Performance Parameters Related to the Synchronization of Clocked Systems

Authors: Willie Yaw-Poh Lim

The effect of nondeterministic resolution times of flip-flops, used for synchronizing externally generated input events, on the reliability of fixed period and a newly introduced variable period clocked system is investigated. The failure probability of the simplest form of the two system types of compared. It is shown that in general, clocked system failure probability can be expressed as the product of three parameters representing, respectively, the effect of input arrival processes, system structures, and nondeterministic flip-flop resolution times.

Follow this and additional works at: https://openscholarship.wustl.edu/cse_research

Recommended Citation
Performance Parameters Related to the Synchronization of Clocked Systems

Willie Yaw-Poh Lim

WUCS-80-01

August 1979

Department of Computer Science
Washington University
Campus Box 1045
One Brookings Drive
Saint Louis, MO 63130-4899

A thesis presented to the Sever Institute of Washington University in partial fulfillment of the requirements for the degree of Master of Science.
WASHINGTON UNIVERSITY
SEVER INSTITUTE OF TECHNOLOGY

ABSTRACT

PERFORMANCE PARAMETERS RELATED TO THE
SYNCHRONIZATION OF CLOCKED SYSTEMS

by Willie Yaw-Poh Lim

ADVISOR: Professor J. R. Cox, Jr.

August, 1979
Saint Louis, Missouri

The effect of nondeterministic resolution times of flip-flops, used for synchronizing externally generated input events, on the reliability of fixed period and a newly introduced variable period clocked system is investigated. The failure probability of the simplest form of the two system types is compared. It is shown that in general, clocked system failure probability can be expressed as the product of three parameters representing, respectively, the effect of input arrival processes, system structure, and, nondeterministic flip-flop resolution times.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>No.</th>
<th>Section Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1</td>
<td>Objective of the Thesis</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>Communication between a Computer System and its Environment</td>
<td>2</td>
</tr>
<tr>
<td>1.3</td>
<td>Anomalous System Behavior Resulting from Synchronizer Failure</td>
<td>4</td>
</tr>
<tr>
<td>1.4</td>
<td>The Syncro Clock - A method to avoid Anomalous System Behavior</td>
<td>5</td>
</tr>
<tr>
<td>1.5</td>
<td>Scope and Organization of the Thesis</td>
<td>6</td>
</tr>
<tr>
<td>2.</td>
<td>A Model for Clocked Systems</td>
<td>9</td>
</tr>
<tr>
<td>2.1</td>
<td>Introduction</td>
<td>9</td>
</tr>
<tr>
<td>2.2</td>
<td>Modelling the Anomalous Behavior of the Bistable Device</td>
<td>9</td>
</tr>
<tr>
<td>2.3</td>
<td>The M Flip-Flop - A Generalized Model for the Clocked Flip-Flop</td>
<td>14</td>
</tr>
<tr>
<td>2.4</td>
<td>System Clocks - Fixed Period and Syncro Clocks</td>
<td>16</td>
</tr>
<tr>
<td>2.5</td>
<td>Modules and Clocked Systems</td>
<td>20</td>
</tr>
<tr>
<td>2.6</td>
<td>Communication between a Module and its Environment</td>
<td>24</td>
</tr>
<tr>
<td>3.</td>
<td>The Truncated Exponential Density Function</td>
<td>25</td>
</tr>
<tr>
<td>3.1</td>
<td>Introduction</td>
<td>25</td>
</tr>
<tr>
<td>3.2</td>
<td>Some Elementary Properties of the TED Function</td>
<td>30</td>
</tr>
<tr>
<td>3.3</td>
<td>The N-fold Convolution of the TED Function</td>
<td>32</td>
</tr>
<tr>
<td>3.4</td>
<td>Further Properties of the Truncated Exponential Density Function</td>
<td>35</td>
</tr>
<tr>
<td>3.5</td>
<td>Using the Approximate Function for the N-fold Convolution</td>
<td>36</td>
</tr>
<tr>
<td>No.</td>
<td>Table of Contents</td>
<td>Page</td>
</tr>
<tr>
<td>-----</td>
<td>----------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4.</td>
<td>Performance of Fixed Period and Synoro Clocked Systems</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>4.1 Fault Condition Arising from Marginal Triggering</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>4.2 Effect of a Glitch Hazard on Module Failure</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>4.3 Basic Assumptions and Notations</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>4.4 Performance Analysis of FPC Modules</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>4.5 Performance Analysis of Synoro Clocked Modules</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>4.6 Comparing FPC and Synoro Clocked Modules</td>
<td>49</td>
</tr>
<tr>
<td>5.</td>
<td>Characterization of Performance of Clocked Systems</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>5.1 Introduction</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>5.2 Role of a Module in a Clocked System</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>5.3 Performance Measure for Clocked Systems</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>5.4 Assumptions and Notations for Analysis of Clocked Systems</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>5.5 Probability of Failure of Fixed Period Clocked Systems</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>5.6 Probability of Failure of Synoro Clocked Systems</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td>5.7 Density Functions for Synoro Clocked Systems</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td>5.8 Parameters for Characterizing Performance of Clocked Systems</td>
<td>69</td>
</tr>
<tr>
<td>6.</td>
<td>Conclusion and Future Research</td>
<td>73</td>
</tr>
<tr>
<td></td>
<td>6.1 Introduction</td>
<td>73</td>
</tr>
<tr>
<td></td>
<td>6.2 Applications of the Results - Examples</td>
<td>73</td>
</tr>
<tr>
<td>No.</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Determining Maximum Resolution Time for FPC Systems</td>
<td>73</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Determining the Value of $T_s$ for Syncro Clocked Systems</td>
<td>78</td>
</tr>
<tr>
<td>6.3</td>
<td>Limitations of the Results</td>
<td>80</td>
</tr>
<tr>
<td>6.4</td>
<td>Conclusion</td>
<td>83</td>
</tr>
<tr>
<td>6.5</td>
<td>Future Research</td>
<td>85</td>
</tr>
<tr>
<td>7.</td>
<td>Acknowledgement</td>
<td>88</td>
</tr>
<tr>
<td>8.</td>
<td>Appendix</td>
<td>89</td>
</tr>
<tr>
<td>8.1</td>
<td>Appendix 8.1 Derivation of the Glitch Window Width from Experimental Data</td>
<td>90</td>
</tr>
<tr>
<td>9.</td>
<td>Bibliography</td>
<td>91</td>
</tr>
<tr>
<td>10.</td>
<td>Vita</td>
<td>94</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>No.</th>
<th>Table Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Parameters for Some Common Bistable Devices</td>
<td>14</td>
</tr>
<tr>
<td>4.1</td>
<td>Values of $\frac{x_0}{\tau}$ for various $P_{FF/G}$ and $N$</td>
<td>55</td>
</tr>
<tr>
<td>5.1</td>
<td>Expressions for $C_{SYS}$</td>
<td>70</td>
</tr>
<tr>
<td>No.</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----</td>
<td>----------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>1.1</td>
<td>Synchronization of Service Request</td>
<td>3</td>
</tr>
<tr>
<td>2.1</td>
<td>A Metastable Indicating Circuit</td>
<td>15</td>
</tr>
<tr>
<td>2.2</td>
<td>Operation of a Syncro Clock</td>
<td>19</td>
</tr>
<tr>
<td>2.3</td>
<td>A Syncro Clocked System</td>
<td>23</td>
</tr>
<tr>
<td>3.1</td>
<td>Truncation of an Exponential Density Function</td>
<td>31</td>
</tr>
<tr>
<td>4.1</td>
<td>Plot of $Y$ versus $\frac{X}{\tau}$</td>
<td>52</td>
</tr>
<tr>
<td>4.2</td>
<td>Plot of $\frac{X}{\tau}$ versus $N$ with no delay and $\ln\left(\frac{P_{FF/G}}{P_{FS/G}}\right) = 1$</td>
<td>56</td>
</tr>
<tr>
<td>4.3</td>
<td>Plot of $\frac{X}{\tau}$ versus $\frac{P_{FS/G}}{P_{FF/G}}$ with no delay</td>
<td>57</td>
</tr>
<tr>
<td>6.1</td>
<td>Plots of $\ln P_{SYS}$ versus $t$ for three Flip-Flop Types</td>
<td>76</td>
</tr>
<tr>
<td>6.2</td>
<td>Job Execution Times for three Parallel Jobs</td>
<td>79</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

