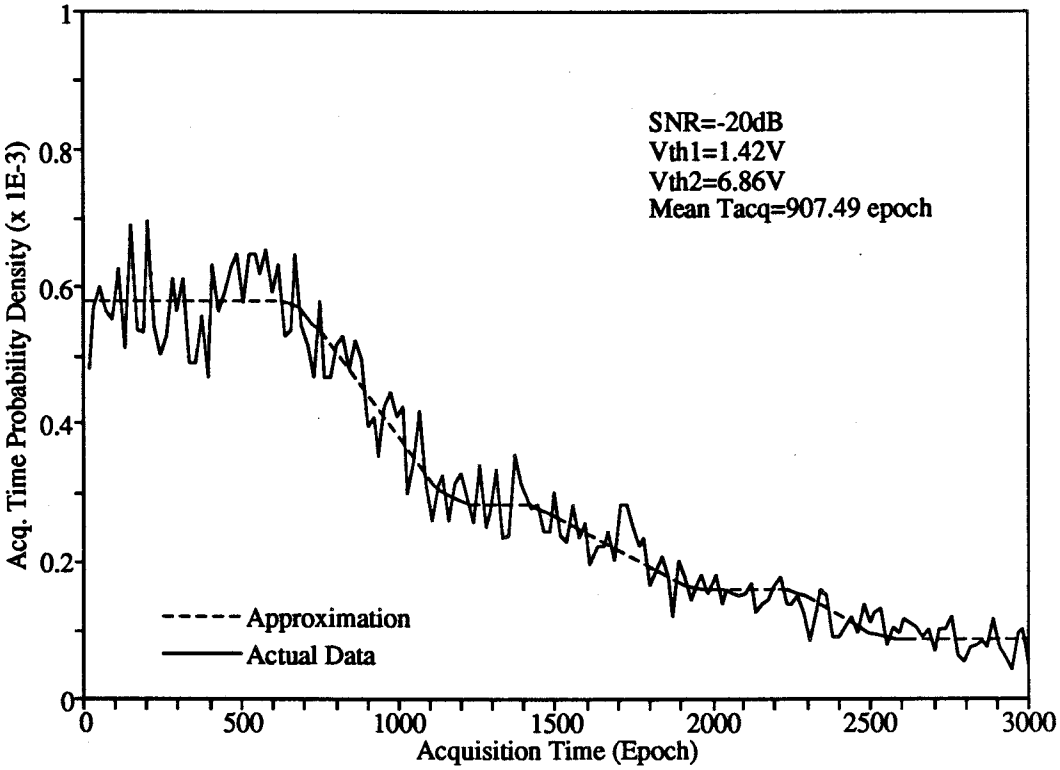


Fig. 6.13 30% Threshold Acquisition Time Test Results

As expected, with 30% detection threshold, the probability of detour and false alarm is larger than with 50% threshold. This can be seen from the slower roll off at each step of the distribution in Fig. 6.13. The shape of the distribution agrees closely with the predicted distribution shown in [16]. Measured mean acquisition time, $\overline{T_{acq}}$, (in units of epoch) was calculated as

$$\overline{T_{acq}} = \frac{1}{n} \sum_{i=1}^{k} (b_i \times c_i) \qquad (6.1)$$

where n is the total number of measurements included in the calculation, k is the total number of bins, b_i is the acquisition time bin value in epoch, and c_i is the number of acquisition time measurements fell into a specific bin b_i. Calculation was truncated at acquisition time up to 3000 epoch. Bin size was 18.77 epoch (or 0.25 second), hence k was 160. The mean acquisition time was found to be 907.49 epoch for 30% threshold compared to 896.42 epoch for 50% threshold.

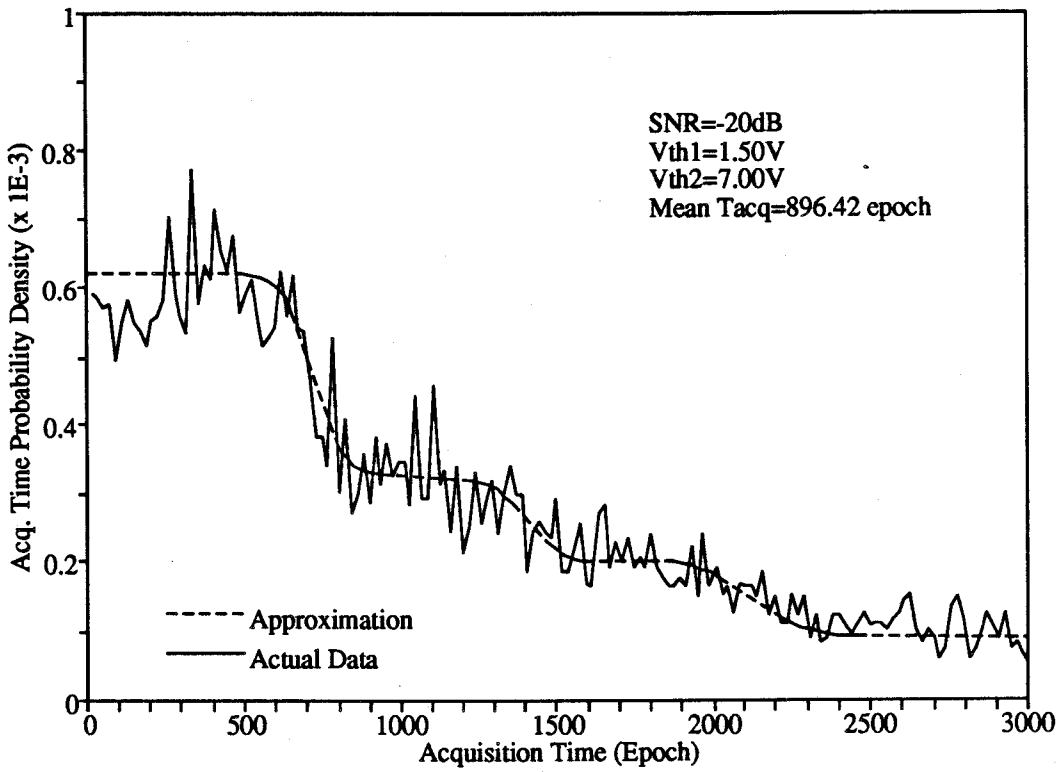


Fig. 6.14 50% Threshold Acquisition Time Test Results

6.4. RAATS Test Results

A series of acquisition time tests were performed on the RAATS system. Tests included no noise and SNR of -3 dB, -6 dB, -8 dB, and -10.5 dB. Threshold voltages must be set in both serial search and RAATS synchronizers. Since serial search acquires code phase by detecting the signal energy after the despread process, detection thresholds have to be adjusted according to SNR. In contrast, the RAATS system acquires code phase based on the transition sequence. The effects of noise are reduced by differential processing and by averaging, thus there is little need to adjust a threshold for different SNR. The RAATS threshold voltage, V_{tc}, is compared with the accumulated transition sequence voltage to give a binary sequence to match with the test pattern. Since both systems use the same verification and lock, V_{th2} is the same in both systems. All five acquisition tests were performed with the same V_{tc} and V_{th2}. A total of 10,000 measurements were obtained in each test. Test results are shown in Fig. 6.15 and 6.16. In all cases, acquisition time probability density function is approximately negative exponential. When noise is added to the SS signal, the distribution is depressed at the very beginning, reaches a peak at approximately 40 epoch, then decays exponentially. This is caused by the fact that additional accumulation is required before a clear test pattern can be detected.

Experimental mean acquisition time for all five cases were calculated using Equation (6.1). Based on the data obtained, the mean acquisition time is plotted in Fig. 6.17. As expected, mean acquisition time increases as noise increases. When SNR is less than -6 dB, the accumulated transition sequence is badly distorted. This causes false alarm to occur more frequently. As a result, acquisition time increases rapidly.

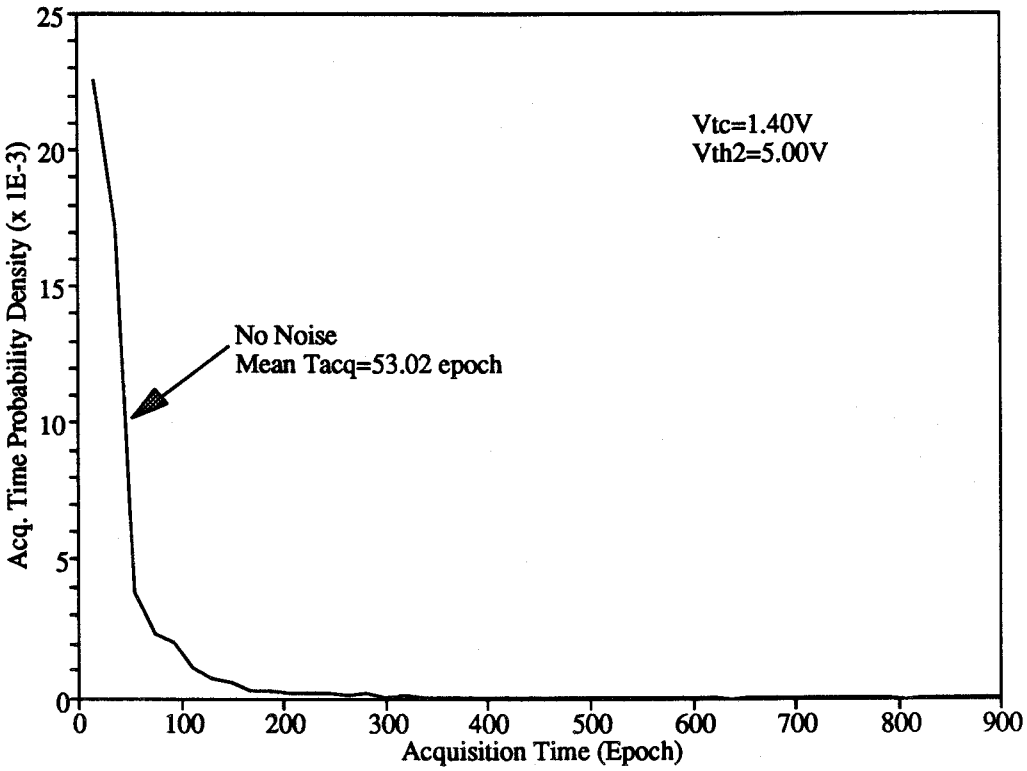


Fig. 6.15 RAATS Acquisition Time with No Noise

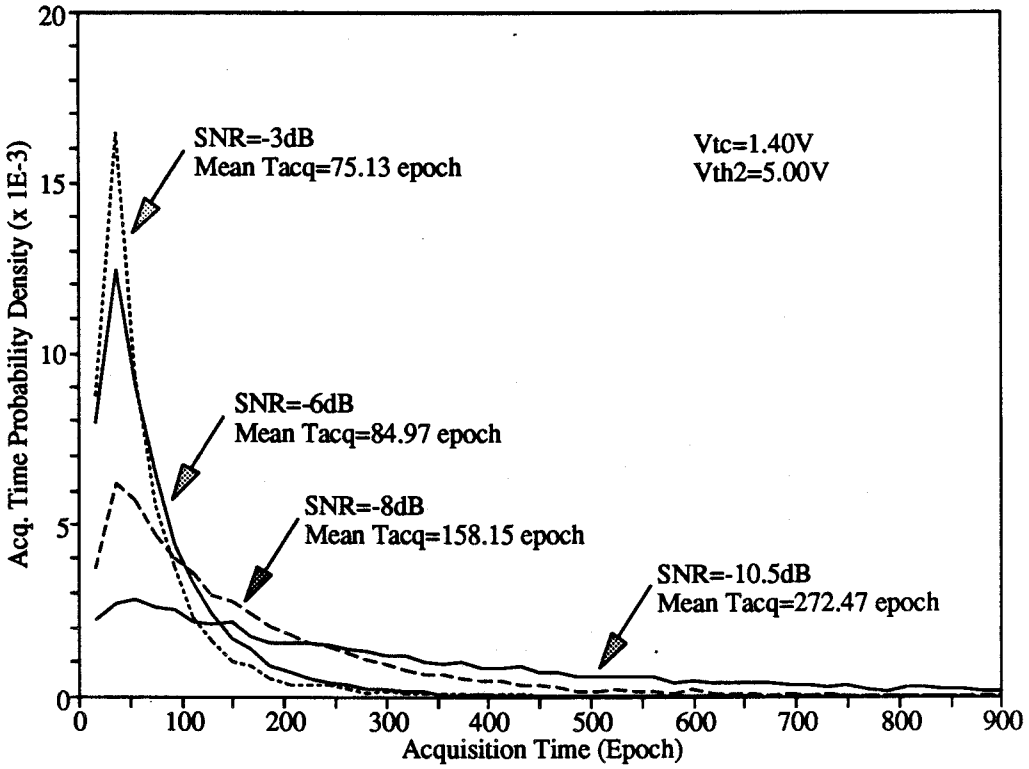


Fig. 6.16 RAATS Acquisition Time at Various SNR

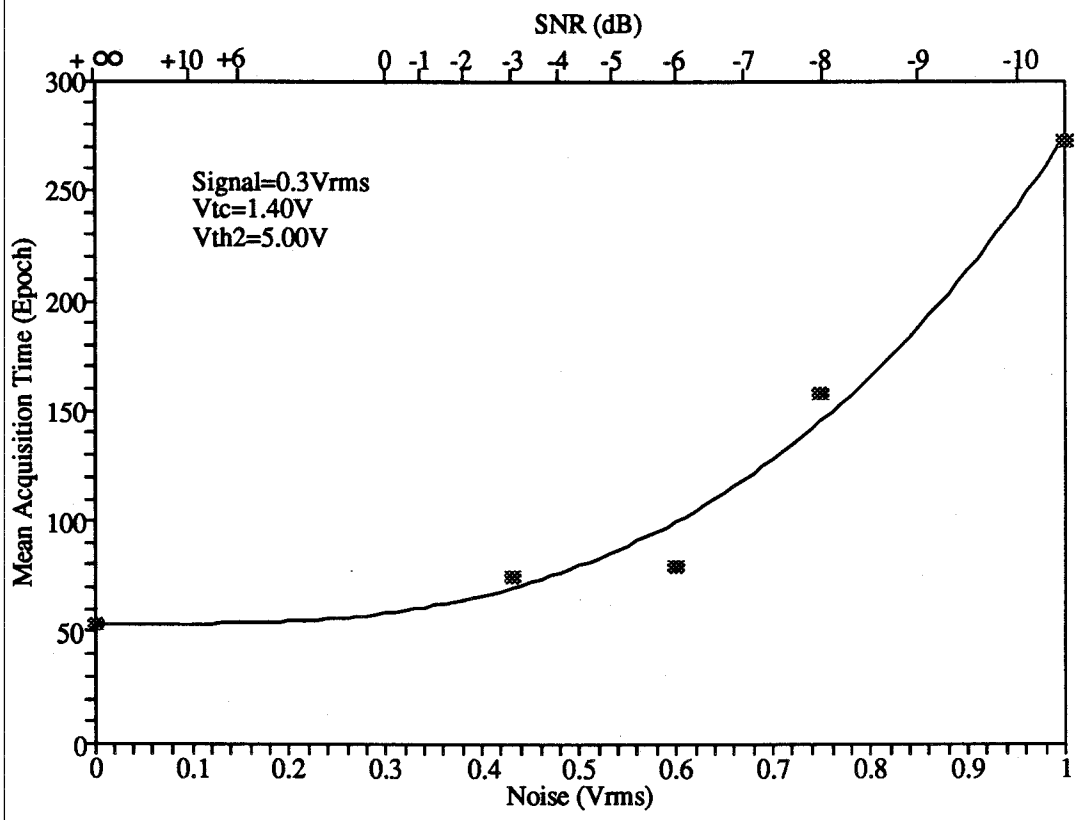


Fig. 6.17 Mean Acquisition Time of RAATS

Samples of the CCD output waveform (i.e. the accumulated transition sequence) at no noise and at SNR of -3 dB, -6 dB, and -8 dB are shown in Fig. 6.18 to 6.25. Referring to Fig. 6.18, the detection pattern is located at the peaks at approximately -6.00 ms and 7.50 ms. It repeats at exactly one PN code period. As shown in the plots, the transition sequences become noisy as SNR decreases. This makes detection difficult. However, the test pattern can still be located as more transition information is accumulated. As a result, acquisition time is expected to increase as SNR decreases.

The accumulated transition sequence voltage shown in Fig. 6.18 has been expanded in the region of the longest run as shown in Fig 6.19. The marked interval is the peak of the accumulated transition sequence and is measured to be 118 μs wide which is equal to nine chips. Since the peak is at approximately 1.50 V, by setting V_{tc} to 1.40 V, the test pattern is very easily detected. Trace 4 is the reset signal issued when the test pattern is located. Note that the high frequency variation of the accumulated transition sequence shown in Fig. 6.19 follows the instantaneous transition sequence shown in Fig. 5.6. The accumulated transition sequence voltage at -3 dB, -6 dB and -8 dB SNR were also expanded near the region of the longest run as shown in Fig. 6.21, 6.23, and 6.25. The effect of noise distortion can be seen by comparing these plots with the no noise condition shown in Fig. 6.19, .

Ideally, the accumulated transition sequence should have only two levels $\pm V_p$ where V_p is the peak voltage that the CCD can store. It is known that CCD have a small amount of change transfer inefficiency [29, 30, 31] which results in a fraction of charge being left behind when charge is transferred from one charge cell to the next. After a few transfers, the charge is no longer localized in one cell, but is spread out over several trailing cells.