1.1 OBJECTIVE OF THE THESIS

Bistable devices when subjected to special input conditions exhibit nondeterministic resolution times. The purpose of this investigation is to study the effect of such nondeterministic resolution times on the reliability of two types of clocked systems - namely, the fixed period clocked system and a special type of variable period clocked system. The investigation also includes a performance comparison of the simplest form of the two system types.

This chapter informally introduces the fundamental problem associated with nondeterministic resolution times of flip-flops. Two synchronizing strategies are discussed with a brief introduction to the scheme involving the variable period clock. The material covered in the chapter is intended to serve as a background for later chapters and will be helpful in the understanding of the more detailed discussion of the purpose and scope of the thesis presented in Section 1.5. The later part of Section 1.5 describes the organization of the thesis.

* This work has been supported by the Division of Research Resources of the National Institutes of Health under Grant RR 00396.
1.2 COMMUNICATION BETWEEN A COMPUTER SYSTEM AND ITS ENVIRONMENT

A computer system is a collection of components working together to provide a specified service (1,2)*. The user of the service is the environment which is the 'world' outside the system. Communication between the system (the server) and its environment (the user) is essential for the successful utilization of the service and as a result there will be events that occur in the environment that will directly influence the behavior of the system.

The temporal structure of a computer system is discrete and is defined by its clock (or set of clocks). In order that an external event can influence the system's behavior, its occurrence must be known to the system. This is done by transforming the occurrence of the external event into an equivalent internal event the occurrence of which is aligned with a clock event. Such a transformation is termed synchronization. It is this internal event that is seen by the system and its occurrence indicates that the corresponding external event has occurred.

Consider the case of a service request sent to the system by asserting one of its input lines. The occurrence of this event (the assertion of the input line) leads to the holding of a condition (the line is asserted). This condition is represented physically by the input line being held 'high' (logical one). Figure 1.1 shows a common way of synchronizing the external event. With the Request line asserted, the arrival of a clock pulse will cause a change of state in the flip-flop leading to the holding of the condition that the request is received (i.e. the Request Received line is held high). The system sees the assertion of the Request

* The numbers in parentheses in the text indicate references in the Bibliography.
Figure 1.1: Synchronization of service request
Received line but not that of the Request line for only the former is available to its components. This usually leads to the processing of service request by the system. If the Request line is not asserted when the clock pulse arrives (i.e. no occurrence of the external event), there will be no change of state in the flip-flop causing the Request Received line to remain unasserted. No service request is processed in this case.

1.3 ANOMALOUS SYSTEM BEHAVIOR RESULTING FROM SYNCHRONIZER FAILURE

Ideally, a system should be able to decide if an external event occurs or not, after every clock pulse. For in either case, a consistent system behavior results. It is found that this is not always true. There are times when the system cannot make a proper decision even though the external event has occurred i.e. there is a failure in the synchronization of the external event. The cause of this failure in practical synchronizer circuits, called synchronizers, similar to the one shown in Figure 1.1 is discussed below.

Most synchronizers use bistable devices like flip-flops to indicate the occurrence of an external event. The arrival of the clock pulse tests for the condition: "the external event has occurred". If this is true, the flip-flop is made to change state. However, if the external event occurs too close in time to the clock pulse, there may not be enough time for the condition to be sensed and an unambiguous state change initiated. This can lead to anomalous behavior of the synchronizer (3,4,5).

It is found that bistable devices have a region of indecisiveness (6,7) or a metastable state between their two stable states. If a state change is initiated by a marginal input to such a device, it can be driven into the metastable state and once in it, the device can stay in that state for an
arbitrarily long time. Such a phenomenon is termed a synchronizer failure, or more informally, a glitch phenomenon (8). The output of the device when it is in the metastable state has an undefined or non-standard value which individual hardware components may interpret as either a '1' or a '0'. Hence part of the system may see a deasserted output while another part may see an asserted output. In the case of the service request, that part of the system that sees it being asserted may initiate actions that will process the request while the other part that sees its deassertion will proceed as if the request was not there. This results in conflicting actions being taken leading to the system’s behavior deviating from specifications. This constitutes a system failure (1).

1.4 THE SYNCRO CLOCK - A METHOD TO AVOID ANOMALOUS SYSTEM BEHAVIOR

A number of techniques have been proposed to avoid anomalous system behavior. One very common technique is to delay the reading of the bistable’s output for a fixed time interval, hoping that the device will leave the metastable state by that time. It turns out that this technique does not work all the time as there can be no guarantee that the device would have left the metastable state after any fixed time interval. This is because given a finite time interval, there is a nonzero probability that a bistable device will take longer than that interval to leave the metastable state (3-8). Hence when the clock pulse that senses the bistable’s output occurs, the output may still be undefined.

A more effective technique is to delay the sensing of the bistable’s output for as long as necessary, as discussed independently in (9) and in
unpublished internal memoranda* of the Computer Science Department, Washington University, St. Louis, as well as in (10). The clock used is a pausable delay based clock which for brevity is termed a syncro clock. Unlike traditional clocks, the time between consecutive clock pulses of a syncro clock can be varied. The time is made to depend on the state of the bistable. Using circuits capable of detecting when the bistable (or synchronizer) is in the metastable state, the next clock pulse is delayed until the bistable (or synchronizer) is out of the metastable state. Thus the definitive clock pulse that senses the output will only be generated after the bistable (or synchronizer) output has a logically defined value.

1.5 SCOPE AND ORGANIZATION OF THE THESIS

A serious implication in the use of syncro clocks in computer system is that the separation between consecutive clock pulses is now a random variable. Hence there is a nonzero probability that a job may be delayed simply because the computer system has to spend sometime waiting for its set of synchronizers to get out of their metastable states. Given that a system's performance is specified in terms of an upper bound on the job's execution time, there is a nonzero probability of this being exceeded resulting in a system failure. The basic question then is whether and under what circumstances this probability is lower than that of the occurrence of a system crash due to anomalous system behavior when fixed period clocks are used. It is the purpose of this thesis to explore this

* In Working Notes 2, 3, 6, and 7 by J. R. Cox, Jr. and M. J. Stucki, Computer Science Department, Washington University, St. Louis.
question and to understand the tradeoffs involved. The problem of characterizing clocked system performance is also dealt with. The intent here is to develop a concise and useful expression for system failure probability that is sufficiently general. The results of the investigation are presented in this thesis which is organized as follows.

Chapter 2 deals with the modelling of clocked systems. Part of the material in that chapter is basically a review of the results of other workers. The model for clocked bistable devices used and the operation of the syncro clock discussed, are derived from these results. Using these results, the models for flip-flops, system clocks and clocked systems are defined. The material from Chapter 3 onwards is a report of the research undertaken for this thesis. Results of the thesis are developed in those chapters which are briefly introduced here. Chapter 3 presents the properties of an interesting probability density function used to describe the probabilistic behavior of the time between consecutive clock events of a syncro clock. Using a theorem proved in Chapter 3, the performance of systems, each with a single clock (fixed period or syncro clock), are compared in Chapter 4. This is extended in Chapter 5 to the case where the systems have more than one clock. A number of parameters for characterizing clocked system performance are obtained. It is found that the comparison of performance of syncro clocked systems is greatly facilitated by the use of a parameter derived in Chapter 5. Chapter 6 presents two examples illustrating the use of the results of Chapter 5 and summarizes the results of the research.

There are two main results of the thesis. The first is that for zero path delay in the feedback paths from the alignment rank to the input of the syncro clock, a syncro clocked module tend to give a lower probability
of failure than a fixed period clocked module of similar structure. However, the performance of the syncro clocked module degrades when the path delays increase causing the syncro clocked module failure probability to increase and for sufficiently large delays, the failure probability can be much higher than that of the fixed period clocked module. The second result of the thesis is that clocked system failure probability can be conveniently expressed as a function of three factors representing, the effect of the input arrival process, the system structure, and the nondeterministic resolution times of the flip-flops used. Limitations of the results are also discussed in Chapter 6 together with some suggestions for future research.
2. A MODEL FOR CLOCKED SYSTEMS

2.1 INTRODUCTION

This chapter defines the structure of a model for clocked systems operating in an environment where synchronization of external events is essential. Since the interface between a system and its environment is the set of synchronizers, the core of each of which is a clocked bistable device (or a clocked bistable in short), the clocked bistable device is a key part of the model. The anomalous behavior of the bistable has to be adequately represented in the model. Moreover as the synchronizers are triggered by clock pulses from the system clock, a model for the clock has to be defined.

2.2 MODELLING THE ANOMALOUS BEHAVIOR OF THE BISTABLE DEVICE

The model for the clocked bistable to be used is based principally on the work of Hurtado (7). Before going further, some assumptions and definitions need to be stated. Firstly, it is assumed that the input and output signals of the device are typical binary signals bounded by an upper bound, $V_{ub}$, and a lower bound, $V_{lb}$. The switching times are taken to be consistent with the typical rise and fall times of the device and its logic family. Moreover there are two thresholds, $V_h$ and $V_l$, which define the two logical sets $H = \{ v \mid V_h \leq v \leq V_{ub} \}$ and $L = \{ v \mid V_{lb} \leq v \leq V_l \}$, the logical high and low sets, respectively. A signal with a magnitude $v$ at the time it is observed is said to be logically defined if $v \in L \cup H$, otherwise it is logically undefined. If $v \in H$ (or $L$) then the signal is said to be high (or low). The term signal "edge" is used to refer to that part of the signal which changes from a steady value in one of the logical sets to a
steady value in the other during a transition from low to high or vice versa. For convenience, the delay or temporal separation between signal edges is measured with respect to the mid-point of the edges i.e. when \( v = \frac{1}{2} (V_h + V_l) \). Only edge triggered bistable devices are considered and the triggering clock edge is idealized to be an event termed a \textit{clock event} occurring at the point in time, \( t_c \), when the mid-point of the edge occurs. Similarly, a data edge is idealized as an \textit{input (or data input) event} occurring at the time the mid-point of the data edge occurs. The time it takes the device to produce a logically defined output after being triggered by a clock event is termed its \textit{resolution time}.

As discussed in Chapter 1, bistable devices have a metastable state and when it is left in such a state, its resolution time is nondeterministic. When the device is operating in the neighborhood of such a state, it can be represented as a finite-dimensional linear dynamic system:

\[
\dot{x}(t) = Ax(t) + n(t), \quad t > t_1, \\
x_1 = x(t_1),
\]

(2.1)

where \( x(t) \) is the state vector, \( A \) is a constant matrix, \( n(t) \) is a vector of Gaussian noise sources, \( t_1 \) is the time at which the data input is disabled due to the deassertion of the clock input and \( x_1 \) is the state of the system at \( t_1 \). If \( x_1 \) is in a small set, \( S_M \), which leads to nondeterministic resolution times, the device is said to be set in the metastable state and a \textit{glitch} is said to have occurred. The probability \( P_G(t) \) of the resolution time exceeding a time \( t \) is given by:

\[
P_G(t) = \int_{t}^{\infty} \int_{x_1 \in S_M} P(S_U \mid x_1, t_1) p(x_1) \, dx_1 \, dy \quad (2.2)
\]

where \( P(S_U \mid x_1, t_1) \) is the probability that the bistable's state \( x \) at time
y is in the set of states $S_U$ where the output is logically undefined, given that its state is $x_1$ at time $t_1$ and $p(x_1)$ is the probability density function of $x_1$.

It is found (7) that the state $x_1$ is dependent on the delay, $\delta$, between the clock and data edges (or alternately between the clock event and input event). If $\delta$ is in some interval, $I$, then $x_1$ will be set in a state in $S_M$. Thus expressing $P_G(t)$ in terms of $\delta$,

$$P_G(t) = \int_{\delta \in I} P(t|\delta) \ p(\delta) \ d\delta,$$

where,

$$P(t|\delta) = \int_{t}^{\infty} P(S_U, y|\delta) \ dy = \int_{t}^{\infty} P(S_U, y|x_1, t_1) \ dy,$$

with $p(\delta)$ is the density function of $\delta$ and $P(t|\delta)$ is the conditional probability of the resolution time exceeding $t$ for a given $\delta$.

Hurtado has shown that for the MC1016 the conditional probability function $P(t|\delta)$ is a continuous function of $\delta$ for $\delta$ in $I$. Assuming this is true for $\delta \in I = [\delta_1, \delta_2]$ and that $p(\delta)$ is also continuous in that interval, then by the Mean Value Theorem for Integrals, there exists a $\delta_0 \in [\delta_1, \delta_2]$ such that,

$$P_G(t) = P(t|\delta_0) \ast (\delta_2 - \delta_1) p(\delta_0).$$

For $I$ sufficiently small, $(\delta_2 - \delta_1) p(\delta_0)$ is approximately the probability of $\delta$ occurring in the interval $I$. Let this probability be $P_M(I)$ a function of the interval $I$, then $P_G(t)$ can be written as:

$$P_G(t) = P(t|\delta_0) \ P_M(I).$$
For a given device, the interval is fixed and hence \( \delta_0 \) is a fixed constant. Thus \( P(t|\delta_0) \) is a function of \( t \) and can be interpreted as the probability of no resolution given that \( \delta \) is in \( I \). Letting this conditional probability be \( P_{G/M}(t) \), then \( P_G(t) \) becomes,

\[
P_G(t) = P_{G/M}(t) \cdot P_M(I).
\]

(2.4)

It is found experimentally (7) that \( P_G(t) \) can be approximated by an exponential density function for \( t \) larger than some lower bound \( k \). Since \( P_M(I) \) is a function of the density function of \( \delta \) and is independent of the resolution time \( t \), then the conditional probability \( P_{G/M}(t) \) is an exponential function of \( t \).