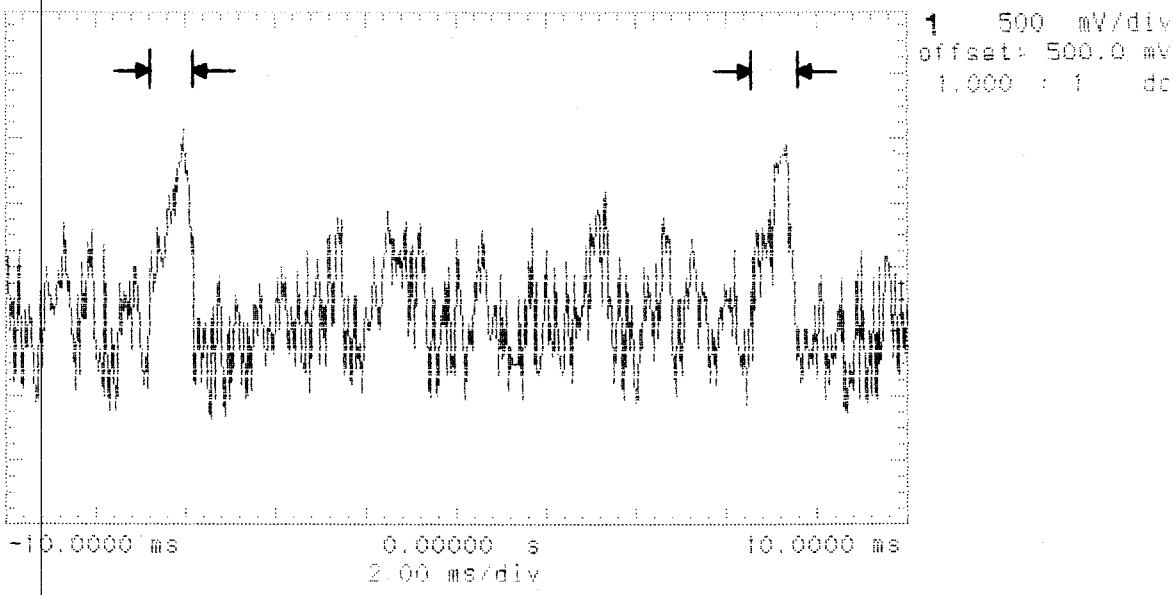


Fig. 6.18 Accumulated Transition Sequence with No Noise

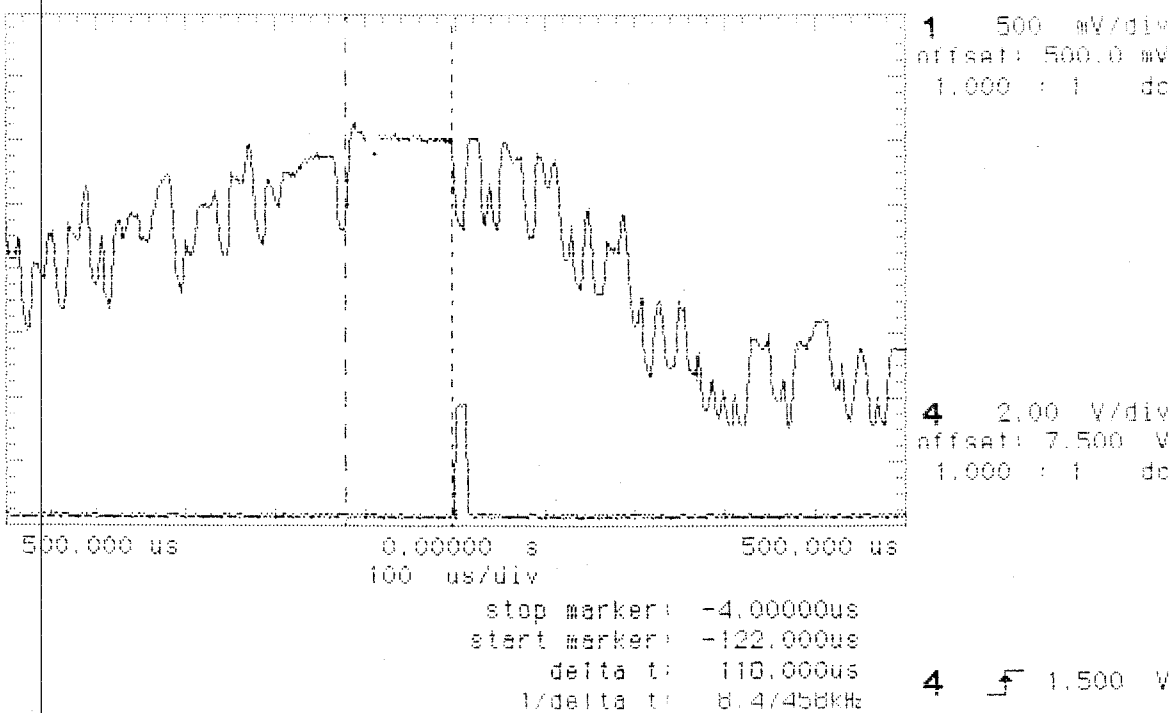


Fig. 6.19 Accumulated Test Pattern Area with No Noise

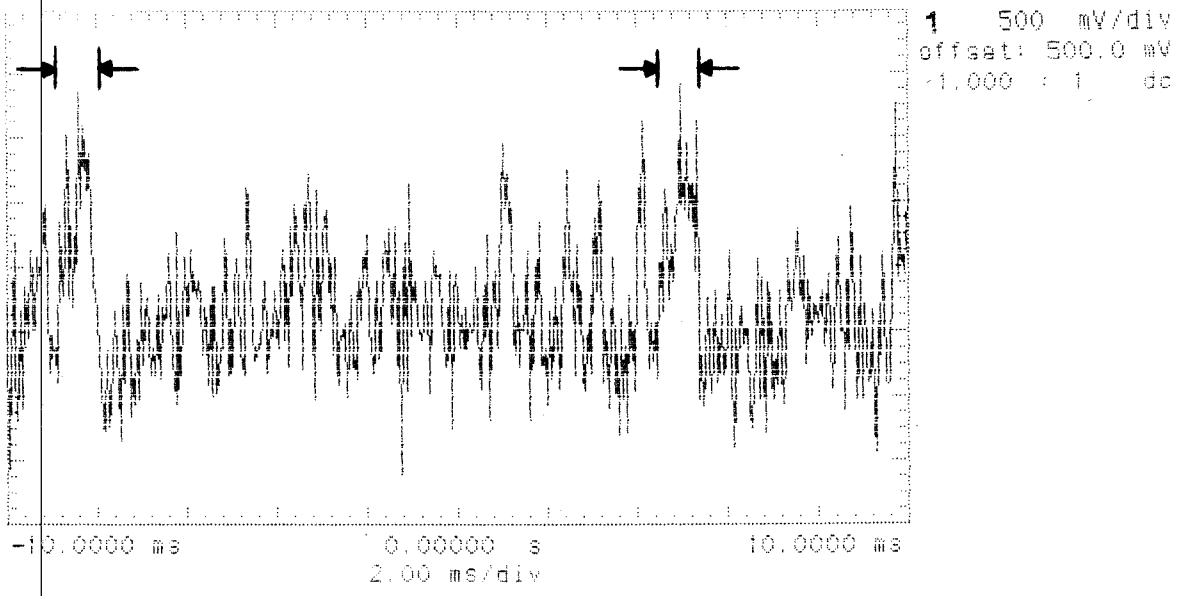


Fig. 6.20 Accumulated Transition Sequence at -3 dB SNR

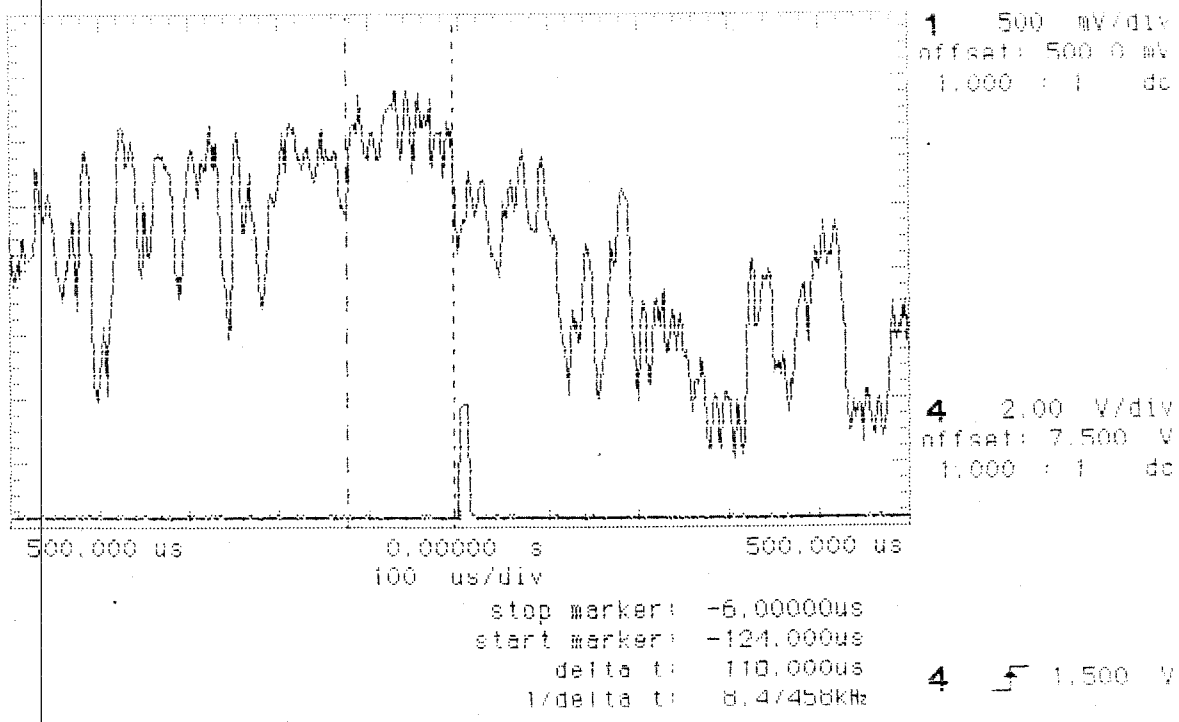


Fig. 6.21 Accumulated Test Pattern Area at -3 dB SNR

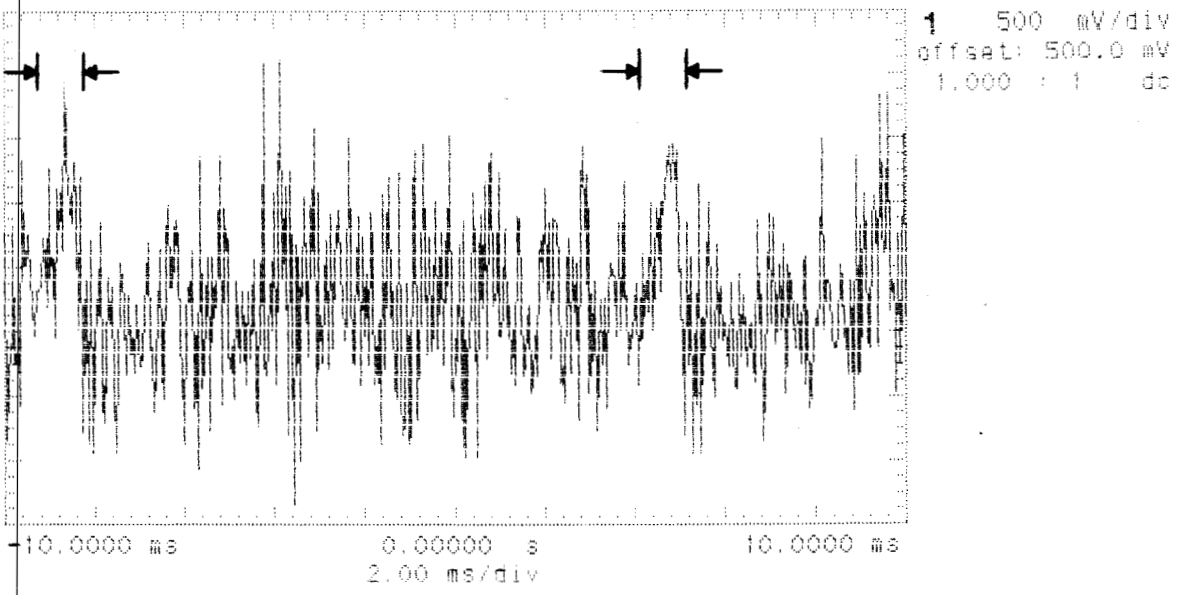


Fig. 6.22 Accumulated Transition Sequence at -6 dB SNR

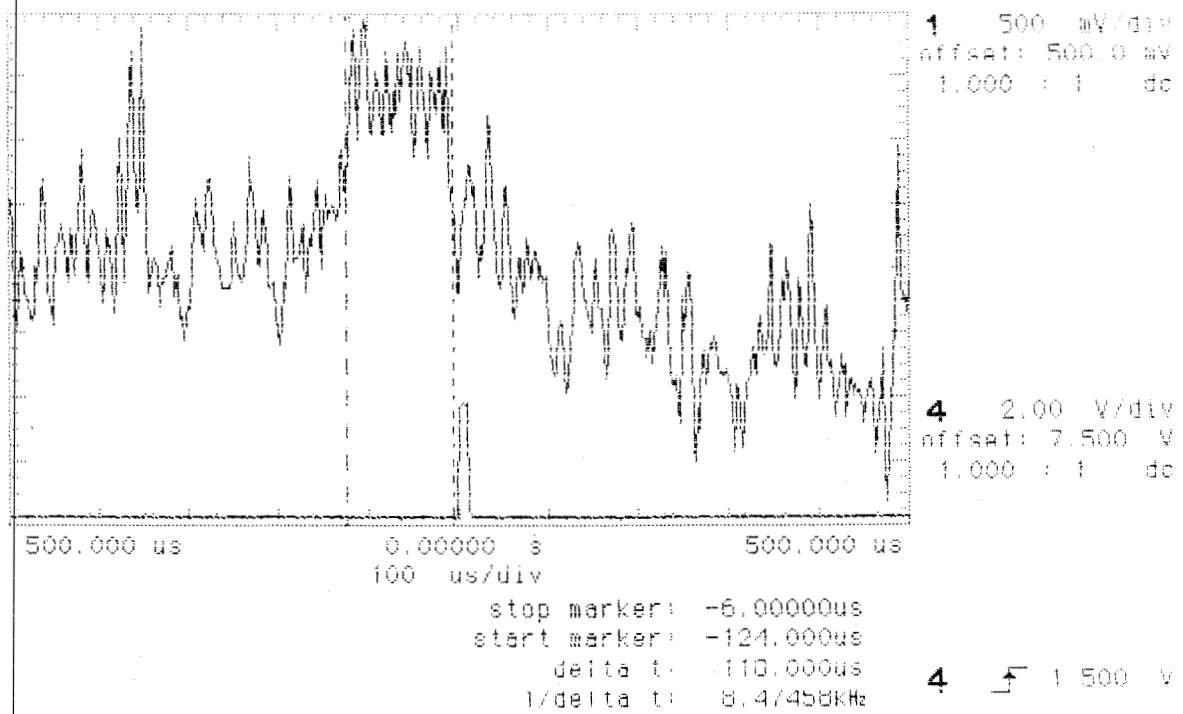


Fig. 6.23 Accumulated Test Pattern Area at -6 dB SNR

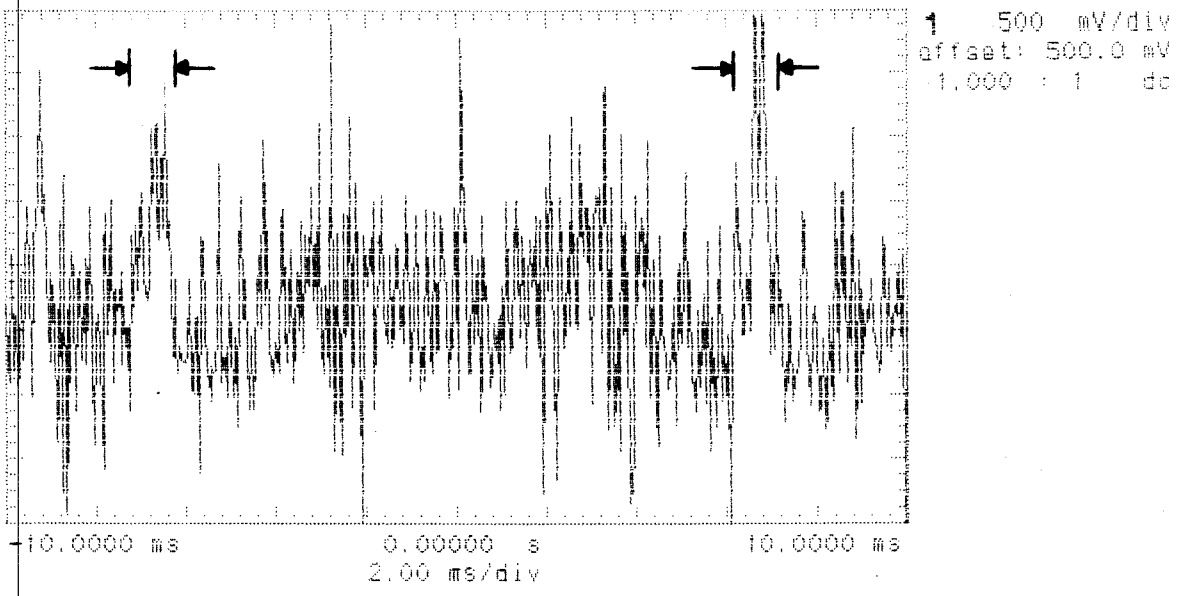


Fig. 6.24 Accumulated Transition Sequence at -8 dB SNR

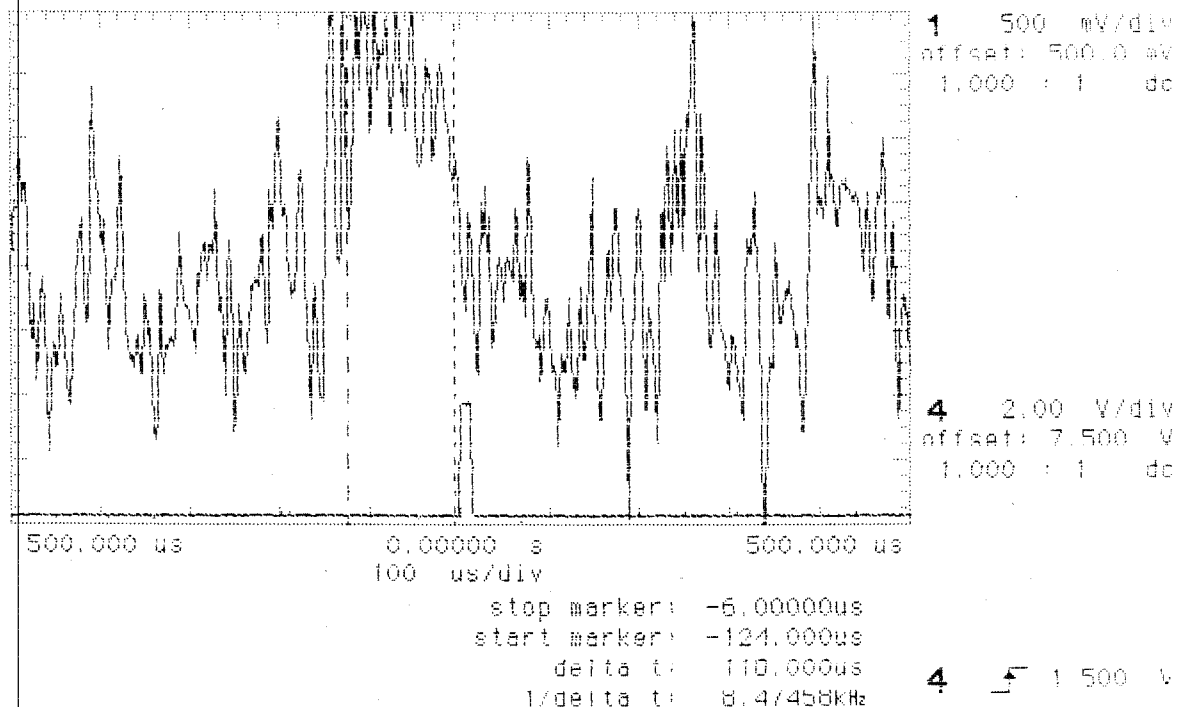


Fig. 6.25 Accumulated Test Pattern Area at -8 dB SNR

This is known as charge dispersion. Charge transfer inefficiency has two effects. One is the charge loss from the leading packet. The other is charge mixing. If there are charge packets followed, then the charge left behind by one packet will add to the trailing packets. Thus transfer inefficiency distorts the information. This is known as crosstalk. Crosstalk results in a charge mixing between adjacent cells which causes charge packet to smear with each other. It was the result of charge dispersion and crosstalk that causes the accumulated transition sequence to appear as shown in Fig. 6.18 to 6.25. To study the effect of charge transfer inefficiency, a program called TRANSIM was written in C to simulate the front portion of the RAATS code phase estimator. The source code is listed in Appendix H. TRANSIM models the transition estimator and the transition accumulator with a forward input gain G_{fw} of 0.1 and a "leaky" loop gain G_l of 0.9. The charge stored in the k^{th} CCD delay cell at sample time $t=mT_c$ is given by [29]

$$X_k(mT_c) = (1-\varepsilon)\, X_{k-1}([m-1]\,T_c) + \varepsilon\, X_k([m-1]\,T_c) \qquad (6.2)$$

where ε is the coefficient of charge transfer inefficiency. Through simulations, it was found that the CCD accumulator implemented has a charge transfer inefficiency of approximately 0.5%, i.e., $\varepsilon=0.005$. Fig. 6.26 shows the simulated transition sequence after 10 epoch of accumulation at no noise condition. It can be seen that the shape of the simulated transition sequence agrees closely with the experimental result shown in Fig. 6.18.