It can be seen from the above that \( I \) is crucial in defining the bistable's anomalous behavior. Furthermore, since \( P_G(t) \) and \( P_{G/M}(t) \) are exponential functions, the time constants of the functions are important. Note that in the experimental measurement of \( P_G(t) \), \( \delta \) has a uniform distribution. This means that \( P_M(I) \) is a constant. Hence \( P_{G/M}(t) \) has to be of the same form as \( P_G(t) \) i.e. they have the same time constant. It is interesting to note that the exponential form of \( P_G(t) \) is not only limited to clocked bistable devices like the MC1016. Such an exponential probability function has also been observed in tunnel diode bistables (11).

A model for the bistable device has to include the interval or window, \( I \), the time constant for \( P_{G/M}(t) \) and the lower bound, \( k \), of the resolution time for which the exponential approximation to the conditional probability function \( P_{G/M}(t) \) holds. The concept of the window, \( I \), has been used in (12) to describe the performance of synchronizers and arbiters. The model for the clocked bistable device to be used in this thesis is defined below.
DEFINITION 2.1: The glitch window of a clocked bistable device is the interval of width \( w \), centered at distance \( d \) from a clock event such that when the delay, \( \delta \), between the data and clock edges is inside the interval, the device exhibits nondeterministic resolution time. The conditional probability density function, \( p(t) \) for the resolution time, \( t \), for \( t > k \) is:

\[
p(t) = \frac{1}{\tau} \exp\left(-\frac{t-k}{\tau}\right) u(t-k)
\]

where \( k, \tau > 0 \) are fixed constants for the device. The device is said to be marginally triggered if \( \delta \) is in \( I \).

Using the above definition, \( P_{G/M}(t) \) of equation (2.4) becomes,

\[
P_{G/M}(t) = \int_{t}^{\infty} p(x) \, dx.
\]  \hspace{1cm} (2.5)

Note that the above model is valid for \( t \) greater than \( k \). For TTL and ECL devices, the value of \( k \) is about twice the maximum propagation time. It is assumed in this thesis that the device's output is only observed at a time larger than \( k \) from the occurrence of the triggering clock event.

The above model allows one to characterize a clocked bistable device by the four parameters \( d, w, k \) and \( \tau \). Table 2.1 shows some values of the parameters. They are derived from experimental data (courtesy of Tom J. Chaney, Computer Systems Laboratory, Washington University, St. Louis) as well as from (7) and (13) using the method given in Appendix 8.1. The analytical values of (7) (see equation 5.35, Chapter 5 of (7)) are used to determine \( w \). It is found that \( w \) for the MC1016 is 63.5 picoseconds while the experimentally determined value from (7) is 70 picoseconds.
TABLE 2.1: Parameters for some common bistable devices

<table>
<thead>
<tr>
<th>Device</th>
<th>Density Cutoff Time k (nsec.)</th>
<th>Glitch Window Offset d (nsec.)</th>
<th>Glitch Window Width w (psec.)</th>
<th>Resolution Time Constant τ (nsec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SN740874*</td>
<td>15</td>
<td>-</td>
<td>31.8</td>
<td>1.1</td>
</tr>
<tr>
<td>MC10131</td>
<td>8</td>
<td>-</td>
<td>18</td>
<td>1.7</td>
</tr>
<tr>
<td>MC1016**</td>
<td>12.5</td>
<td>1.13</td>
<td>70</td>
<td>1.8</td>
</tr>
</tbody>
</table>

* From Wann et al (13)
** From Hurtado (7)

The parameters $k$, $w$ and $\tau$ can be used to measure the performance of the device. The smaller the value of $k$, the better is the exponential approximation to $P_C(t)$. A smaller $w$ will result in the device being less likely to be marginally triggered for data events which are Poisson distributed, provided that the mean interarrival time (i.e. the mean time between data events) is large compared to the glitch window width. Since $\tau$ is the time constant of the exponential density function, $p(t)$, a smaller $\tau$ leads to a steeper exponential function which approaches zero more rapidly as $t$ increases. Moreover the mean resolution time is less for a smaller $\tau$.

2.3 THE M FLIP-FLOP - A GENERALIZED MODEL FOR THE CLOCKED FLIP-FLOP

The clocked bistable device or flip-flop, by itself, cannot be used to control the clock period of the syncro clock (9). An additional output has to be provided so that when a glitch occurs, that output will be asserted and hence can be used as a feedback to the syncro clock to inhibit the generation of clock events. There are numerous practical circuits for
M = \begin{cases} 
H, & \text{when } |V_Q - V_{\overline{Q}}| < V_T, \\
L, & \text{otherwise.}
\end{cases}

Figure 2.1: A metastable indicating circuit
glitch detection (3,5,8,9,11) and terms like trinary flip-flop (8), metastable detecting circuit (3) and indicating or 3-output flip-flop (5) have been used to describe them. A typical circuit is shown in Figure 2.1. In this thesis, such circuits are modelled by a more general version of the clocked bistable, termed an M flip-flop defined below. Except for the additional output, an M flip-flop behaves just like a clocked bistable and can be characterized by the four parameters $k$, $d$, $w$ and $\tau$.

**DEFINITION 2.2:** An M indicating flip-flop (or M flip-flop) is a clocked bistable device with parameters $k$, $d$, $w$ and $\tau$, having an additional output, $M$, such that $M$ is asserted whenever the device is in the metastable state.

### 2.4 SYSTEMS CLOCKS - FIXED PERIOD AND SYNCRO CLOCKS

M flip-flops when used as synchronizing elements are triggered by clock events generated inside the system. A clock is viewed as a generator of such events. The temporal separation between consecutive clock events need not necessarily be fixed. However if it is fixed, then the clock is termed a fixed period clock (FPC) and the fixed separation is the clock period, $T_p$. This type of clock is commonly used in clocked systems.

With the introduction of the syncro clock discussed in Section 1.4, the separation between consecutive clock events need no longer be fixed. Furthermore the separation can be varied by appropriately asserting and deasserting a special input, termed the WALT input, to the clock. When modelling such a clock it must be realized that it is essentially a two state device. When it is generating clock events without inhibition, it
is in one state, and when its WAIT input is asserted, it is in another state where the generation of clock events is inhibited. As soon as the WAIT input is deasserted, it returns to the "free-running" state, generating clock events. For normal operation of the syncro clock, the WAIT input has to be either asserted or deasserted at any given time. With this mode of operation, the syncro clock behaves very much like an unclocked flip-flop with a single input used for setting and resetting its state. Realizing this and the fact that a bistable device has a metastable state, there is a constraint on when the WAIT input can be asserted. The exact details of such a constraint is unimportant as long as one realizes that the assertion of the WAIT input tends to inhibit the generation of clock events and if the clock is in the midst of generating a clock event, one cannot predict what would happen if its WAIT input is asserted at that time. Using an approach similar to that of the glitch window, the constraint can be represented as an interval defined with respect to a clock event. If the transition from deassertion to assertion of WAIT occurs within the interval, then the generation of that clock event is unpredictable due to the fact that the generation of the clock event is already in progress when the transition occurs. For brevity, the occurrence of such a transition of the WAIT input is termed a WAIT event.

Let $t_1$ be the time of occurrence of the $i$-th clock event and $I_i$ be the critical interval associated with it. If the sequence $\{t_i\}$, for $t_i$ real and $i \in N$, is the sequence of times of occurrence of clock events generated by the clock, then the set, $S = \bigcup_{i \in N} I_i$, termed the constraint set, is the set of intervals where a WAIT event cannot occur. When the
clock is free-running, the clock events generated are at a fixed distance, $T_S$, apart. The assertion of the WAIT input merely lengthens the distance between the last generated clock event and the next clock event. If the WAIT input is never asserted, the syncro clock becomes a fixed period clock with a clock period of $T_S$.

The principle of operation of a syncro clock is illustrated in Figure 2.3. Shown in Figure 2.3a is the symbol used to represent a syncro clock with $C_{OUT}$ being its output. Without loss of generality, the clock is assumed to start generating clock events at time $t_0$, $t_1$, $t_2$, $\ldots$, etc. Let $t_{\text{WAIT}}$ be the time of occurrence of the WAIT event and $t_{\text{WAIT}}'$ be the time a WAIT transition from assertion to deassertion occurs. To inhibit the $i+1$-st clock event, $t_{\text{WAIT}}$ must be greater than $t_i$ and less than $t_i+T_S$. If $t_{\text{WAIT}}$ is larger than $t_i+T_S$ then the $i+1$-st clock event would not be inhibited and it would occur at the time $t_i+T_S$. Let $I_i$ and $I_i'$ ($\epsilon S_c$) be the critical intervals as shown. Since the $i$-th clock event was successfully generated, $t_{\text{WAIT}}$ cannot be inside $I_i$ and as the $i+1$-st clock event was successfully inhibited, $t_{\text{WAIT}}'$ has to be outside $I_{i+1}'$ - that is $t_{\text{WAIT}}'$ is not in $S_c$. Hence $t_{\text{WAIT}}'$ has to be in the shaded part of the time axis. As soon as WAIT becomes deasserted, the clock event appears and its time of occurrence, $t_i'+T_S$, is $t_{\text{WAIT}}'$. The extension, $e_{i+1}'$, in the temporal distance between the $i$-th and $i+1$-st clock events is $t_i'-(t_i+T_S) = t_{i+1}'-(t_i+T_S)$. When the WAIT input is not asserted between clock events or $t_{\text{WAIT}}' \leq t_i+T_S$, the extension is zero. The time $t_i+T_S$ is the time the $i+1$-st clock event is scheduled to be generated and a nonzero $e_{i+1}'$ means that it is delayed by that amount from its scheduled time of generation. A definition of the syncro clock is given
(a) Symbol for a synco clock

(b) Timing diagrams for a synco clock

Figure 2.2: Operation of a synco clock
below. The case where the WAIT input is not asserted in the time interval between \( t_i \) and \( t_{i+1} \) (\( = t_i + T_S \) for this case) is handled by allowing \( t'_{\text{WAIT}} \) to be equal to \( t_{\text{WAIT}} \). Hence when \( t'_{\text{WAIT}} = t_{\text{WAIT}} \), it is taken to mean that there is no assertion of the WAIT input.

**DEFINITION 2.3:** A *syncro clock* is a source of clock events, having a WAIT input and a constraint set, \( S_c \), such that the separation between consecutive clock events is variable with a minimum value of \( T_S \). Let \( t_{\text{WAIT}} \) be the time of occurrence of a deassertion-to-assertion transition of the WAIT input just after a clock event at time \( t_i \) and \( t'_{\text{WAIT}} \) \( (\geq t_{\text{WAIT}}) \) be the time the input's subsequent assertion-to-deassertion transition occurs. If \( t_{\text{WAIT}} \notin S_c \), then, the next clock event will be generated at the time \( t_{i+1} \) given by,

\[
\begin{align*}
t_{i+1} = \begin{cases} 
  t_i + T_S, & \text{if } t'_{\text{WAIT}} \leq t_i + T_S, \\
  t'_{\text{WAIT}}, & \text{otherwise.}
\end{cases}
\end{align*}
\]

When \( t_{\text{WAIT}} \notin S_c \), the behavior of the clock is undefined. \( T_S \) is termed the nominal clock period.

Throughout this thesis, \( t_{\text{WAIT}} \) is assumed to be outside the constraint set, \( S_c \). This is usually true in practice. It can be seen that the syncro clock is more general than the fixed period clock. With WAIT always deasserted, the nominal clock period of the former becomes the clock period of the latter.

### 2.5 MODULES AND CLOCKED SYSTEMS

Having defined the clock and the \( M \) flip-flop, the model for clocked systems can now be presented. The clocks defined above are used as system clocks while \( M \) flip-flops are used as synchronizing elements.
A system communicates with its environment via a set of logical paths and the subset of such paths carrying incoming messages is termed the input set, I, and that for outgoing messages is termed the output set, O. It is the incoming messages via I that requires synchronization. Thus for each (independent, unconstrained) path in I, there has to be at least one M flip-flop to synchronize the external events*, namely, the arrival of the message at the system boundary. The collection of such M flip-flops in the system is termed an alignment rank as they essentially align the occurrences of the external events with clock events generated by the system's clock or set of clocks.

For each clock within a system, there is a subsystem associated with it. All components within the subsystem depend only on that clock and the transfer of messages between such subsystems involve synchronization of the messages with the receiving subsystem's clock. Hence, in general, a system is made up of many such clocked subsystems which are termed modules.

**DEFINITION 2.4:** A module is a system having its own clock and interacting with its environment via an input, I, and an output, O. The translation of the input I into its own time reference is performed by a set of M flip-flops operating in parallel with I applied at the D-inputs and its own clock events applied at the C-inputs. The set of translation devices is termed an alignment rank.

*Unless otherwise stated, the term "external event" is used throughout this thesis to mean an externally generated event that requires synchronization.
Since a module has a single clock, synchronization of external events conveyed via I is only with respect to that clock. If a module has a fixed period clock then the M outputs of the M flip-flops have no control over the clock period. On the other hand, if it is a syncro clocked module then the separation between consecutive clock events is controlled by the M outputs. This is done by collectively ORing the M outputs to control the WAIT input of the syncro clock. The input will be asserted as soon as one or more of the flip-flops are marginally triggered by the same clock event and it will remain asserted until all the flip-flops have resolved. The fixed period and syncro clocked modules are defined below.

**Definition 2.5:** If a module has a fixed period clock with clock period $T_p$, then it is a fixed period clocked (FPC) module.

Note that the flip-flop used in an FPC module do not need the metastable detecting capability. In this thesis, the M flip-flop is used in FPC modules merely as a representative clocked bistable device and to facilitate performance analysis of FPC and syncro clocked modules.

**Definition 2.6:** If the module has a syncro clock with nominal clock period $T_s$ and a rank of M flip-flops (with parameters $k$, $d$, $w$, and $r$) with the M outputs logically ORed into the WAIT input of the syncro clock, then it is a syncro clocked module, provided that the operation of the clock is not in conflict with the constraint imposed by the constraint set, $S_c$.