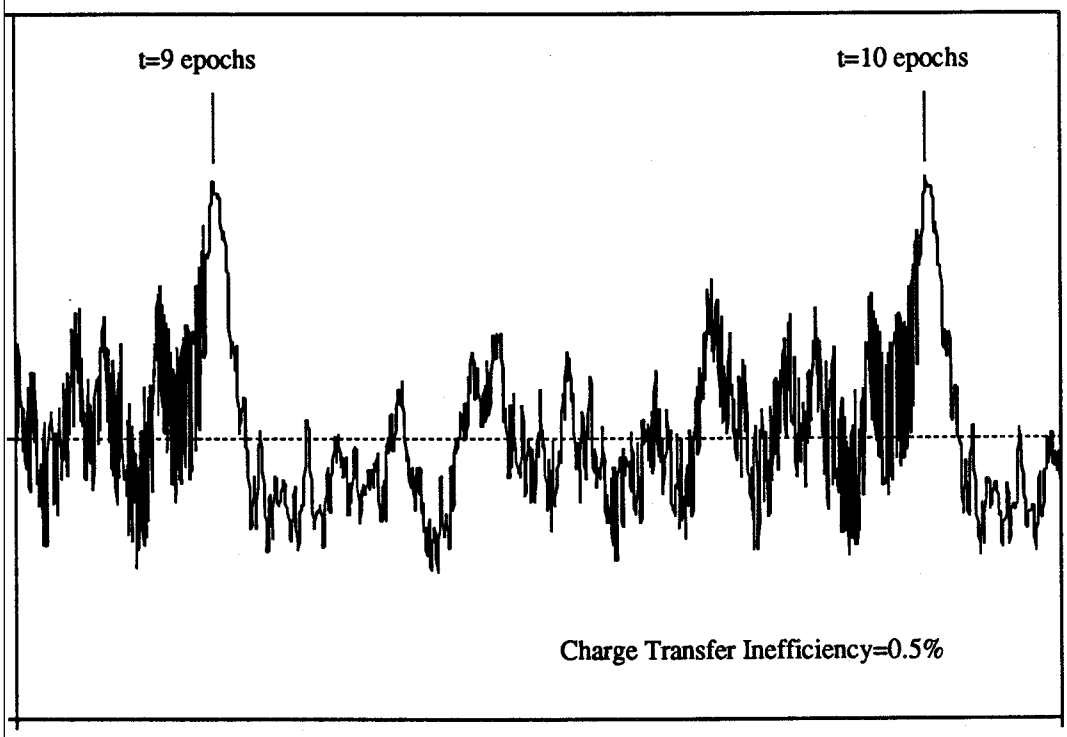


Fig. 6.26 Simulated Transition Sequence with No Noise

Chapter 7

SUMMARY AND CONCLUSIONS

Two DS SS acquisition systems were implemented and tested in this project. Source data was spread by a 1023-chip maximal length PN code sequence generated by a (3, 10) generator. A computerized test setup was developed which measures the acquisition time automatically.

7.1. Summary of Serial Search Acquisition System

The first acquisition scheme investigated in this thesis is the traditional "straight line" serial search acquisition. This scheme acquires PN code phase by a "trial and error" approach where all possible code phases are tested, one at a time, in sequence. Each test compares the despread signal energy during the dwell time with a pre-determined detection threshold. The acquisition scheme includes a 2-lock-state search/verification/lock strategy. The transition from one state to another depends on the despread signal energy.

Five main components of the serial search acquisition systems were implemented. They are the correlator, dwell time controller, detection threshold controller, SVL controller and PN code phase controller. The test system is designed for a 2400 bps input data and a chip rate of 76.8 kchips/sec which yields a processing gain of 32. The system clock rate is

921.6 kHz. The dwell time in search mode (T_{d1}) is eight data bit times (3.33 ms), and the dwell time in verification and lock modes (T_{d2}) is five times T_{d1}. A series of tests were performed on the system at -20 dB SNR. The threshold voltage V_{th1} (for dwell time T_{d1}) and V_{th2} (for dwell time T_{d2}) were adjusted in each test (10,000 measurements).

7.2. Summary of RAATS

The new acquisition scheme presented in this thesis is called Rapid Acquisition by Accumulation of Transition Sequence (RAATS). The code phase is acquired by detecting the transitions in the transmitted PN code sequence. Transition information is stored and accumulated in a cyclic analog shift register which provides an averaging effect and allows acquisition in presence of high noise. A PN code phase estimate is made by detecting a specific pattern of the average transition sequence.

In this implementation, the cyclic accumulator has a length equal to twice the PN code length. The code phase test pattern is "0111111111" which corresponds to the transition of the longest run (10 chips) in the PN sequence. The two-lock-state SVL strategy used in the serial search system is also used in RAATS, however, the serial search state is replaced by the code phase estimation. Acquisition tests were performed from no noise condition to -10.5 dB SNR with fixed detection thresholds. Each test consisted of 10,000 measurements.

7.3. Conclusions

Through a series of experiments, it was concluded that the RAATS outperforms the serial search. Experimental results have shown that the serial search acquisition time probability density declines in steps as predicted by Dodds, Kumar and Pan [16]. Mean acquisition time at -20 dB SNR was 907.49 epoch and 896.42 epoch for 30% and 50% thresholds, respectively. These tests also confirmed that the acquisition time performance of serial search is somewhat sensitive to detection thresholds.

In contrast, the RAATS acquisition time probability density has an exponentially declining shape. Mean acquisition time was measured at 53.02 epoch at no noise, and increased to 272.47 epoch at -10.5 dB SNR. The mean acquisition time increased significantly for SNR less than -6 dB. This was caused by the distortion of the transition sequence due to high noise. These experiments have demonstrated that the RAATS is able to operate effectively over a range of SNR without a change in its detection threshold, however, the verification and lock threshold should be changed as required for serial search. In addition, the averaging effect of the RAATS system significantly reduced the occurrence of detour and false alarm during the search process. In most circumstances, the system entered the lock state without any detour or false alarm.

7.4. Suggestions for Future Study

Although the RAATS acquisition system demonstrates significant improvement in acquisition time performance over the traditional serial search system, the experimental

prototype was only a suboptimal system and there are areas in which further study could be undertaken.

The cyclic accumulation of the transition sequence is a feature unique to the new acquisition system. However, informal tests indicated that acquisition time is sensitive to the change of the accumulation time constant. Optimum performance can be achieved by adjusting the forward and the feedback gain of the transition accumulator. Furthermore, the content of the transition accumulator should be cleared every time that detour or false alarm occurs. Theoretical studies or computer simulations could provide a way to optimize the time constant for the best performance at various SNR.

The cyclic transition accumulator was created by cascading CCD chips in series. Distortion was experienced at the output of each IC package. In addition, the maximum sampling frequency of the CCD had limited the system to a rather low chip rate. Other forms of sample-and-hold devices could be studied to enhance the system.

The RAATS system has an upper limit on SNR. When SNR is below -10.5 dB, the system is unable to locate the code phase. Changes could be made to improve the system so that lower SNR performance could be achieved. Enhanced performance could be obtained by replacing the transition accumulator and the short binary pattern check with a full transition sequence correlator built on a surface acoustic wave (SAW) or equivalent device.

In summary, the work presented in this thesis verifies the acquisition time performance of a "straight line" serial search scheme. The new RAATS system is shown

to be a faster acquisition method. With advances in analog and digital Application Specific Integrated Circuits (ASIC) technology, it is possible to implement RAATS in an integrated package. RAATS could become an attractive option for spread spectrum communication systems which require fast code phase acquisition at moderate signal-to-noise ratio.

REFERENCES

1. Sklar, B., *Digital Communications Fundamentals and Applications*, Prentice Hall, New Jersey, 1988.

2. Simon, M. K., Omura, J. K., Scholtz, R. A. and Levitt, B. K., *Spread Spectrum Communications, Vol. I.* Computer Science Press, 1985.

3. Cooper, G. R. and McGillem C. D., *Modern Communications and Spread Spectrum*, McGraw-Hill, New York, 1986.

4. Ziemer, R. E. and Peterson, R. L., *Digital Communications and Spread Spectrum Systems*, McMillan, 1985.

5. Denaro, R. P., "Navstar: The All-purpose Satellite," *IEEE Spectrum*, May 1981, pp.35-40.

6. Sudworth, J. P., "A Simple Navstar Receiver," *Proceeding of the International Specialist Seminar on Case Studies in Advanced Signal Processing*, Sept. 18-21, 1979, pp. 85-89.

7. Cager, R. H., Jr., LaFlame, D. T. and Parode, L. C., "Orbiter Ku-Band Integrated Radar and Communications Subsystem," *IEEE Transactions on Communications*, Vol. COM-26, No. 11, Nov. 1978, pp.1604-1619.

8. Alem, W. K., Huth, G. K., Holmes, J. K. and Udalov, S., "Spread Spectrum Acquisition and Tracking Performance for Shuttle Communication Links," *IEEE Transactions on Communications*, Vol. COM-26, No. 11, Nov. 1978, pp.1689-1703.

9. Tayor, J. T. and Omura, J. K., "Spread Spectrum Technology: A Solution to the Personal Communications Services Frequency Allocation Dilemma," *IEEE Communications Magazine*, Feb. 1991, Vol. 29, No. 2, pp.48-51.

10. Smith, D. G., "Spread Spectrum for Wireless Phone Systems: The Subtle Interplay between Technology and Regulation," *IEEE Communications Magazine*, Feb. 1991, Vol. 29, No. 2, pp.44-46.

11. Lynch, R. J., "PCN: Son of Cellular? The Challenges of Providing PCN Service," *IEEE Communications Magazine*, Feb. 1991, Vol. 29, No. 2, pp.56-57.

12. Singer, R. M. and Irwin, D. A., "Personal Communications Services: The Next Technological Revolution," *IEEE Communications Magazine*, Feb. 1991, Vol. 29, No. 2, pp.62-66.

13. Schilling, D. L., Milstein, L. B., Pickholtz, R. L., Kullback, M. and Miller. F., "Spread Spectrum for Commercial Communications," *IEEE Communications Magazine*, April 1991, Vol. 29, No. 4, pp.66-79.

14. Hopkins, P. M., "A Unified Analysis of Pseudonoise Synchronization by Envelope Correlation," *IEEE Transactions on Communications*, Vol. COM-25, No. 8, Aug. 1977, pp.770-778.

15. Pan, S. M., Dodds, D. E. and Kumar, S., "Statistical Distribution of PN Acquisition Time in Direct-sequence Spread Spectrum Receivers," *Proceedings of ICC'89*, Boston, June 1989, pp.27.3.1-27.3.8.

16. Dodds, D. E., Kumar, S. and Pan, S. M., "Factors Affecting Serial Search Performance in Spread Spectrum Systems," *Proceedings of Canadian Conference on Electrical and Computer Engineering*, Montreal, Sept. 1989, pp. 338-391.

17. Kumar, S., Salt, J. E., Dodds, D. E. and Pan, S. M., "Single and Dual Multi-Dwell Synchronization for a Direct-Sequence Spread Spectrum Receiver," *Canadian Journal of Electrical & Computer Engineering*, Vol. 16, No. 1, 1991, pp.34-40.

18. Scholtz, R. A., "The Spread Spectrum Concept," *IEEE Transactions on Communications*, Vol. COM-25, No. 8, Aug. 1977, pp.748-755.

19. Pickholtz, R. L., Schilling, D. L. and Milstein, L. B., "Theory of Spread-Spectrum Communications-A Tutorial," *IEEE Transactions on Communications*, Vol. COM-30, No. 5, May 1982, pp.855-884.

20. Holmes, J. K., *Coherent Spread Spectrum Systems*. Wiley-Interscience, New York, 1982.

21. Weinberg, A., "Generalized Analysis for the Evaluation of Search Strategy Effects on PN Acquisition Performance," *IEEE Transactions on Communications*, Vol. COM-31, No. 1, Jan. 1983, pp.37-49.

22. Polydoros, A. and Weber, C. L., "A Unified Approach to Serial Search Spread-Spectrum Code Acquisition-Part I: General Theory," *IEEE Transactions on Communications*, Vol. COM-32, No. 5, May 1984, pp.542-549.

23. Jovanovic, V. M., "Analysis of Strategies for Serial-Search Spread-Spectrum Code Acquisition-direct Approach," *IEEE Transactions on Communications*, Vol. 36, No. 11, Nov. 1988, pp.1208-1220.

24. Ward, R. B., "Acquisition of Pseudonoise Signals by Sequential Estimation," *IEEE Transactions on Communications*, Vol. COM 13, Dec. 1965, pp.475-483.

25. Rappaport, S. S. and Grieco, D. M., "Spread-Spectrum Signal Acquisition: Methods and Technology," *IEEE Communications Magazine*, June 1984, Vol. 22, No. 6, pp.6-21.

26. Hamamoto, N. and Masamura, T., "Japan's CS (SAKURA) Communications Satellite Experiments. Part VI C. Communications Experiments: Spread Spectrum Multiple Access," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-22, No. 3, May 1986, pp.281-290.

27. Hayes, D. P. and Ha, T. T., "A Performance Analysis of DS-CDMA and SCPC VSAT Networks," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 26, No. 1, Jan. 1990, pp.12-21.

28. Matthews, H., *Surface Wave Filters*, John Wiley & Sons, Inc., New York, 1977.

29. Magill, E. G., Grieco, D. M., Dyck, R. H. and Chen, P. C. Y., "Charge-Coupled Device Pseudo-Noise Matched Filter Design," *Proceedings of the IEEE*, Vol. 67, No. 1, Jan. 1979, pp.50-60.

30. Hobson, G. S., *Charge-Transfer Devices*, John Wiley & Sons, Inc., New York, 1978.

31. Beynon, J. D. E. and Lamb, D. R., *Charge-Coupled Devices and Their Applications*, McGraw-Hill, Maidenhead, Berkshire, England, 1980.

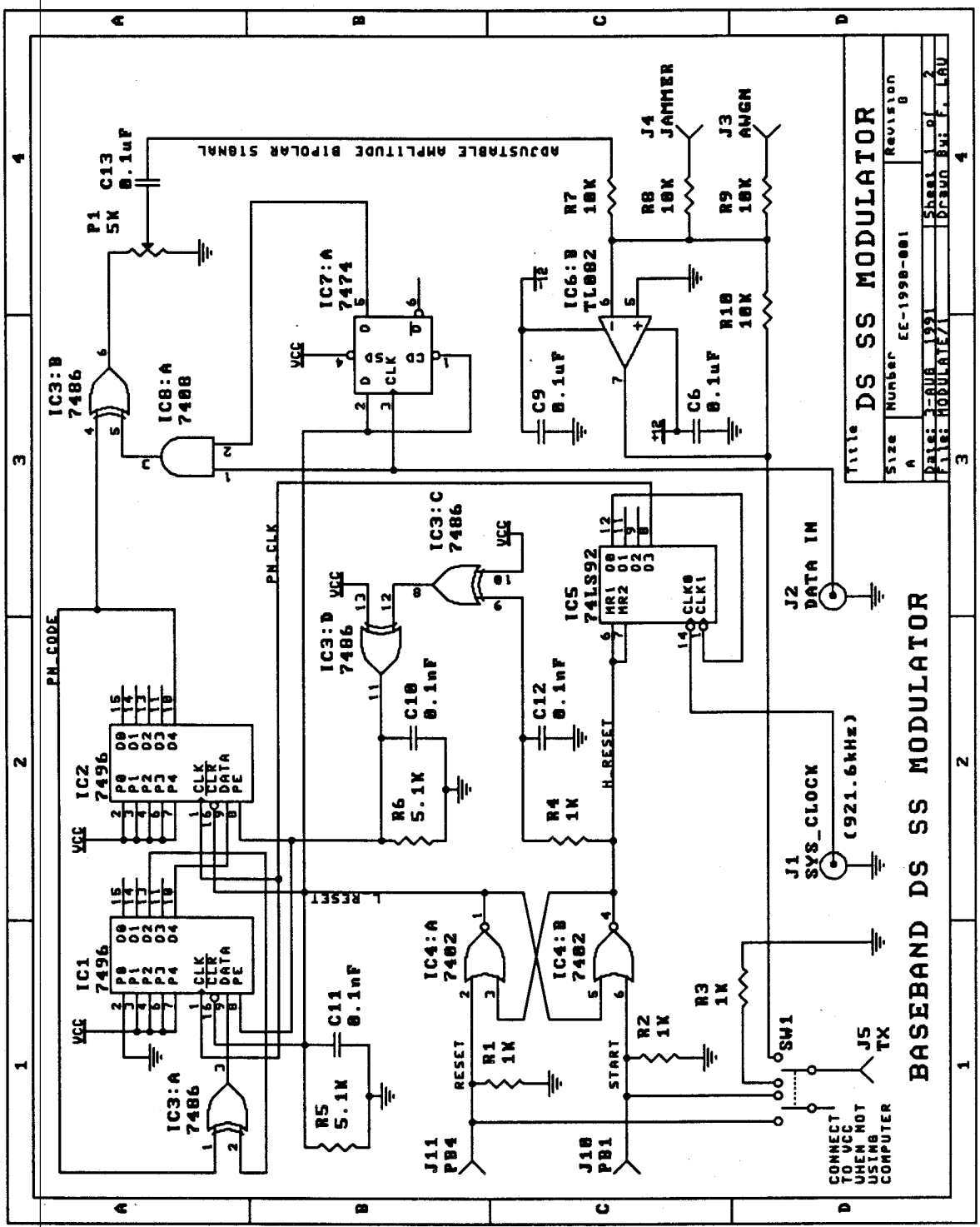
32. *Analog Signal Processing Integrated Circuits*, EG&G Reticon, Sunnyvale, California, 1987.

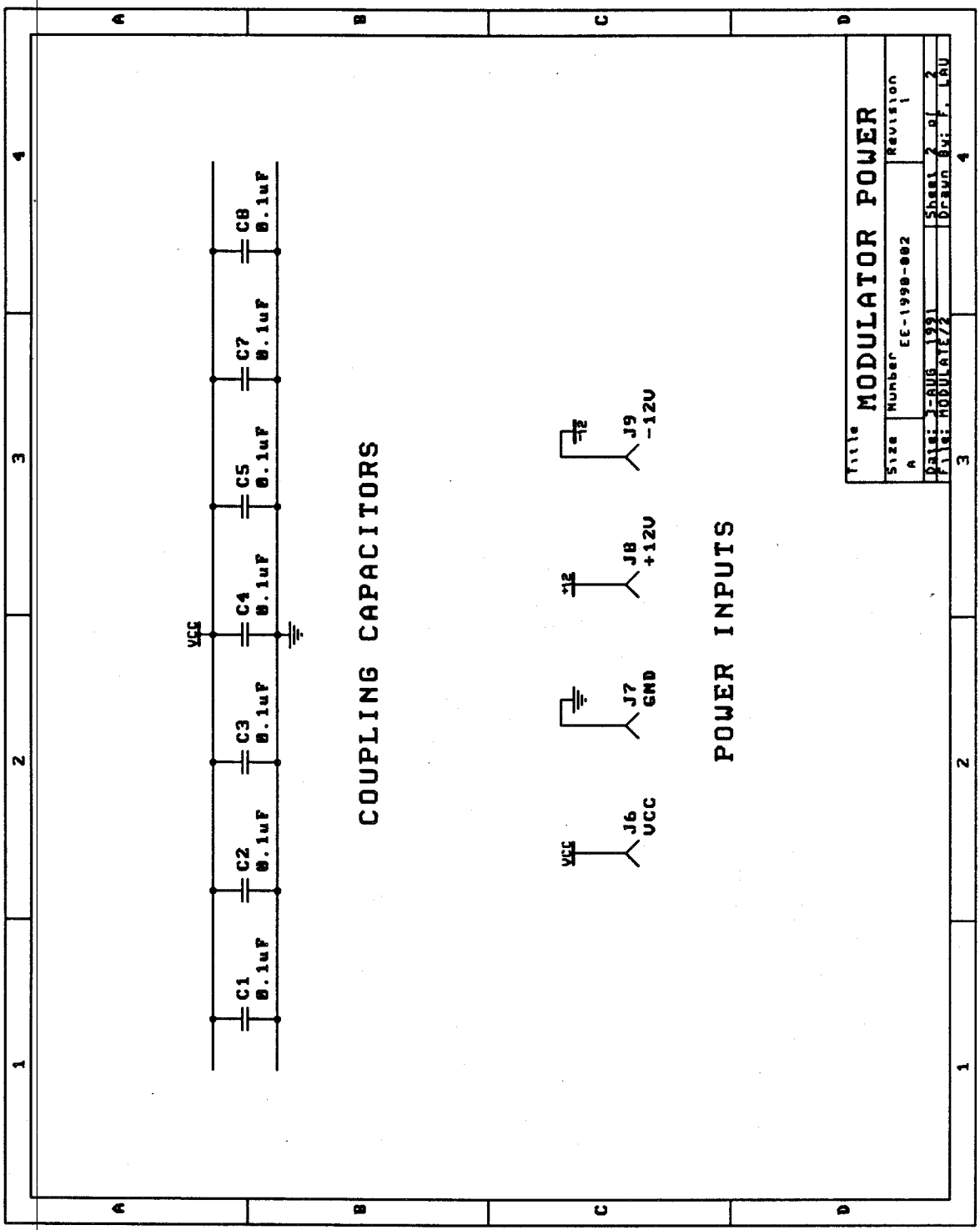
33. Pan, S. M., *Robust Frame Synchronization for Noisy PCM Systems*. M. Sc. Thesis, Department of Electrical Engineering, University of Saskatchewan, Saskatoon, Saskatchewan, March 1985.

34. Osborne, A., Strasma, J. and Strasma, E., *CBM Professional Computer Guide*, Osborne/McGraw-Hill, Berkeley, California, 1982.

APPENDIX A

BASEBAND DS SS MODULATOR SCHEMATIC

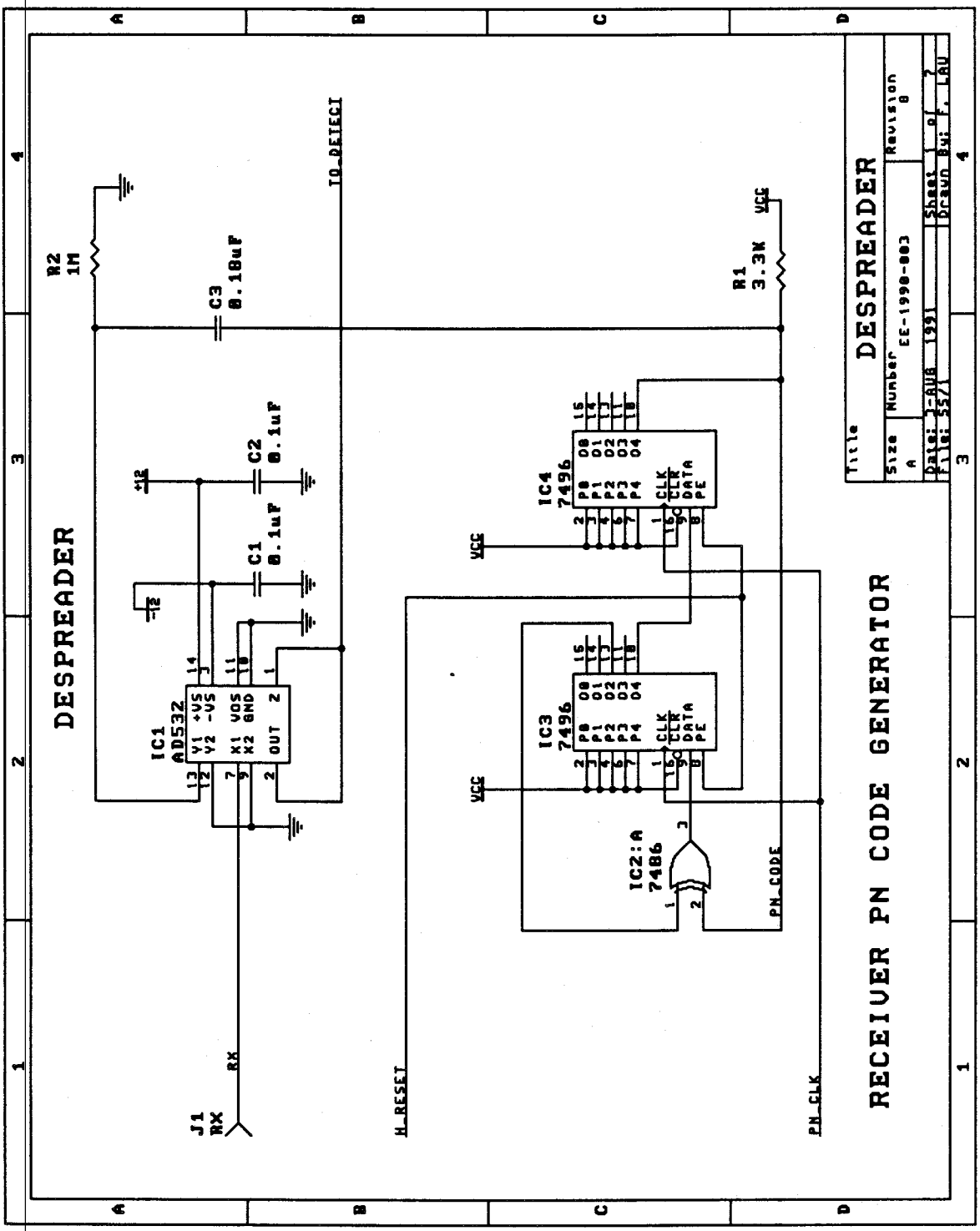

BASEBAND DS SS MODULATOR
DS SS MODULATOR
ADJUSTABLE AMPLITUDE BIPOLAR SIGNAL
PN_CODE
PN_CLK
H_RESET
RESET
START
SYS_CLOCK
(921.6kHz)
DATA IN
JAMMER
AWGN
TX
CONNECT TO VCC WHEN NOT USING COMPUTER
IC1 7496
IC2 7496
IC3:A 7486
IC3:B 7486
IC3:C 7486
IC3:D 7486
IC4:A 7402
IC4:B 7402
IC5 74LS92
IC6:B TL082
IC7:A 7474
IC8:A 7408
P1 5K
R1 1K
R2 1K
R3 1K
R4 1K
R5 5.1K
R6 5.1K
R7 10K
R8 10K
R9 10K
R10 10K
C6 0.1uF
C9 0.1uF
C10 0.1nF
C11 0.1nF
C12 0.1nF
C13 0.1uF
J1 SYS_CLOCK
J2 DATA IN
J3 AWGN
J4 JAMMER
J5 TX
J10 PB1
J11 PB4
SW1

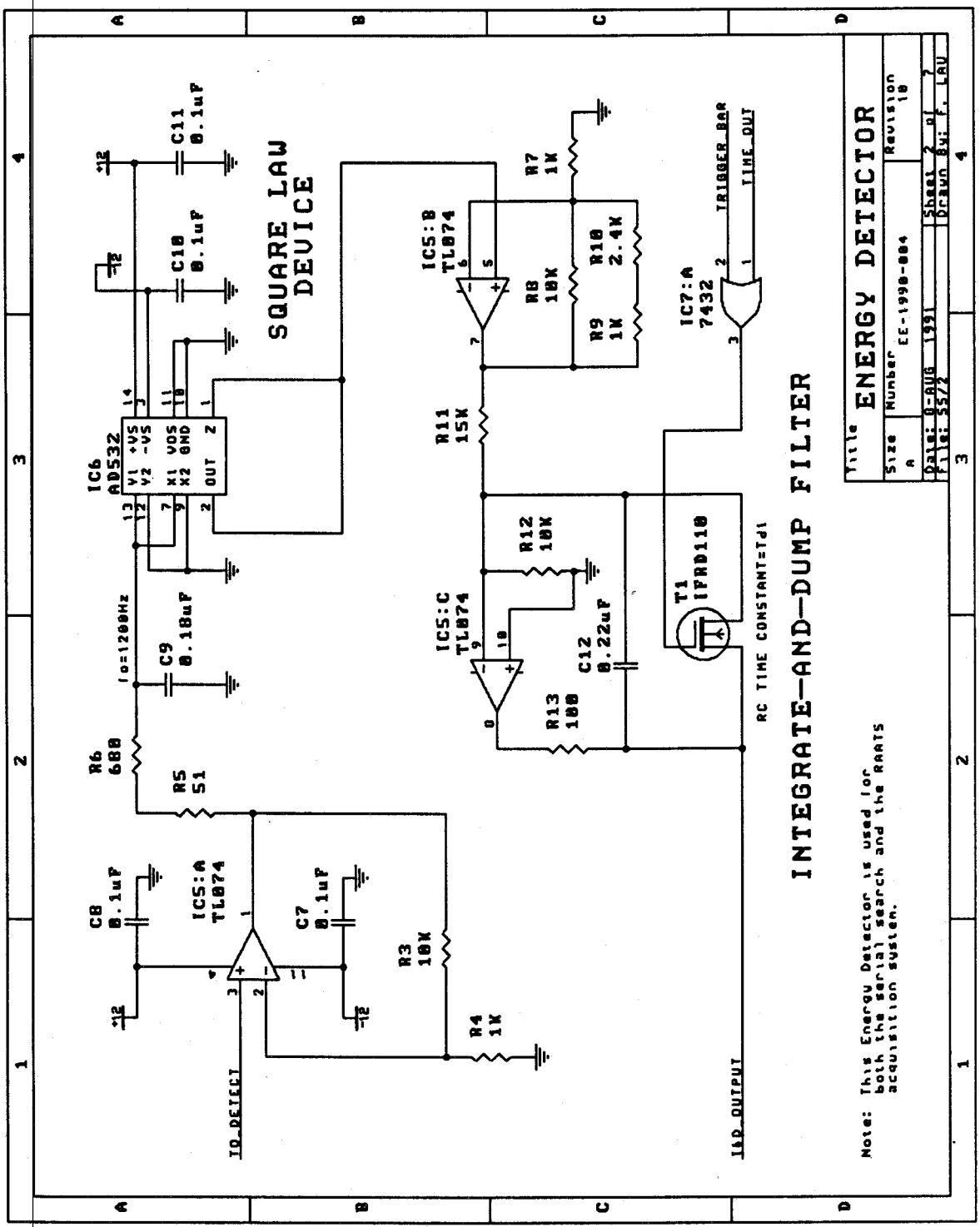

COUPLING CAPACITORS
C1 0.1uF
C2 0.1uF
C3 0.1uF
C4 0.1uF
C5 0.1uF
C7 0.1uF
C8 0.1uF
VCC
POWER INPUTS
J6 VCC
J7 GND
J8 +12V
J9 -12V
Title MODULATOR POWER
Size A
Number EE-1990-002
Revision 1
Date: 3-AUG 1991
File: MODULATE/2
Sheet 2 of 2
Drawn By: F. LAU

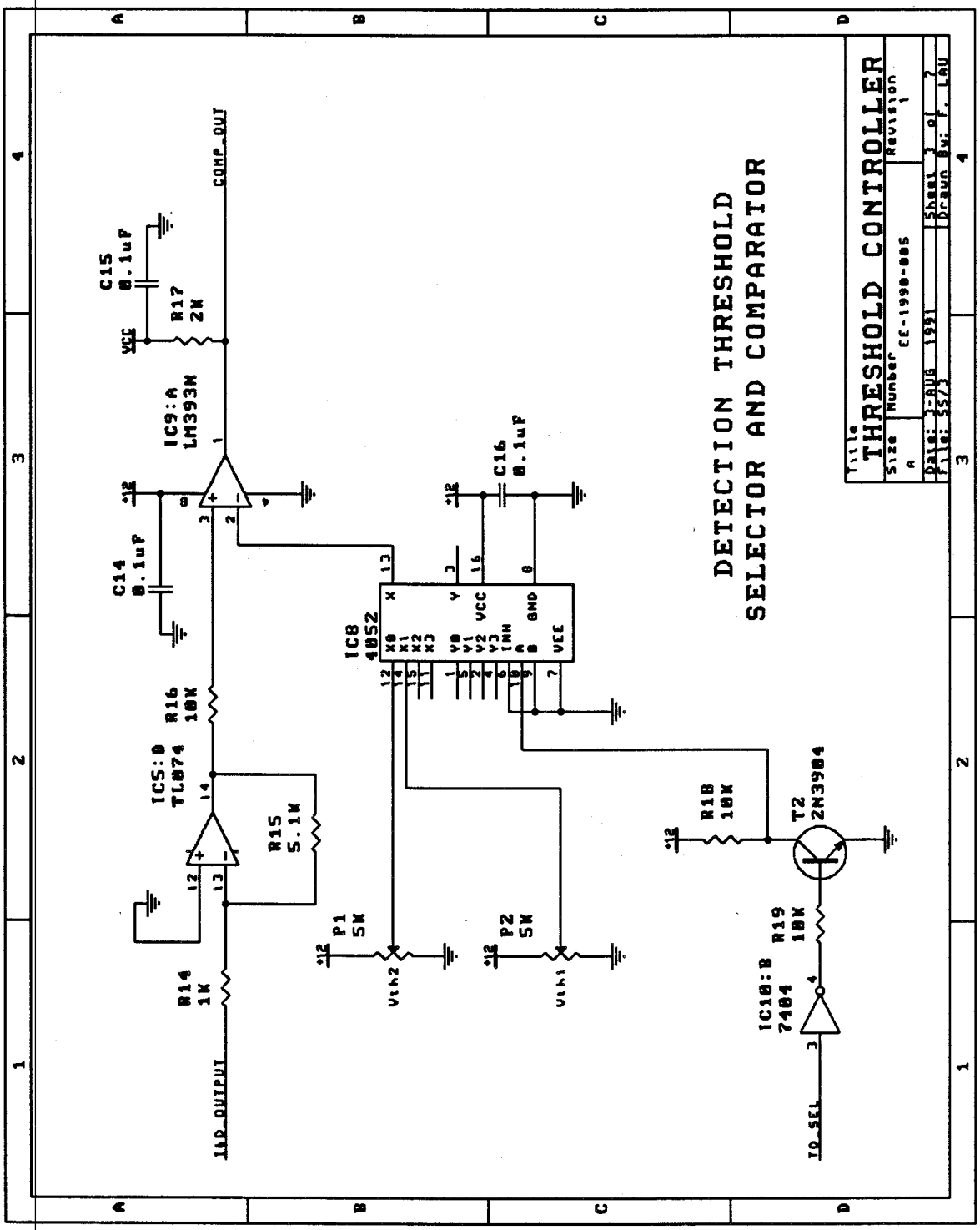
APPENDIX B

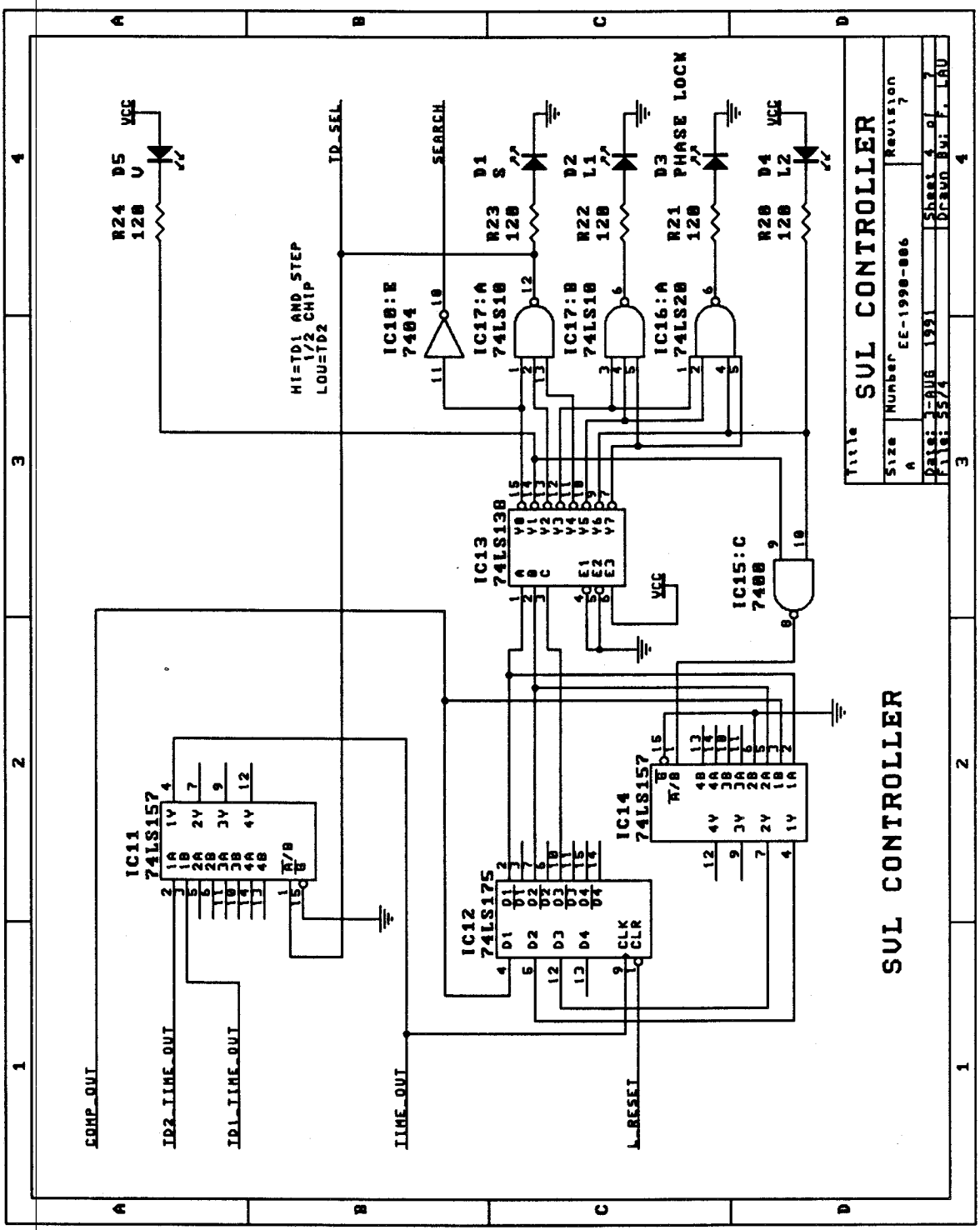
SERIAL SEARCH ACQUISITION SYSTEM SCHEMATIC

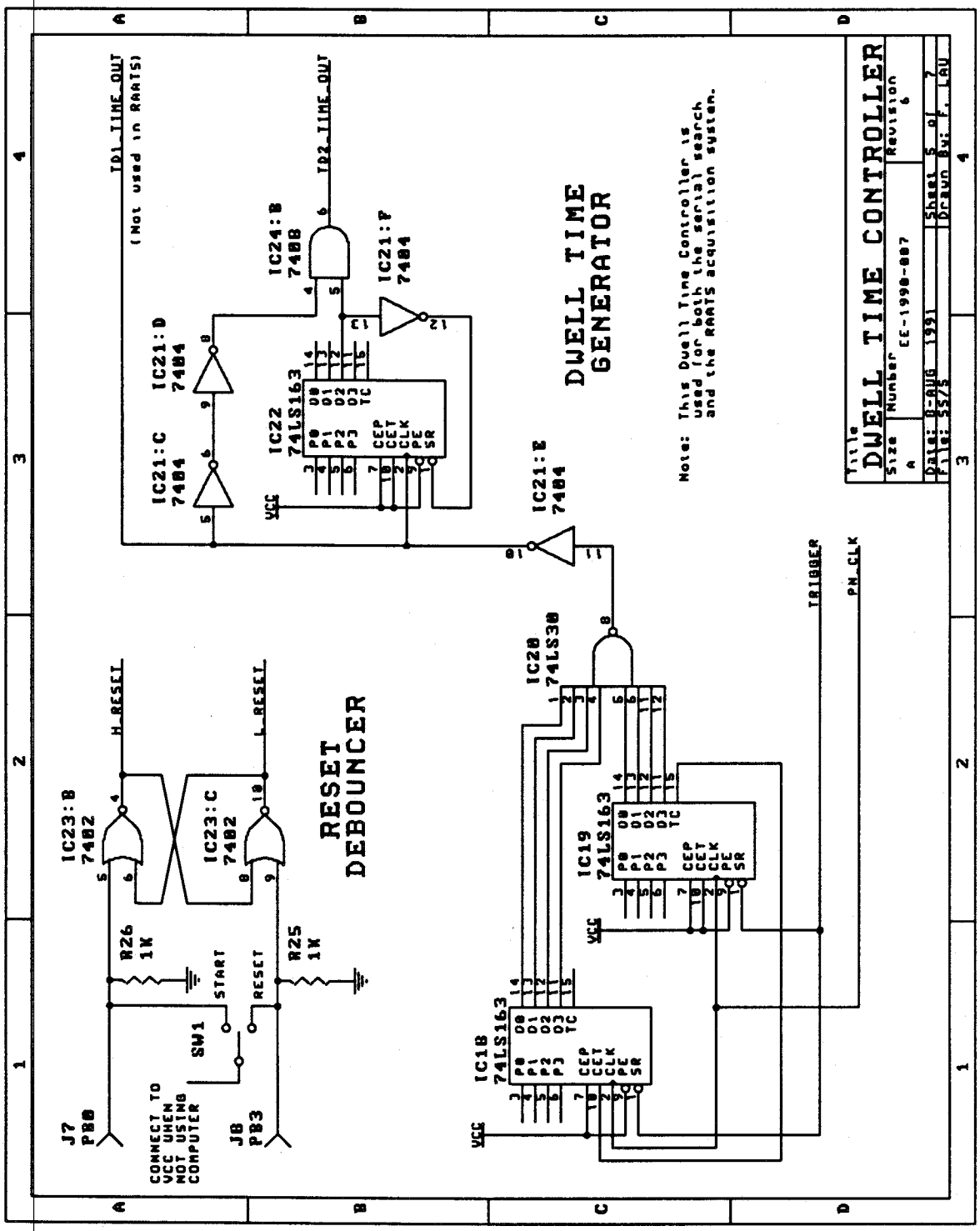
Note: The Energy Detector and the Dwell Time Controller shown on page 111 and 114, respectively, are used in both the serial search and the RAATS acquisition system.