A model for a syncro clocked module is shown in Figure 2.3. Delays in the paths from the M outputs to the WAIT input are included in the figure.
Figure 2.3: A syncro clocked system
It is assumed that such delays do not cause the WAIT input to be asserted at a time in conflict with the constraint imposed by the set $S_c$. Furthermore the delay $D_w$ is restricted to be less than $T_S$.

2.6 COMMUNICATION BETWEEN A MODULE AND ITS ENVIRONMENT

The sets I and O are defined to be sets of logical paths. Each such path is a medium for transmission of a single independent entity be it a signal or a group of related signals. Thus signals on different paths are independent of each other while signals within a path are not. With each of such paths is associated an M flip-flop. Its purpose being to synchronize the arrival of the signal or group of signals. For example, a path may be the set of nine physical conductors carrying one control signal and eight data signals. The control signal is used to alert the module of the arrival of the data signals. Thus only the control signal need be synchronized and the data signal lines are only sampled after successful synchronization of the control signal. In other instances, there need not be an explicit control signal, but rather the set of physical conductors or lines in the path may carry signals which follow some encoding scheme. Through proper decoding, a signal can be derived from the arriving signals to indicate the arrival of the message. This derived signal is the one that needs to be synchronized. Thus the form of communication is assumed to be oriented to the transmission of a message entity and to the detection of the arrival of such an entity at the module. Only when such an arrival is known (i.e. successful synchronization) can reading of the message be attempted.
3. THE TRUNCATED EXPONENTIAL DENSITY FUNCTION

3.1 INTRODUCTION

The clock period of a syncro clocked module is a random variable with minimum value $T_S$. Its density function is composed of an impulse at $T_S$ and an exponential tail as derived below.

Consider the case where the arrival of external events at each input to the alignment rank is described by an integer valued random process, $[N(t), 0 \leq t]$, with stationary independent increments (14, 15) and $N(t)$ being the number of arrivals within a time interval $t$ with $N(0) = 0$.

Marginal triggering occurs when, for a given time of occurrence of a triggering clock event, $t_c$, the number of arrivals, $n(t_c)$, within the glitch window, $[t_c + d - \frac{W}{2}, t_c + d + \frac{W}{2}]$ is greater than zero. That is,

$$n(t_c) = N(t_c + d + \frac{W}{2}) - N(t_c + d - \frac{W}{2}) > 0.$$

By the stationary independent increment property of $N(t)$, $n(t_c)$ has the same distribution as $N(w)$. Hence the number of arrivals within the glitch window is independent of $t_c$, provided that the glitch windows do not overlap. Assuming that the arrival distribution is a continuous function, $F_N(t)$, of time, then, the probability of marginal triggering, $P_M$, for any given $t_c$, is given by,

$$P_M = Pr[N(w) \geq 1]$$

$$= F_N(w),$$

that is, a constant for a fixed $w$. 
Let $T$ be the random variable representing the clock period, $n$ be the number of $M$ flip-flops in the rank, $t_{Rj}$ the resolution time for the $j$-th $M$ flip-flop, $D_j$ the path delay from the $M$ output of the $j$-th $M$ flip-flop to the input of the OR gate (see Figure 2.3) and $D_w$ be the path delay from the output of the OR gate to the WAIT input of the syncro clock. The OR gate is assumed to have no delay. The effect of $D_j$ and $D_w$ is to reduce the effective resolution time by that amount. Hence to effectively allow the next clock event to be generated at time $T_s$, the nominal clock period, the OR input has to be deasserted at a time earlier than

$$T_s - D_w \text{ or } t_{Rj} \leq T_s - D_w \text{ for all } j = 1, 2, \ldots, n.$$  

The generation of the next clock event will be delayed when $t_{Rj} > T_s - D_w$ for any $j$. Assuming that $P_M \ll 1$, and that the $M$ flip-flops are independent of each other, then,

$$\Pr[T = T_s] = \Pr[t_{Rj} \leq T_s - D_w \text{ for all } j = 1, \ldots, n]$$

$$= \prod_{j=1}^{n} \Pr[t_{Rj} \leq T_s - D_w]$$

$$= \prod_{j=1}^{n} [1 - \Pr[t_{Rj} > T_s - D_w]].$$

The effect of the delay $D_j$ is to shift the conditional density function for $t_{Rj}$ to the right along the time axis by an amount $D_j$ so that the exponential density curve starts at $k + D_j$ instead of at $k$. In other words, the conditional density function for $t_{Rj}$ becomes $\frac{1}{\tau} \exp(-\frac{t-k-D_j}{\tau}) u(t-k-D_j)$. Thus,

$$\Pr[t_{Rj} > T_s] = P_M \exp(-\frac{T_s-k-D_j}{\tau}), \text{ for all } j = 1, \ldots, n,$$

or,

$$\Pr[t_{Rj} > T_s - D_w] = P_M \exp(-\frac{T_s - D_w - k - D_j}{\tau}), \text{ for all } j = 1, \ldots, n.$$
Hence,
\[ \Pr[T = T_S] = \prod_{j=1}^{n} \left[ 1 - P_M \exp\left( - \frac{T_{S-D_i^-}-k-D_i}{\tau} \right) \right] \]
\[ \approx 1 - \sum_{j=1}^{n} P_M \exp\left( - \frac{T_{S-D_i^-}-k-D_i}{\tau} \right) \]
\[ = 1 - n P_M \exp\left( - \frac{T_{S-k}}{\tau} \right) \exp\left( \frac{D_i}{\tau} \right) \sum_{j=1}^{n} \exp\left( \frac{D_i}{\tau} \right). \]

Similarly,
\[ \Pr[T > t > T_S] = 1 - \Pr[t_{Rj} \leq t-D_i^- \text{ for all } j=1, \ldots, n] \]
\[ = 1 - \prod_{j=1}^{n} \left[ 1 - P_M \exp\left( - \frac{t-D_i^- - k-D_i}{\tau} \right) \right] \]
\[ \approx P_M \exp\left( - \frac{t-k}{\tau} \right) \exp\left( \frac{D_i}{\tau} \right) \sum_{j=1}^{n} \exp\left( \frac{D_i}{\tau} \right). \]

Thus the probability density function, \( f_T(t) \), for the clock period is given by,
\[ f_T(t) = \Pr[T = T_S] \delta(t-T_S) + \left\{ - \frac{d}{dt} (\Pr[T > t]) \right\} u(t-T_S), \]
or,
\[ f_T(t) = \left[ 1 - P_M \exp\left( - \frac{T_{S-k}}{\tau} \right) \exp\left( \frac{D_i}{\tau} \right) \sum_{j=1}^{n} \exp\left( \frac{D_i}{\tau} \right) \right] \delta(t-T_S) \]
\[ + \left\{ P_M \frac{1}{\tau} \exp\left( - \frac{T_{S-k}}{\tau} \right) \exp\left( \frac{D_i}{\tau} \right) \left[ \sum_{j=1}^{n} \exp\left( \frac{D_i}{\tau} \right) \right] \right. \]
\[ \left. \exp\left( - \frac{t-T_S}{\tau} \right) \right) u(t-T_S). \]  \tag{3.1} \]

where \( \delta(t) \) is the unit strength impulse function.
Letting,

\[ a = p_m \exp\left(-\frac{T_s-k}{\tau}\right) \exp\left(\frac{D_w}{\tau}\right) \sum_{j=1}^{n} \exp\left(\frac{D_i}{\tau}\right) \]

\[ = p_m \exp\left(-\frac{T_s-k}{\tau}\right) d_n, \quad (3.2) \]

where, \( d_n = \exp\left(-\frac{D_w}{\tau}\right) \sum_{j=1}^{n} \exp\left(\frac{D_i}{\tau}\right) \),

then \( f_T(t) \) becomes,

\[ f_T(t) = (1-a) \delta(t-T_s) + \frac{a}{\tau} \exp\left(-\frac{t-T_s}{\tau}\right) u(t-T_s). \quad (3.3) \]

The factor \( d_n \) used in (3.2) is essentially a delay factor representing the effect of delays in the feedback paths. A function of this form is defined below. In the general form of equation (3.3) \( a \) can be any real number but since in this thesis the function is used as a density function, \( a \) is restricted to be non-negative and no larger than 1.