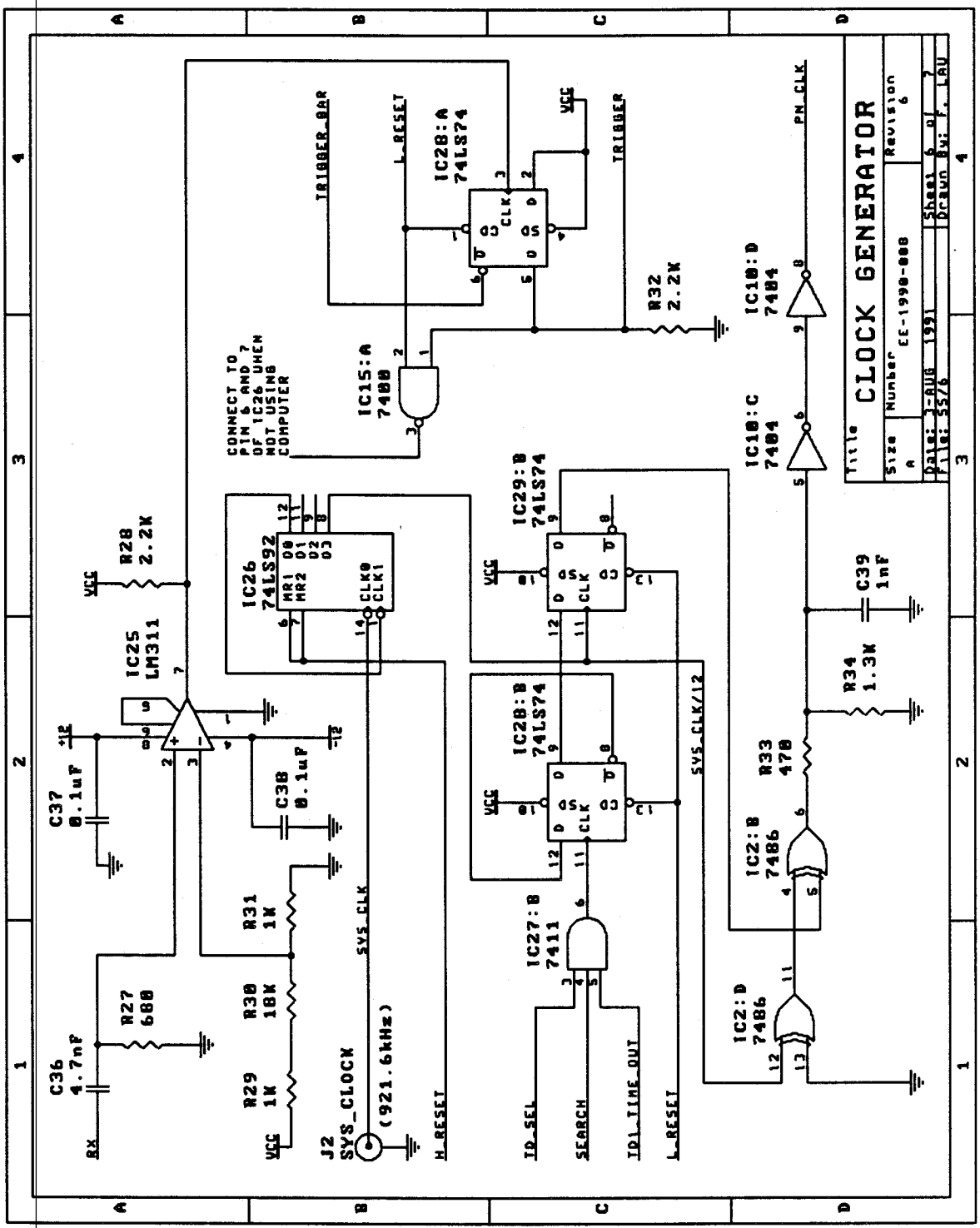

DESPREADER
R2
1M
C3
0.18uF
TO_DEIECI
+12
C2
0.1uF
C1
0.1uF
-12
IC1
AD532
Y1 +VS
Y2 -VS
X1 VOS
X2 GND
OUT
J1
RX
RX
R1
3.3K
VCC
IC4
7496
IC3
7496
P0 P1 P2 P3 P4
CLK
CLR
DATA
PE
IC2:A
7486
H_RESET
PN_CODE
PN_CLK
RECEIVER PN CODE GENERATOR
Title
DESPREADER
Size
A
Number
EE-1998-003
Revision
0
Date: 1-AUG 1991
File: SS7\
Sheet 1 of 2
Drawn By: F. LAU

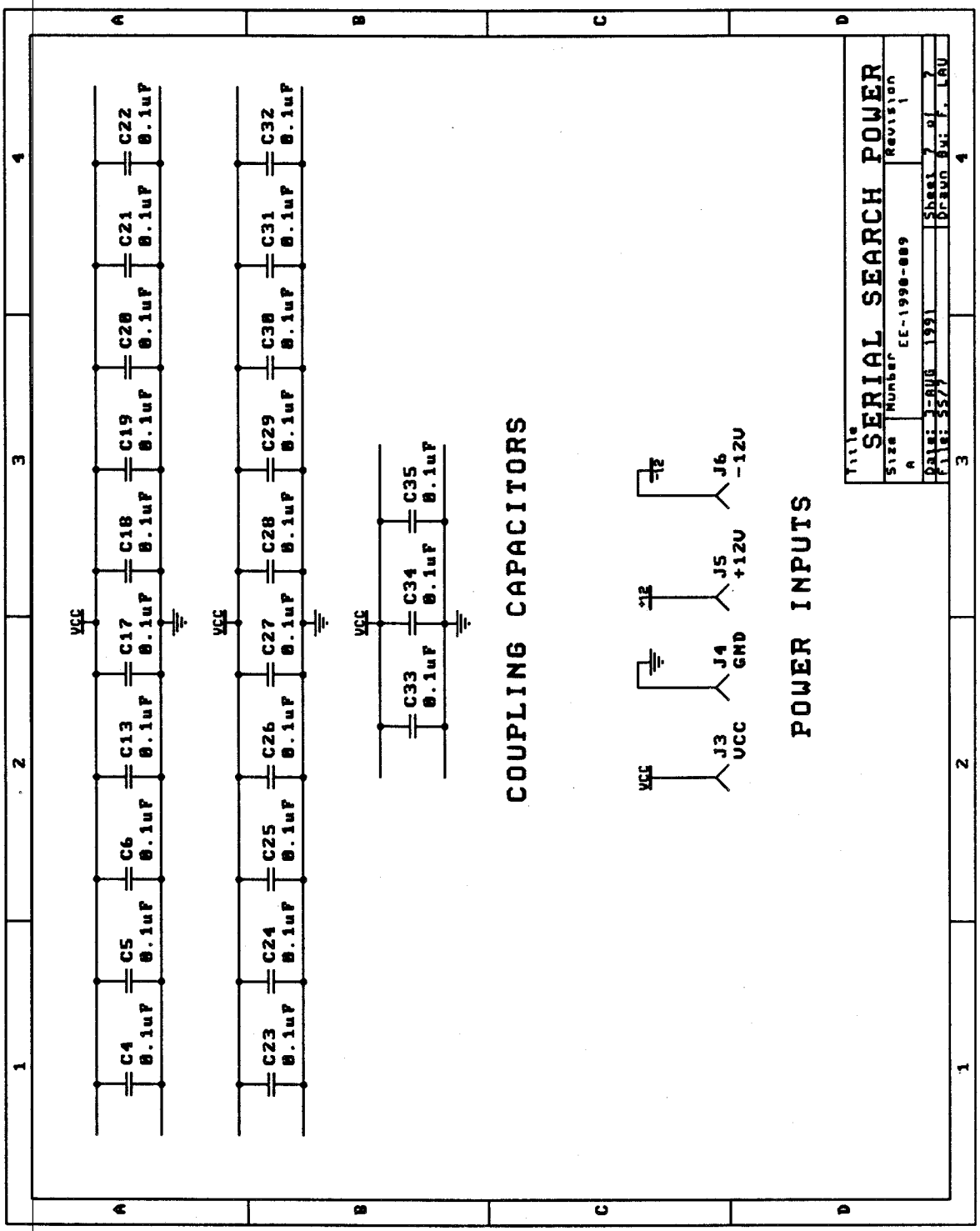

SQUARE LAW DEVICE
INTEGRATE-AND-DUMP FILTER
IC6 AD532
IC5:A TL074
IC5:B TL074
IC5:C TL074
IC7:A 7432
T1 IFRD110
C8 0.1uF
C7 0.1uF
C10 0.1uF
C11 0.1uF
C9 0.18uF
C12 0.22uF
R3 10K
R4 1K
R5 51
R6 680
R7 1K
R8 10K
R9 1K
R10 2.4K
R11 15K
R12 10K
R13 100
fo=1200Hz
RC TIME CONSTANT=Td1
TRIGGER BAR
TIME OUT
IAD OUTPUT
IQ DETECT
Title ENERGY DETECTOR
Size A
Number EE-1990-004
Revision 10
Date: 8-AUG 1991
File: SS72
Sheet 2 of 2
Drawn By: F. LAU
Note: This Energy Detector is used for both the serial search and the RAATS acquisition system.


DETECTION THRESHOLD
SELECTOR AND COMPARATOR
THRESHOLD CONTROLLER
COMP_OUT
IC9:A
LM393N
IC8
4052
IC5:D
TL074
IC10:B
7404
T2
2N3904
R14
1K
R15
5.1K
R16
10K
R17
2K
R18
10K
R19
10K
P1
5K
P2
5K
C14
0.1uF
C15
0.1uF
C16
0.1uF
VCC
GND
VEE
IND_OUTPUT
IO_SEL


SUL CONTROLLER
Title
SUL CONTROLLER
Size
A
Number
EE-1990-006
Revision
7
Date: 3-AUG 1991
File: 55/4
Sheet 4 of 7
Drawn By: F. LAU
HI=TD1 AND STEP
1/2 CHIP
LOW=TD2
TD_SEL
SEARCH
R24 128
D5 U
R23 128
D1 S
R22 128
D2 L1
R21 128
D3 PHASE LOCK
R20 128
D4 L2
VCC
IC10:E 7404
IC17:A 74LS10
IC17:B 74LS10
IC16:A 74LS20
IC13 74LS138
IC15:C 7408
IC11 74LS157
IC12 74LS175
IC14 74LS157
COMP_OUT
TD2_TIME_OUT
TD1_TIME_OUT
TIME_OUT
L_RESET


DWELL TIME GENERATOR
RESET DEBOUNCER
DWELL TIME CONTROLLER
Title
Size
Number
Revision
Date: 6-AUG 1991
File: SS7S
Sheet 5 of 7
Drawn By: F. LAU
Note: This Dwell Time Controller is used for both the serial search and the RAATS acquisition system.
TD1_TIME_OUT
(Not used in RAATS)
TD2_TIME_OUT
IC24:B 7408
IC21:F 7404
IC21:D 7404
IC21:C 7404
IC21:E 7404
IC22 74LS163
IC23:B 7402
IC23:C 7402
H_RESET
L_RESET
R26 1K
R25 1K
START
RESET
SW1
J7 PB0
J8 PB3
CONNECT TO VCC WHEN NOT USING COMPUTER
IC20 74LS30
IC19 74LS163
IC18 74LS163
TRIGGER
PN_CLK
VCC

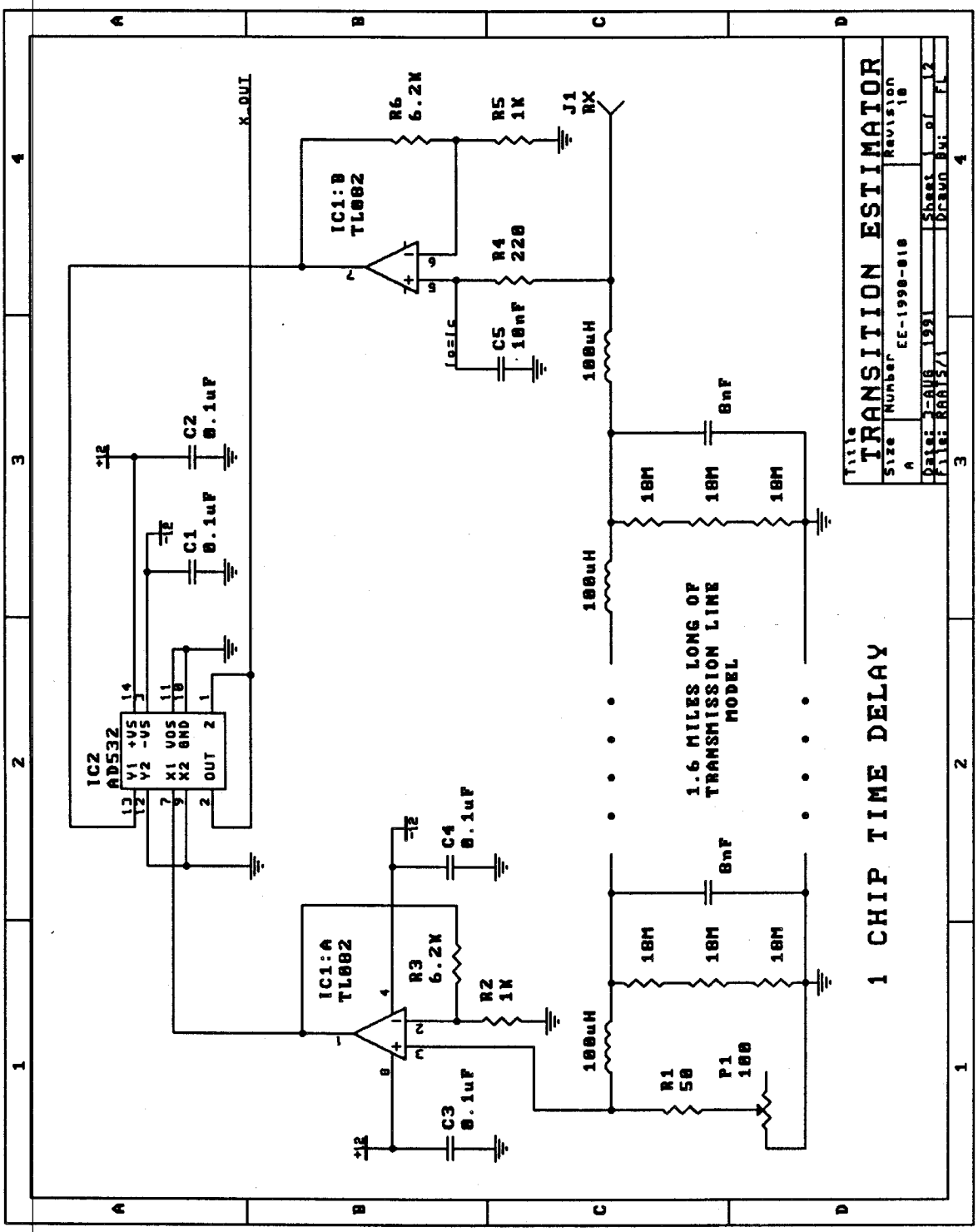

CLOCK GENERATOR
Size A
Number EE-1990-008
Revision 6
Date: 3-AUG 1991
File: SS/6
Sheet 6 of 7
Drawn By: F. LAU
IC25 LM311
IC26 74LS92
IC28:A 74LS74
IC28:B 74LS74
IC29:B 74LS74
IC15:A 7400
IC27:B 7411
IC2:B 7486
IC2:D 7486
IC10:C 7404
IC10:D 7404
R27 680
R28 2.2K
R29 1K
R30 18K
R31 1K
R32 2.2K
R33 470
R34 1.3K
C36 4.7nF
C37 0.1uF
C38 0.1uF
C39 1nF
J2 SYS_CLOCK (921.6kHz)
VCC
-12
+12
RX
SYS_CLK
H_RESET
L_RESET
TRIGGER_BAR
TRIGGER
PH_CLK
SYS_CLK/12
ID_SEL
SEARCH
ID1_TIME_OUT
CONNECT TO PIN 6 AND 7 OF IC26 WHEN NOT USING COMPUTER

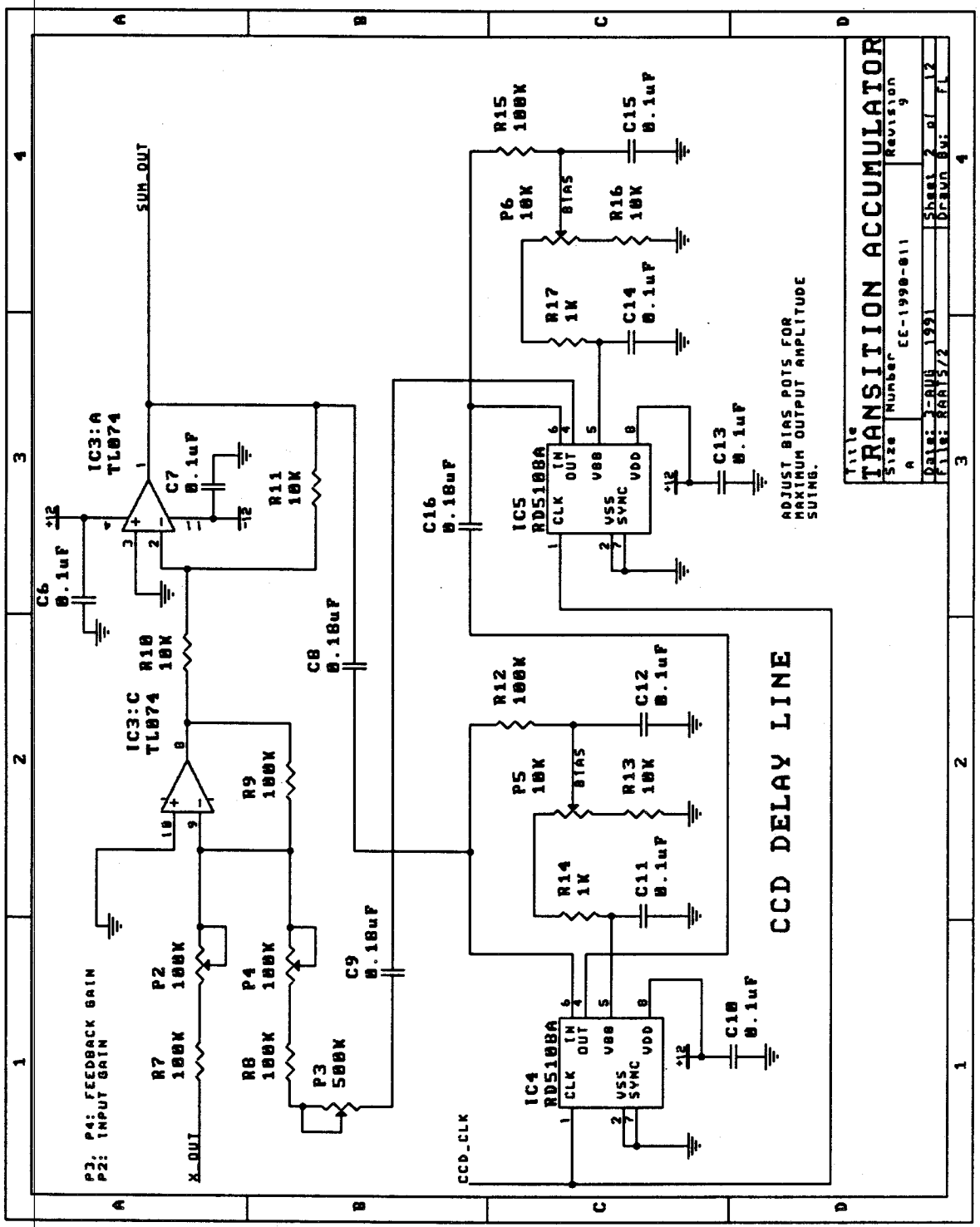

SERIAL SEARCH POWER
Size A
Number EE-1990-089
Revision 1
Date: 3-AUG 1991
Sheet 2 of 2
File: SS2
Drawn By: F. LAU
COUPLING CAPACITORS
POWER INPUTS
C4 0.1uF
C5 0.1uF
C6 0.1uF
C13 0.1uF
C17 0.1uF
C18 0.1uF
C19 0.1uF
C20 0.1uF
C21 0.1uF
C22 0.1uF
C23 0.1uF
C24 0.1uF
C25 0.1uF
C26 0.1uF
C27 0.1uF
C28 0.1uF
C29 0.1uF
C30 0.1uF
C31 0.1uF
C32 0.1uF
C33 0.1uF
C34 0.1uF
C35 0.1uF
VCC
J3 VCC
J4 GND
J5 +12V
J6 -12V

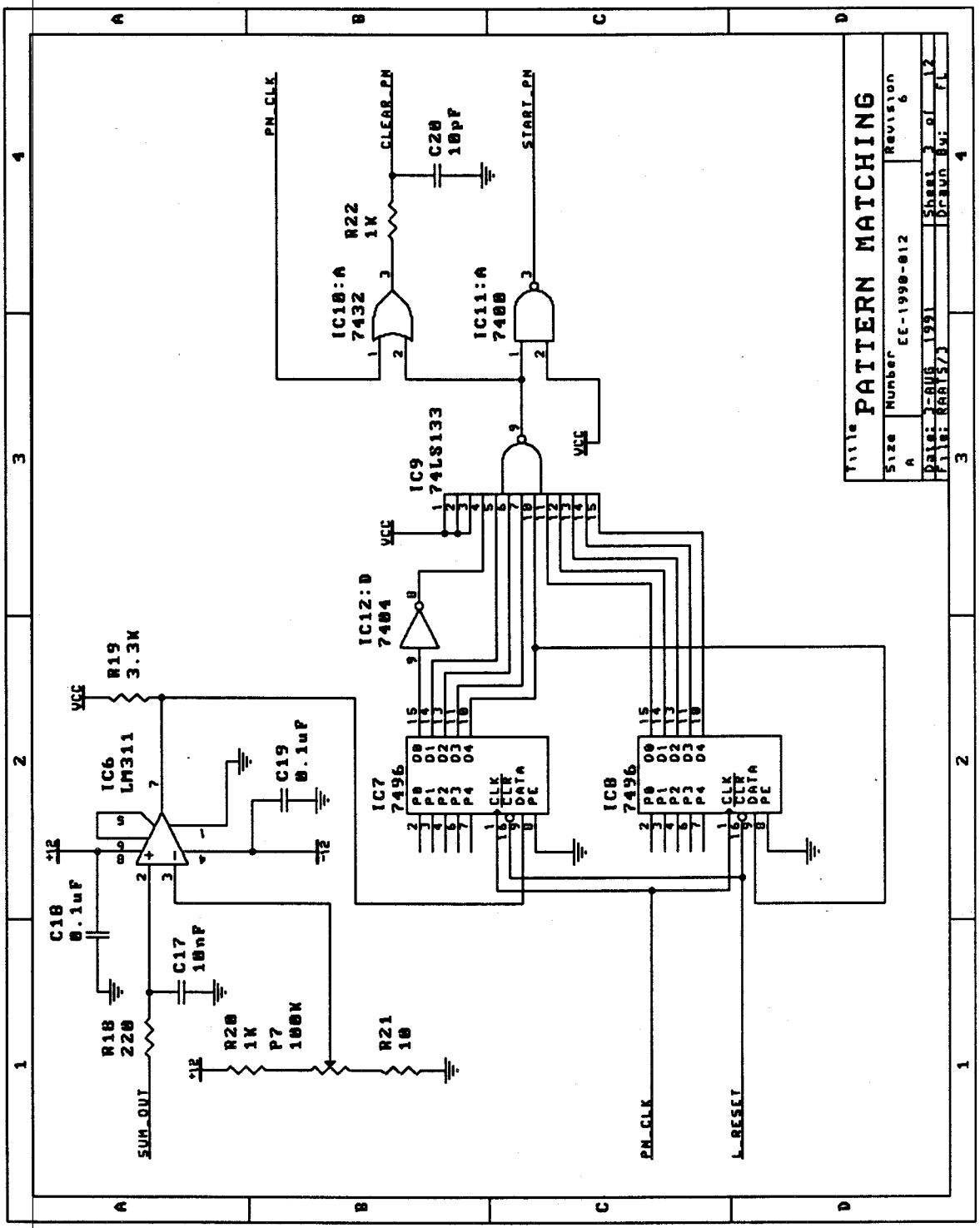
APPENDIX C

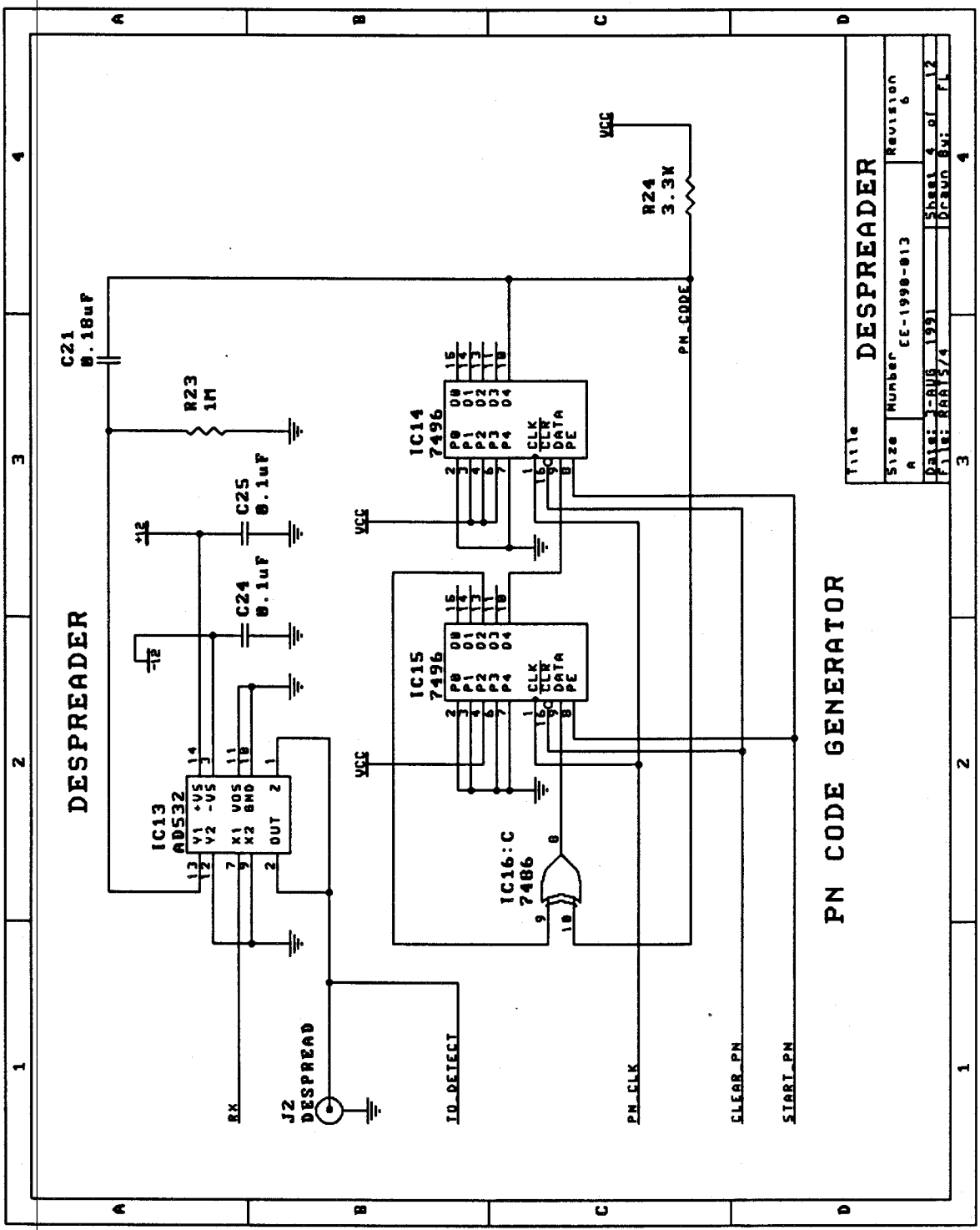
RAATS ACQUISITION SYSTEM SCHEMATIC

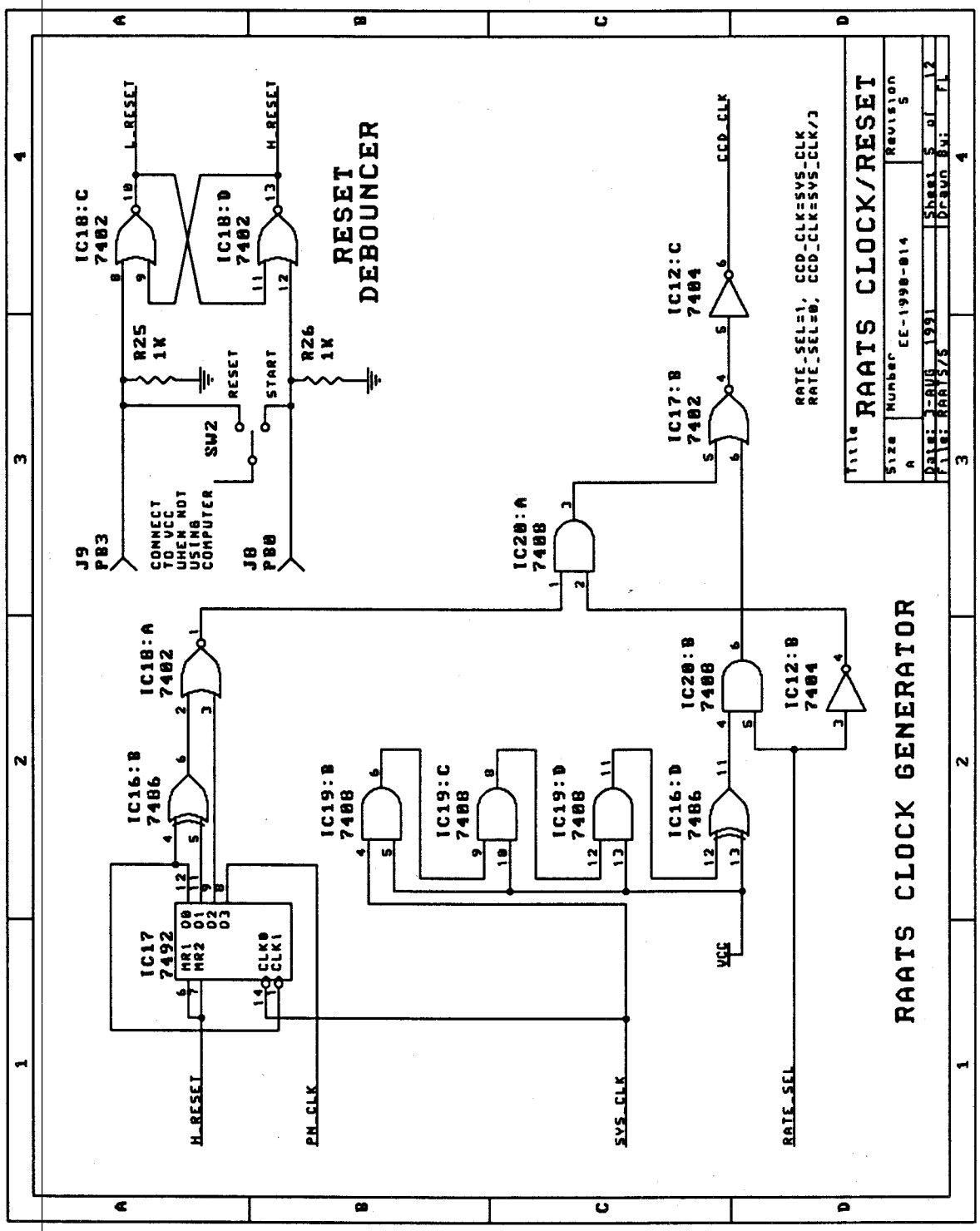
Note: The Energy Detector and the Dwell Time Controller used in the RAATS acquisition system are the same one used in the serial search system (see pages 111 and 114).

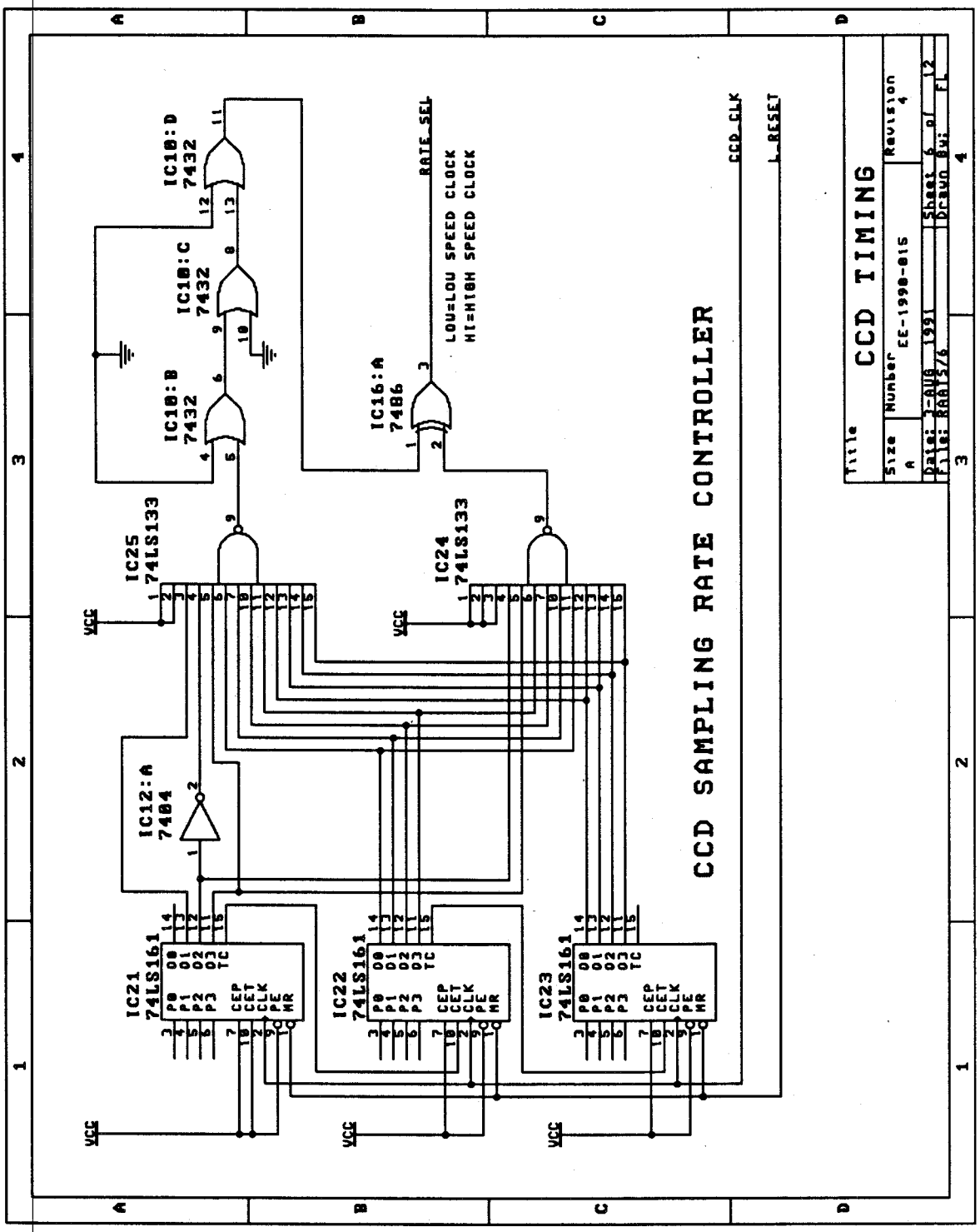

1 CHIP TIME DELAY
TRANSITION ESTIMATOR
1.6 MILES LONG OF TRANSMISSION LINE MODEL
IC1:A TL082
IC1:B TL082
IC2 AD532
R1 50
P1 100
R2 1K
R3 6.2K
R4 220
R5 1K
R6 6.2K
C1 0.1uF
C2 0.1uF
C3 0.1uF
C4 0.1uF
C5 10nF
8nF
18M
100uH
J1 RX
X-OUT
Logic