It is interesting to note that when \( D_j = D_w \) for all \( j \) and for some \( D_M \), then,

\[ d_n = \exp\left(-\frac{D_w}{\tau}\right) n \exp\left(-\frac{D_M}{\tau}\right) \]

\[ = n \exp\left(-\frac{D_w+D_M}{\tau}\right). \]

For the special case when \( D_j = 0 \) for all \( j \),

\[ d_n = n \exp\left(-\frac{D_w}{\tau}\right). \]

Comparing the above two relations for \( d_n \), the effect of having the same delay for all feedback paths is the same as having no delays at all from the \( M \) outputs to the OR inputs, and the delay from the OR output to the \( \text{WAIT} \) input increased by an amount \( D_w \) from \( D_w \) to \( D_w+D_M \).
If the delays, $D_j$ for $j=1, \ldots, n$, are such that there is a maximum delay $D_i$ such that $\sum_{j \neq i} \exp(\frac{D_j}{\tau}) \ll \exp(\frac{D_i}{\tau})$, then,

$$d_n \approx \exp(-\frac{D_n+D_i}{\tau}).$$

This means that a sufficiently large delay in one of the paths from the $M$ outputs to the OR inputs tends to dominate over all the smaller delays in affecting the behavior of the system. In all the above cases, the effect of delays in the feedback paths is to degrade the performance of the syncro clock by making it more likely for clock events to be delayed. A larger $a$ means a higher probability of delaying the generation of the next clock event. This probability increases exponentially with $D_j$ and $D_n$.

For convenience, the function of the same form as $f_T(t)$ in equation (3.3) is named a truncated exponential function (see definition below). This is a more general form of equation (3.3).

**DEFINITION 3.1**: The function,

$$g(t) = (1-a)\delta(t-T) + \frac{a}{\tau} \exp(-\frac{t-T}{\tau})u(t-T),$$

with $0 \leq a \leq 1$ and $T, \tau > 0$, is termed a truncated exponential density (TED) function.

This chapter is devoted to the discussion of the general properties of the TED function. An important theorem related to the convolutions of such functions is given and proved. The theorem establishes a bound on the error involved when a TED function is used to approximate the convolution function. This approximation is used in later chapters.
(Chapters 4 and 5) to describe the probabilistic behavior of the sum of syncro clock periods allowing one to deal with sequences of clock events.

3.2 SOME ELEMENTARY PROPERTIES OF THE TED FUNCTION

A TED function has the following fundamental properties.

Property 1: For any real \( a \),
\[
\int_{-\infty}^{\infty} g(t) \, dt = \lim_{b \to T} \int_{b}^{\infty} g(t) \, dt = 1.
\]

In general, for any real \( b \),
\[
\int_{b}^{\infty} g(t) \, dt = \begin{cases} 
1, & \text{for } b < T, \\
\exp(-\frac{t-T}{\tau}), & \text{otherwise}.
\end{cases}
\]

Property 2: When \( a = 1 \),
\[ g(t) = \frac{1}{\tau} \exp(-\frac{t-T}{\tau}) u(t-T). \]
Thus \( g(t) \) becomes a negative exponential function of \( t-T \).

Property 3: When \( a = 0 \),
\[ g(t) = \delta(t-T), \]
that is, the TED function degenerates to the unit strength impulse function located at \( t = T \).

If \( a \) is restricted to be non-negative and less than 1, then \( g(t) \) is the density function of the random variable \( Y(t), t \geq T \), derived from the random variable \( X(t) \), such that,
\[
Y(t) = \begin{cases} 
X(t), & \text{for } t \geq T, \\
0, & \text{otherwise},
\end{cases}
\]
and \( X(t) \) has the exponential density function,
\[ f_X(t) = \frac{1}{\tau} \exp(-\frac{t-c}{\tau}) u(t-c) \quad \text{for some constant } c < T. \]
Figure 3.1: Truncation of an exponential density function
Thus \( g(t) \) represents the density function of a random variable which is
the truncated form of the density function of an exponentially distrib-
uted random variable with \( 1-a \) being the probability of the latter
taking on a value for all \( t \) less than \( T \), below which truncation occurs,
as illustrated in Figure 3.1.

3.3 THE N-FOLD CONVOLUTION OF THE TED FUNCTION

The \( N \)-fold convolution of a TED function can be approximated by
another TED function. The following theorem gives an upper bound on the
error involved.

**Theorem 3.1** The \( N \)-fold convolution of the TED function,

\[
\begin{align*}
g(t) &= (1-a)\delta(t-T) + \frac{a}{\tau} \exp\left(-\frac{t-T}{\tau}\right)u(t-T) \\
\end{align*}
\]

with \( 0 \leq a \leq 1 \), \( T, \tau > 0 \) is the function,

\[
\begin{align*}
c_N(t) &= (1-Na)\delta(t-NT) + \frac{Na}{\tau} \exp\left(-\frac{t-NT}{\tau}\right)u(t-NT) + h(t)
\end{align*}
\]

where,

\[
0 < \left| \int_{-\infty}^{x+NT} h(t) \, dt \right| < u(x) \frac{e^{-x/\tau}}{1 + x/\tau} \left[ (1+|a|(1 + x/\tau))^N - (1 + N|a|(1 + x/\tau)) \right].
\]

**Proof:**

Let \( G(s) \) and \( G_N(s) \) be the Laplace transforms of the TED function \( g(t) \)
and its \( N \)-fold convolution function, \( c_N(t) \), respectively.