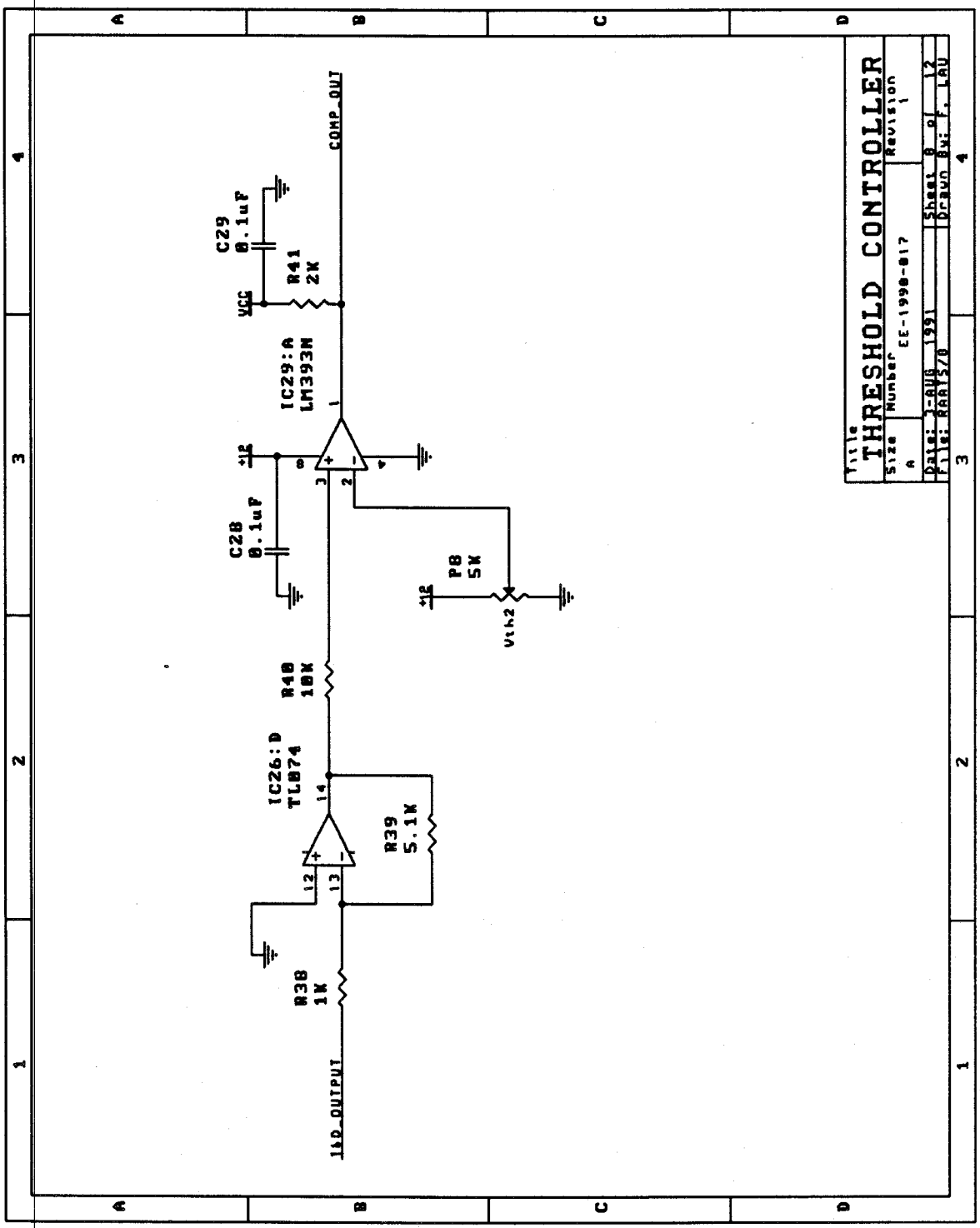

CCD DELAY LINE
TRANSITION ACCUMULATOR
SUM_OUT
X_OUT
CCD_CLK
IC3:A TL074
IC3:C TL074
IC4 RD5108A
IC5 RD5108A
P2: P4: FEEDBACK GAIN
P2: INPUT GAIN
ADJUST BIAS POTS FOR MAXIMUM OUTPUT AMPLITUDE SWING.
BIAS
CLK
IN
OUT
VSS
SYNC
VDD
+12
-12

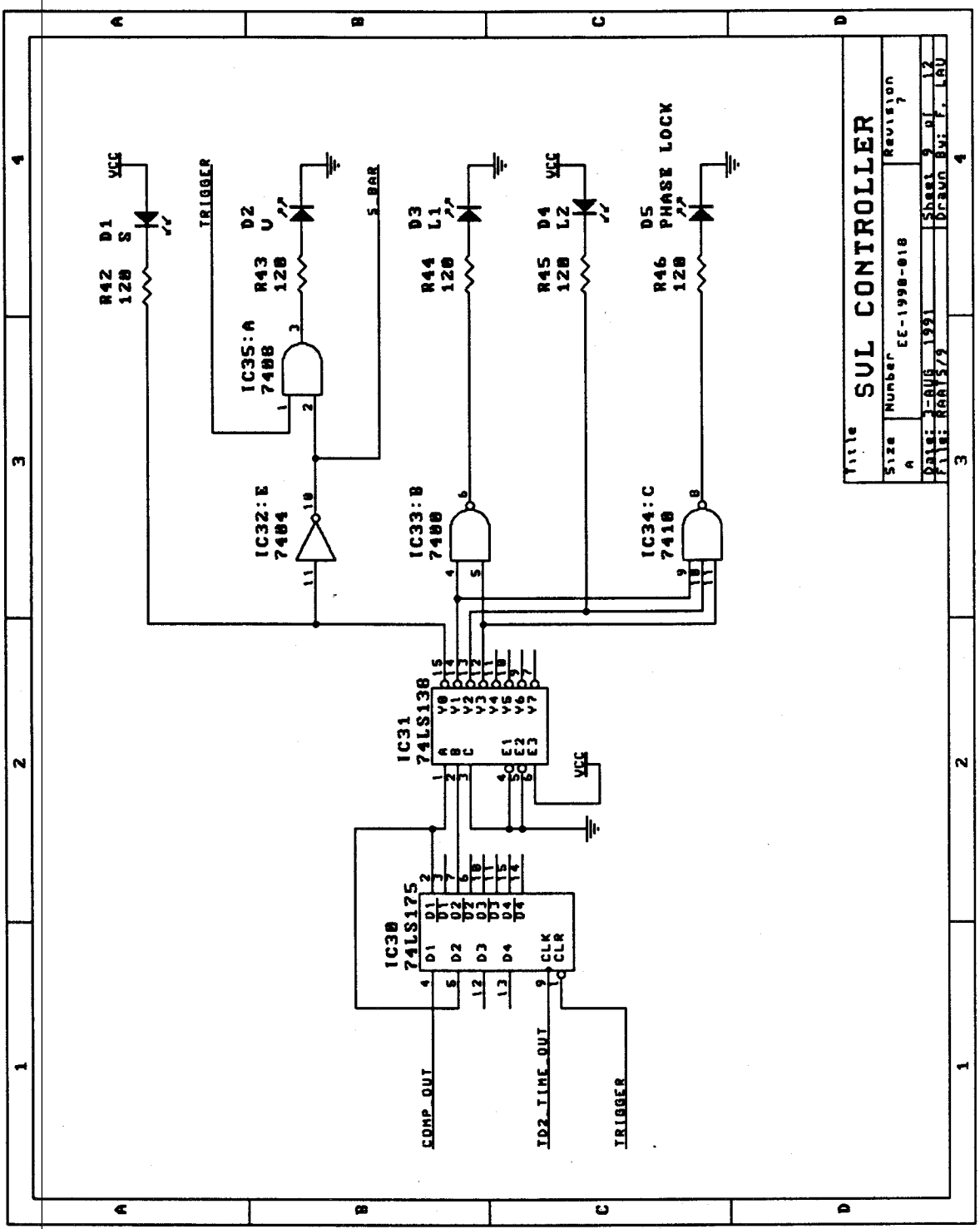

PATTERN MATCHING
Size A
Number EE-1990-012
Revision 6
Date: 3-AUG 1991
Sheet 2 of 12
File: RAA1S7.3
Drawn By: FL
PN_CLK
CLEAR_PN
START_PN
C20 10pF
R22 1K
IC10:A 7432
IC11:A 7408
IC9 74LS133
VCC
IC12:D 7404
R19 3.3K
IC6 LM311
C19 0.1uF
IC7 7496
IC8 7496
P0 P1 P2 P3 P4
CLK CLR DATA PE
+12
C18 0.1uF
C17 10nF
R18 220
R20 1K
P7 100K
R21 10
SUM_OUT
L_RESET

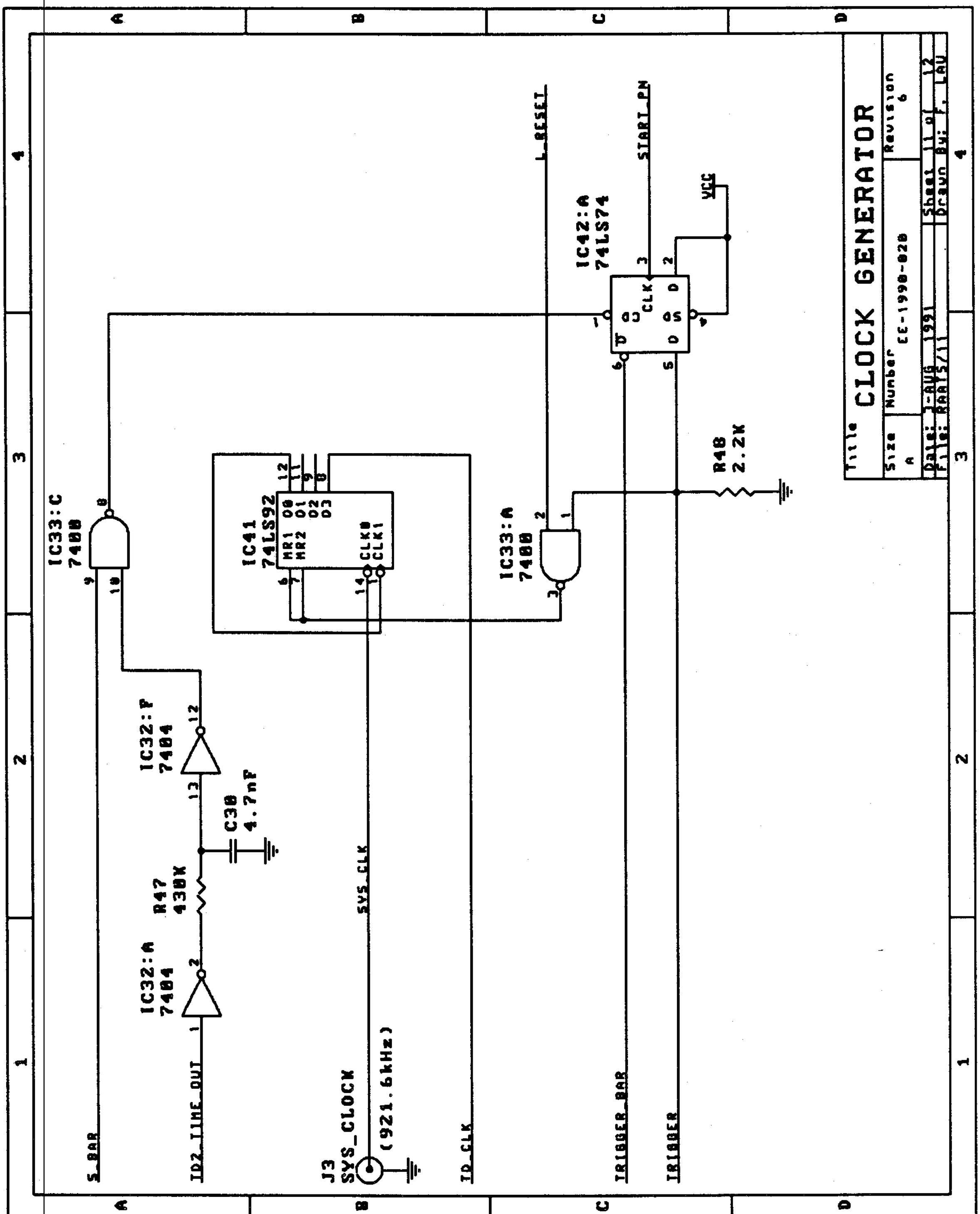

DESPREADER
PN CODE GENERATOR
C21
0.18uF
R23
1M
C25
0.1uF
C24
0.1uF
+12
-12
IC13
AD532
IC14
7496
IC15
7496
IC16:C
7486
R24
3.3K
VCC
PN_CODE
J2
DESPREAD
RX
IO_DETECT
PN_CLK
CLEAR_PN
START_PN
Title
DESPREADER
Size
A
Number
EE-1990-013
Revision
6
Date: 3-AUG 1991
Sheet 4 of 12
File: RAATS/4
Drawn By:

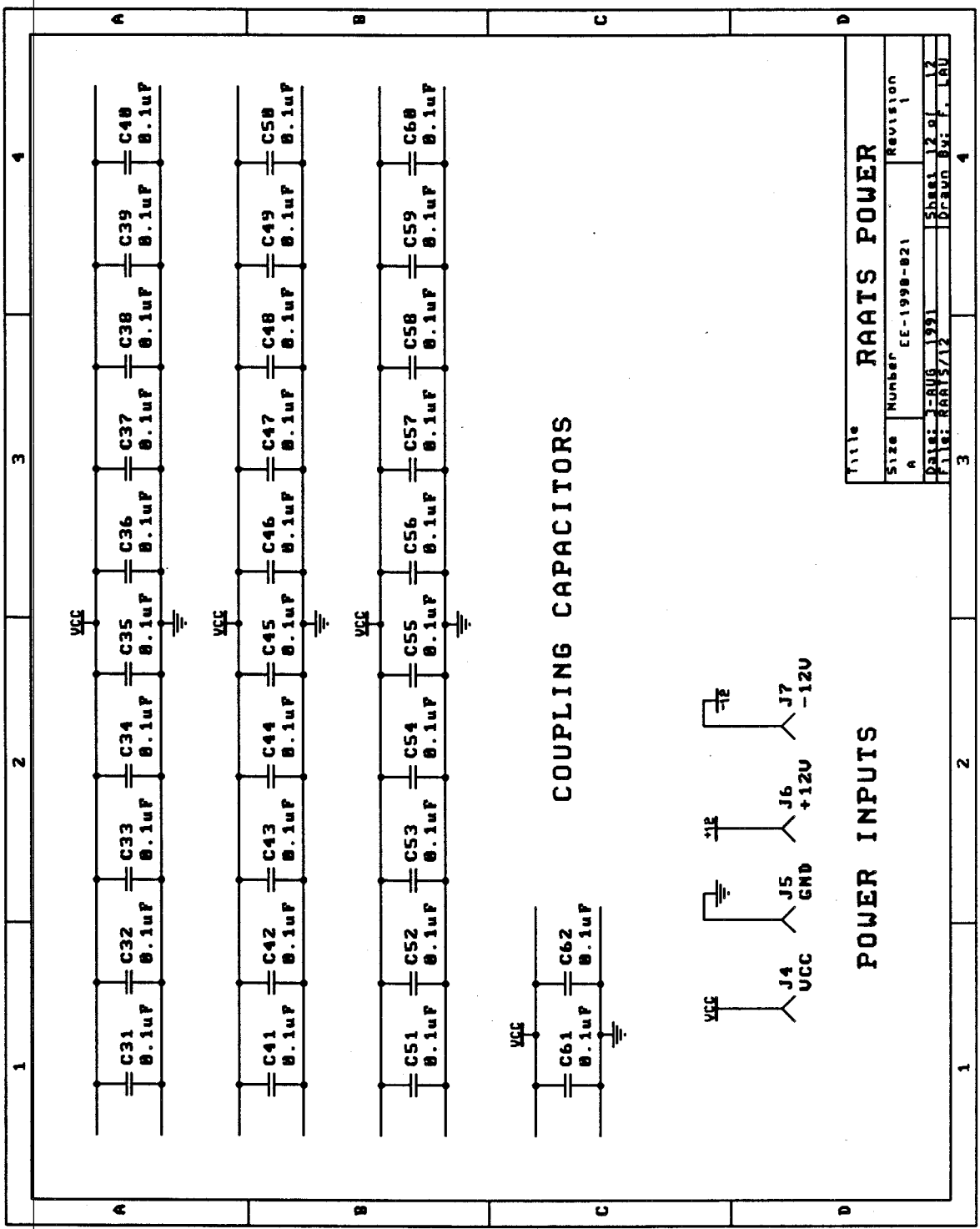

RAATS CLOCK GENERATOR
RAATS CLOCK/RESET
RESET DEBOUNCER
IC18:C 7402
IC18:D 7402
IC18:A 7402
IC16:B 7486
IC17 7492
IC19:B 7408
IC19:C 7408
IC19:D 7408
IC16:D 7486
IC20:B 7408
IC12:B 7404
IC20:A 7408
IC17:B 7402
IC12:C 7404
R25 1K
R26 1K
SW2
J9 PB3
J8 PB0
CONNECT TO VCC WHEN NOT USING COMPUTER
L_RESET
H_RESET
RESET
START
CCD_CLK
RATE_SEL=1; CCD_CLK=SYS_CLK
RATE_SEL=0; CCD_CLK=SYS_CLK/2
PM_CLK
SYS_CLK
RATE_SEL
VCC


CCD SAMPLING RATE CONTROLLER
CCD TIMING
IC10:D 7432
IC10:C 7432
IC10:B 7432
IC16:A 7486
IC12:A 7404
IC25 74LS133
IC24 74LS133
IC21 74LS161
IC22 74LS161
IC23 74LS161
RATE_SEL
LOU=LOU SPEED CLOCK
HI=HIGH SPEED CLOCK
CCD_CLK
L_RESET
VCC
Title
Size A
Number EE-1998-815
Revision 4
Date: 2-AUG 1991
Sheet 6 of 12
File: RADI5/6
Drawn By: FL


THRESHOLD CONTROLLER
Size
A
Number
EE-1998-017
Revision
1
Date: 3-AUG 1991
File: RAA1270
Sheet 8 of 12
Drawn By: F. LAU
COMP_OUT
C29
0.1uF
R41
2K
VCC
IC29:A
LM393N
+12
C28
0.1uF
P8
5K
+12
Vth2
R40
10K
IC26:D
TL074
R39
5.1K
R38
1K
1kΩ_OUTPUT


SUL CONTROLLER
Number EE-1998-018
Revision 1
Size A
Date: 7-AUG 1991
File: RAA15/9
Sheet 2 of 12
Drawn By: F. LAU
VCC
D1 S
R42 120
TRIGGER
D2 U
R43 120
S_BAR
IC35:A 7408
IC32:E 7404
D3 L1
R44 120
IC33:B 7408
D4 L2
R45 120
D5 PHASE LOCK
R46 120
IC34:C 7410
IC31 74LS138
IC30 74LS175
COMP_OUT
ID2_TIME_OUT
TRIGGER


CLOCK GENERATOR
Title
Size A
Number CE-1990-028
Revision 6
Date: 3-AUG-1991
Sheet 11 of 12
File: RAAIS/11
Drawn By: F. LAU
IC33:C 7400
IC41 74LS92
IC42:A 74LS74
IC33:A 7400
R48 2.2K
IC32:F 7404
C30 4.7nF
R47 430K
IC32:A 7404
SYS_CLK
J3 SYS_CLOCK (921.6kHz)
S_BAR
ID2_TIME_OUT
ID_CLK
TRIGGER_BAR
TRIGGER
L_RESET
START_PN
VCC


COUPLING CAPACITORS
POWER INPUTS
RAATS POWER
Title
Size A
Number EE-1998-021
Revision 1
Date: 1-AUG 1991
File: RAATS/12
Sheet 12 of 12
Drawn By: F. LAU
VCC
J4 VCC
J5 GND
J6 +12V
J7 -12V
C31 0.1uF
C32 0.1uF
C33 0.1uF
C34 0.1uF
C35 0.1uF
C36 0.1uF
C37 0.1uF
C38 0.1uF
C39 0.1uF
C40 0.1uF
C41 0.1uF
C42 0.1uF
C43 0.1uF
C44 0.1uF
C45 0.1uF
C46 0.1uF
C47 0.1uF
C48 0.1uF
C49 0.1uF
C50 0.1uF
C51 0.1uF
C52 0.1uF
C53 0.1uF
C54 0.1uF
C55 0.1uF
C56 0.1uF
C57 0.1uF
C58 0.1uF
C59 0.1uF
C60 0.1uF
C61 0.1uF
C62 0.1uF

APPENDIX D

ACQT SOURCE CODE

```
1 rem ACQT by Floyd L.F. Lau,  August 28, 1991
3 dim b(240)
4 for n=1 to (4*60):rem initialize acquisition time array
6 b(n)=0
8 next n
10 print chr$(147):rem clear screen
12 print "[1] threshold tests":print "[2] acquisition time tests"
13 print:print "enter your choice:"
14 get ch$
15 if (ch$<>"1") and (ch$<>"2") then goto 14
20 print chr$(147)
22 input "how often do you want to backup test data on disk ";bak
24 bc=1:rem initialize backup counter
30 print chr$(147)
31 input "signal level (rms)";signal:print
32 input "noise level (rms)";noise:print
33 input "low threshold voltage (volts)";lv:print
34 input "high threshold voltage (volts)";hv:print
35 input "number of data points";dp:print
40 poke 59459, 127:rem set ddr a
45 poke 59471, 24:rem reset all pcb
50 print:print "press any key when ready"
56 time$="000000":rem reset internal clock
57 get a$
58 if a$="" then goto 57
59 print chr$(147):print "tests begin"
60 for c=1 to dp
65 if ch$="2" then gosub 1000
90 s=time:rem record starting time
100 poke 59471, 3:rem start both modulator and demodulator
105 if peek(59471)<128 then goto 105:rem look for LOCK signal
110 for x=1 to 29:rem check duration of LOCK signal (1 second)
116 y=time
117 if time<>y+1 then 117
```

```
118 if peek(59471)<128 then goto 105
119 next x
120 acqt=(int(((time-s)/60-1)*10000))/10000:rem calculate acquisition time
122 print "test no.";c,"acquisition time =";acqt
124 for j=1 to (4*60):rem update acquisition time array
126 if (acqt>=(j-1)*0.25) and (acqt<j*0.25) then b(j)=b(j)+1
127 next j
130 poke 59471, 24:rem reset all pcb
132 if c=bc*bak then gosub 2000
134 next c
140 open 1,4:rem send results to printer
141 print#1, "signal level =";signal;" volts rms"
142 print#1, "noise level =";noise;" volts rms"
143 print#1, "low threshold voltage =";lv;" volts"
144 print#1, "high threshold voltage =";hv;" volts"
145 print#1, "total number of tests preformed =";c-1
146 print#1:print#1
147 print#1, "acquisition time (seconds):"
150 for m=1 to (4*60)
160 print#1, (m-1)*0.25,"to",m*0.25-0.01,b(m)
180 next m
190 close 1
200 print:print "total run time: ";time$
999 end
1000 poke 59471, 17:rem start demodulator
1010 for i=1 to int(rnd(0)*1000+0.5):next
1020 return
2000 print:print "saving data on floppy disk, please wait...":print
2008 scratch "testdata":rem in case subroutine is rerun
2010 if ds>1 then print ds$:stop:rem check for disk error
2020 dopen#1, "testdata",w:rem open data file
2030 if ds>1 then print ds$:stop
2032 print#1, signal
2033 print#1, noise
2034 print#1, lv
2035 print#1, hv
2036 print#1, c
2040 for k=1 to (4*60)
2050 print#1, b(k)
2060 next k
2070 if ds>1 then print ds$:stop
2080 dclose
2090 if ds>1 then print ds$:stop
2100 bc=bc+1:rem update backup counter
2110 print "continue...":print
2120 return
```

APPENDIX E

RECALL SOURCE CODE

```
1 rem RECALL by Floyd L.F. Lau
2 rem August 28, 1991
15 dim b(245)
20 dopen#1, "testdata":rem open data file
25 if ds>1 then print ds$:stop:rem check for disk error
30 for c=1 to 5+(4*60):rem recall test information and acquisition time array
40 input#1, d
50 if ds>1 then print ds$:stop
55 b(c)=d
70 next c
80 dclose
90 if ds>1 then print ds$:stop
91 print chr$(147)
92 print "signal level =";b(1);" volts rms"
93 print "noise level =";b(2);" volts rms"
94 print "low threshold =";b(3);" volts"
95 print "high threshold =";b(4);" volts"
96 print "total number of tests preformed =";b(5):print
97 print "acquisition time (seconds):"
98 for n=6 to 5+(4*60):rem print acquisition time array
99 print ((n-5)-1)*0.25,"to",(n-5)*0.25-0.01,b(n)
100 next n
101 print:print:print "print results? (y/n)"
102 get q$
103 if (q$<>"y" and q$<>"n") then 102
104 if q$="n" then goto 999
105 rem send data to printer
110 open 1,4
112 print#1, "signal level =";b(1);" volts rms"
113 print#1, "noise level =";b(2);" volts rms"
114 print#1, "low threshold =";b(3);" volts"
115 print#1, "high threshold =";b(4);" volts"
116 print#1, "total number of tests preformed =";b(5)
117 print#1:print#1
120 print#1, "acquisition time (seconds):"
```

```
130 for m=6 to 5+(4*60)
140 print#1, ((m-5)-1)*0.25,"to",(m-5)*0.25-0.01,b(m)
150 next m
160 close 1
999 end
```

APPENDIX F

INTEGRATION VOLTAGE ENVELOPE MEASUREMENTS

Table F.1 to F.4 record the voltage envelopes measured at the integrate-and-dump filter output at -20 dB SNR. Ten measurements were taken for each of the four voltage envelopes with an oscilloscope and their mean were then calculated.

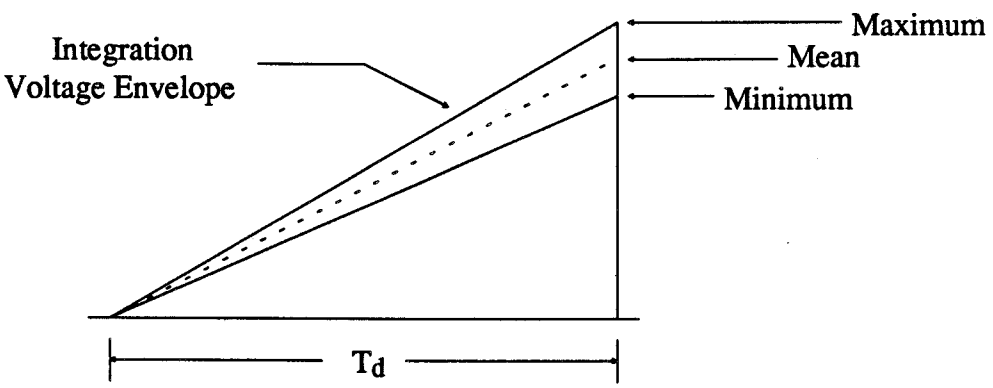


Fig. F.1 Notation Used in Measuring the Integrate-and-Dump Voltage Envelope

Table F.1 Integrate-and-Dump Filter Output Envelope A

	Envelope A		
	Minimum	Maximum	Mean
	1.00	1.60	1.30
	1.00	1.60	1.30
	1.00	1.40	1.20
	1.00	1.50	1.25
	1.00	1.60	1.30
	1.00	1.60	1.30
	1.00	1.60	1.30
	1.00	1.80	1.40
	1.00	1.60	1.30
	1.00	1.70	1.35
Average	1.00	1.60	1.30

Table F.2 Integrate-and-Dump Filter Output Envelope B

	Envelope B		
	Minimum	Maximum	Mean
	1.30	2.10	1.70
	1.30	2.00	1.65
	1.30	2.30	1.80
	1.25	2.00	1.63
	1.30	2.20	1.75
	1.20	1.90	1.55
	1.30	1.90	1.60
	1.20	2.30	1.75
	1.25	2.00	1.63
	1.40	2.45	1.93
Average	1.28	2.12	1.70

Table F.3 Integrate-and-Dump Filter Output Envelope C

	Envelope C		
	Minimum	Maximum	Mean
	6.00	7.20	6.60
	6.00	7.50	6.75
	6.20	7.50	6.85
	6.00	7.20	6.60
	5.90	7.00	6.75
	5.70	7.40	6.55
	6.10	7.20	6.65
	6.20	7.40	6.80
	6.20	7.40	6.80
	6.20	7.20	6.70
Average	6.05	7.30	6.68

Table F.4 Integrate-and-Dump Filter Output Envelope D

	Envelope D		
	Minimum	Maximum	Mean
	6.50	8.00	7.25
	6.40	8.00	7.20
	6.40	7.90	7.15
	6.40	8.00	7.20
	6.55	7.90	7.23
	6.60	8.20	7.40
	6.20	8.20	7.20
	6.60	8.30	7.45
	6.50	8.00	7.25
	6.50	8.50	7.50
Average	6.47	8.10	7.28

Table F.5 Calculated Threshold Voltages

	Vth1	Vth2
60% threshold	1.54	7.04
50% threshold	1.50	6.98
40% threshold	1.46	6.92
30% threshold	1.42	6.86
20% threshold	1.38	6.80

APPENDIX G

SERIAL SEARCH THRESHOLD TEST RESULTS

Note: Search had been started from code phase position furthest from the correct code phase.

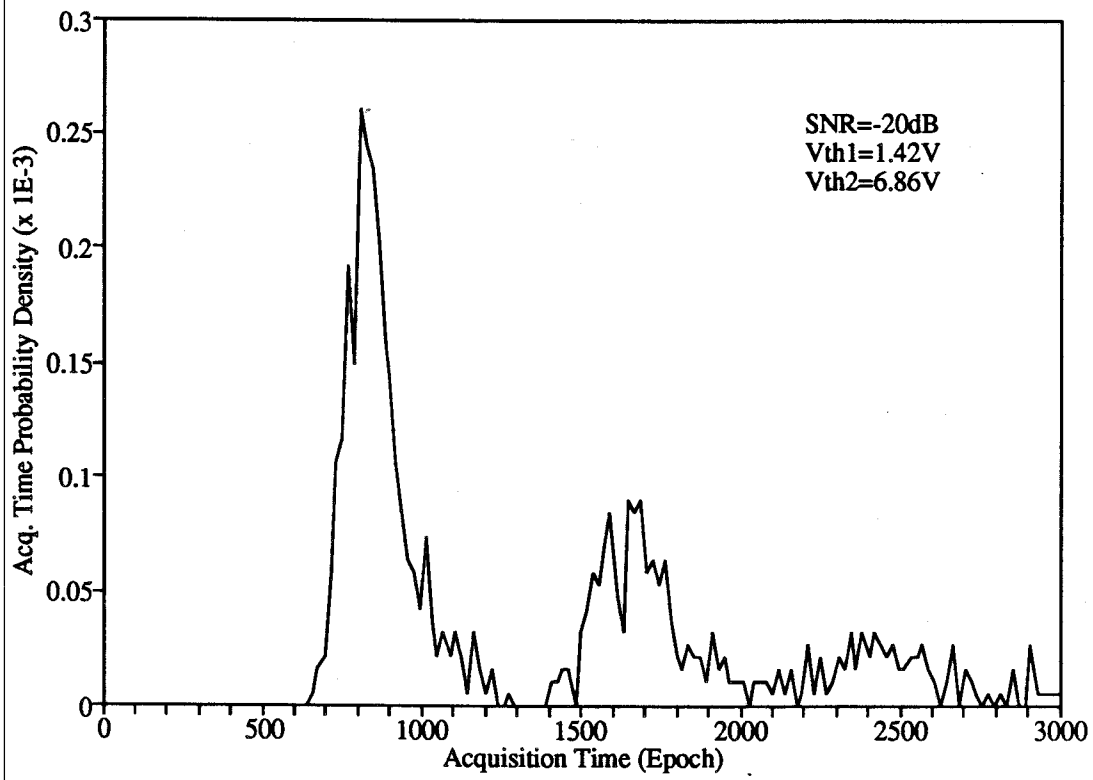


Figure G.1 30% Threshold Test

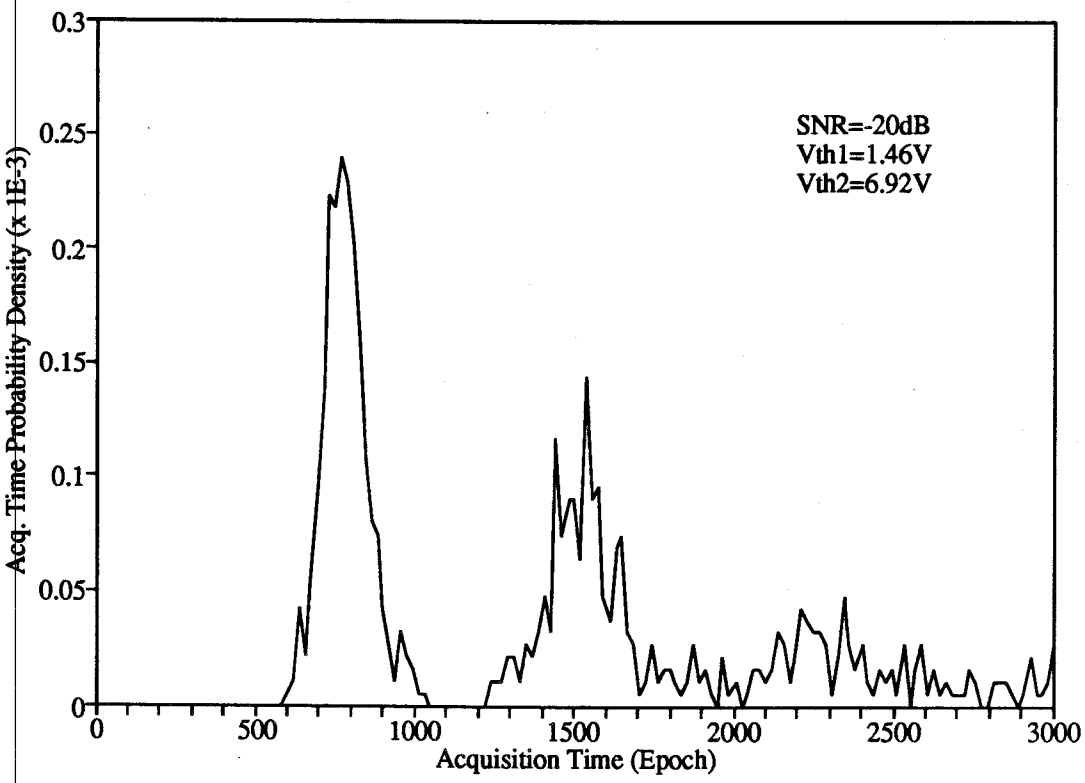


Figure G.2 40% Threshold Test

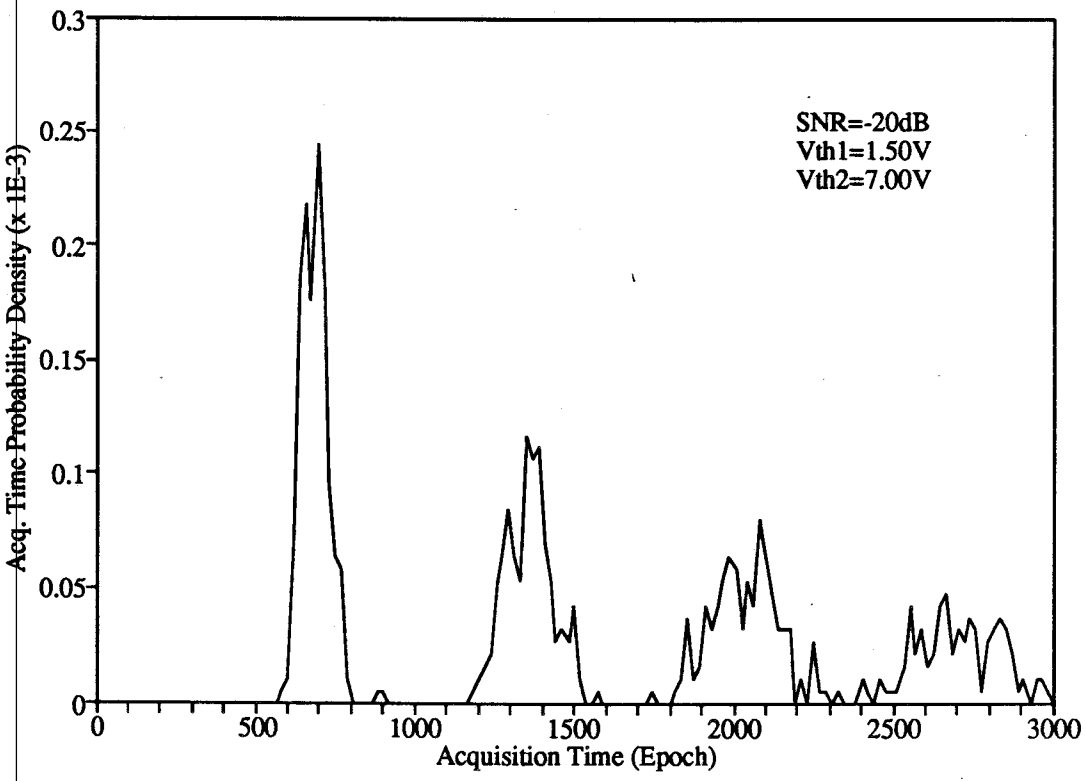


Figure G.3 50% Threshold Test

APPENDIX H

TRANSIM SOURCE CODE

```
/* TRANSIM by Floyd L. F. Lau */
/* August 28, 1991 */

#include <stdio.h>
#include <math.h>

int pn[1023];
float ccd[2046];
void accumulator(int epoch, float eff);
void output(void);

void main(void)
{
    int d[10], q[10], adder[10];
    int i, k, n, s, t, u;
    int j;              /* Use 'unsigned long' for long code generator */
    int mod, cycle, temp;
    float cti;
    char input[2];
    int length=10;    /* Define length of PN code generator */

/* Acquire simulation parameters */
    printf ("Number of Modulo-2 adder: ");
    gets (input);
    sscanf (input, "%d", &mod);
    printf ("\n");

    printf ("Enter PN code generator:\n");
    for (n=0; n<=mod-1; n++)
    {
        gets (input);
        sscanf (input, "%d", &adder[n]);
    }
    printf ("\n");
```

```
    printf ("Number of accumulation cycle: ");
    gets (input);
    sscanf (input, "%d", &cycle);
    printf ("\n");

    printf ("Charge Transfer Inefficiency (%): ");
    gets (input);
    sscanf (input, "%f", &cti);
    printf ("\n");

/* Initialize PN code generator */
    for (i=0; i<=length-1; i++)
    {
        d[i]=q[i]=1;
    }

/* Generate PN code sequence */
    for (j=1; j<=pow(2, length)-1; j++)
    {
        d[0]=q[adder[0]-1] ^ q[adder[1]-1];
        for (s=2; s<=mod-1; s++)
        {
            d[0]=d[0] ^ q[adder[s]-1];
        }
        for (k=0; k<=length-1; k++)
        {
            q[k]=d[k];
            printf ("%d", q[k]);          /* Print PN code sequence */
        }

        printf ("\n");
        for (u=0; u<=length-1-1; u++)
        {
            d[u+1]=q[u];
        }

        if (q[length-1]==0) pn[j-1]=-1;
        else pn[j-1]=1;
    }
    accumulator(cycle, cti);
}

void accumulator(int epoch, float ti)
{
    int ring1[1023], ring2[1023], delay[1023];
    int i, j, m, a, q, r, s;
```

```
    float Gfw=0.1;
    float Gl=0.9;
    float temp;

/* Initialize transition accumulator */
    for (r=0; r<=2045; r++)
    {
        ccd[r]=0.0;
    }

    for (i=0; i<=1022; i++)
    {
        ring1[i]=pn[i];
        if (i==0) ring2[i]=pn[1022];
        else ring2[i]=pn[i-1];
    }

/* Perform transition estimation */
    for (j=0; j<=1022; j++)
    {
        delay[j]=ring1[j]*ring2[j];
    }

/* Start transition accumulation */
    for (q=0; q<=epoch; q++)
    {
        for (m=0; m<=1022; m++)
        {
            for (s=0; s<=1; s++)
            {
                temp=ccd[0];
                ccd[0]=delay[m]*Gfw+ccd[2045]*Gl+ccd[0]*(ti/100.0);
                for (a=2045; a>=2; a--)
                {
                    ccd[a]=ccd[a]*(ti/100.0)+ccd[a-1]*((100.0-ti)/100.0);
                }
                ccd[1]=ccd[1]*(ti/100.0)+temp*((100.0-ti)/100.0);
            }
        }
        if (q>=epoch-2) output();
    }
}

void output(void)
{
    int k;
```

```
    for (k=2045; k>=0; k--)
    {
        printf ("%6.3f\n", ccd[k]);
    }
}
```