Then,

\[
G(s) = \exp(-sT) \left[ (1-a) + \frac{a}{1+sT} \right],
\]

(3.4)
and, \[ C_N(s) = [G(s)]^N \]
\[ = \exp(-sNT) \left[(1-a) + \frac{a}{1+st}\right]^N. \quad (3.5) \]

Let
\[ c_N^i(t) = (1-Na)6(t-NT) + \frac{Na}{\tau} \exp\left(-\frac{t-NT}{\tau}\right) u(t-NT), \quad (3.6) \]
then the Laplace transform of \( c_N^i(t) \) is,
\[ c_N^i(s) = \exp(-sNT) \left[(1-Na) + \frac{Na}{1+st}\right]. \quad (3.7) \]

Let \( h(t) = c_N(t) - c_N^i(t), \) then,
\[ H(s) = \mathcal{L}[h(t)] \]
\[ = C_N(s) - c_N^i(s) \]
\[ = \exp(-sNT) \left[\left[(1-a) + \frac{a}{1+st}\right]^N - \left[(1-Na) + \frac{Na}{1+st}\right]\right] \]
\[ = \exp(-sNT) \left[\left[1 - \frac{ast}{1+st}\right]^N - \left[1 - \frac{Nast}{1+st}\right]\right] \]
\[ = \exp(-sNT) \sum_{i=2}^{N} \binom{N}{i} (-a)^i \frac{s^i}{(s + \frac{1}{\tau})^i}. \]

As \( \mathcal{L} \left[ \int_0^y h(t) \, dt \right] = \frac{1}{s} H(s), \) then,
\[ \frac{1}{s} H(s) = \sum_{i=2}^{N} \binom{N}{i} (-a)^i \exp(-sNT) \frac{s^i}{(s + \frac{1}{\tau})^i}. \]

By the Heaviside Expansion Theorem (pp. 1021 equation 29.2.21 of (16)),
\[ \mathcal{L}^{-1} \left[ \frac{s^i-1}{(s + \frac{1}{\tau})^i} \right] = \exp\left(-\frac{y}{\tau}\right) \sum_{n=1}^{i} \frac{p^{i-n} \left(-\frac{1}{\tau}\right)^n}{(i-n)! (n-1)!} \frac{y^{n-1}}{(n-1)!}, \]
where, \( p(s) = s^{i-1}. \)
and \[ p^{(i-n)}(\frac{-1}{\tau}) = \frac{d^{i-n}}{ds^{i-n}} (s^{i-1}) \bigg|_{s = \frac{-1}{\tau}} = \frac{(i-1)!}{(n-1)!} (\frac{-1}{\tau})^{n-1}. \]

Hence,
\[ \mathcal{L}^{-1} \left[ \frac{s^{i-1}}{(s + \frac{1}{\tau})^i} \right] = \exp\left( -\frac{Y}{\tau} \right) \sum_{n=1}^{i} \frac{(i-1)!}{(n-1)! (i-n)!} (\frac{-1}{\tau})^{n-1} \frac{Y^{n-1}}{(n-1)!} \]
\[ = \exp\left( -\frac{Y}{\tau} \right) \sum_{n=1}^{i} \frac{(i-1)!}{(n-1)!} \left( \frac{-Y}{\tau} \right)^{n-1} \frac{Y^{n-1}}{(n-1)!} \]
\[ = \exp\left( -\frac{Y}{\tau} \right) \sum_{k=0}^{i} \frac{(i-1)!}{k!} \left( \frac{-Y}{\tau} \right)^{k} , \text{ with } k = n-1. \]

Using the translation property (pp. 1021 equation 29.2.15 of (16)) of the Laplace transforms,
\[ \mathcal{L}^{-1} \left[ \exp(-sNT) \frac{s^{i-1}}{(s + \frac{1}{\tau})^i} \right] = \exp\left( -\frac{Y-NT}{\tau} \right) \sum_{k=0}^{i-1} \frac{(i-1)!}{k!} \left( \frac{-Y-NT}{\tau} \right)^{k} u(y-NT). \]

Let \( x = y - NT \) and,
\[ E(x) = \int_{0}^{x+NT} h(t) \, dt. \quad (3.8) \]

Then,
\[ E(x) = \mathcal{L}^{-1} \left[ \frac{1}{S} H(s) \right] = \sum_{i=2}^{N} \binom{N}{i} (-a)^{i} \exp\left( -\frac{x}{\tau} \right) \sum_{k=0}^{i-1} \frac{(i-1)!}{k!} \left( -\frac{x}{\tau} \right)^{k} u(x). \]

\[ 0 < E(x) < u(x) \exp\left( -\frac{x}{\tau} \right) \sum_{i=2}^{N} \binom{N}{i} (-a)^{i} \sum_{k=0}^{i-1} \frac{(i-1)!}{k!} \left( \frac{-x}{\tau} \right)^{k} \]
\[ < u(x) \exp\left( -\frac{x}{\tau} \right) \sum_{i=2}^{N} \binom{N}{i} \left| a \right|^{i} \sum_{k=0}^{i-1} \frac{(i-1)!}{k!} \left( \frac{x}{\tau} \right)^{k} \]
\[ = u(x) \exp\left( -\frac{x}{\tau} \right) \sum_{i=2}^{N} \binom{N}{i} \left| a \right|^{i} (1 + \frac{x}{\tau})^{i-1} \]
\[ -35 - \]
\[ u(x) \exp\left(-\frac{x}{\tau}\right) \sum_{i=2}^{N} \binom{N}{i} \left|a\right|^i \left(1 + \frac{x}{\tau}\right)^i \]
\[ = \frac{u(x) \exp\left(-\frac{x}{\tau}\right)}{1 + \frac{x}{\tau}} \left\{ \left[1 + \left|a\right| \left(1 + \frac{x}{\tau}\right)\right]^N - \left[1 + N\left|a\right| \left(1 + \frac{x}{\tau}\right)\right]\right\}. \]

Q. E. D.

3.4 FURTHER PROPERTIES OF THE TRUNCATED EXPONENTIAL DENSITY FUNCTION

With the approximation introduced above, some further properties of the TED function can be derived.

**Property 4:** From the final value theorem of the Laplace transform (17),
\[ \int_0^\infty h(t) \, dt = E(\infty) \]
\[ = \lim_{s \to 0} \frac{1}{s} H(s) \]
\[ = 0. \]

That is,
\[ \int_0^{NT+x} h(t) \, dt = - \int_{NT+x}^{\infty} h(t) \, dt = \left| E(x) \right|. \]

**Property 5:** When \( a = 0 \),
\[ \int_0^{x+NT} h(t) \, dt = 0 \text{ and } c_N(t) = \delta(t-NT). \]

This is true since for \( a = 0 \), \( g(t) \) becomes an impulse and \( c_N(t) \) is the \( N \)-fold convolution of the impulse.

**Property 6:** For \( a = 1 \),
\[ 0 < E(x) < \frac{u(x) \exp\left(-\frac{x}{\tau}\right)}{1 + \frac{x}{\tau}} \left\{ \left(2 + \frac{x}{\tau}\right)^N - (1 + N(1 + \frac{x}{\tau})) \right\}. \]
The TED function, $g(t)$, becomes an exponential function and $C_N(s)$ becomes $\exp(-sNT) \frac{1}{(1 + sr)^N}$, or,

$$c_N(t) = \mathcal{L}^{-1}[C_N(s)] = \mathcal{L}^{-1}\left[\exp(-sNT) \frac{1}{(1 + s)^N}\right] = \frac{(t-NT)^{N-1}}{(N-1)!} \exp\left(-\frac{t-NT}{\tau}\right).$$

Hence $c_N(t)$ is identical to the density function of an Erlangian (or gamma) distribution of order $N$.

**Property 7:** For $N = 1$, $E(x) = 0$ and hence,

$$c_N'(t) = c_N(t).$$

**Property 8:** For $0 < \frac{ax}{\tau} << 1$,

$$|E(x)| \approx \frac{\exp\left(-\frac{x}{\tau}\right)}{1 + \frac{x}{\tau}} (1^N - 1) = 0.$$

The smaller the value of $ax$, the smaller is the error bound.

### 3.5 Using the Approximate Function for the N-Fold Convolution

Consider a random variable $Y$ with density function,

$$f_Y(y) = (1 - a)\delta(y - Y_0) + \frac{a}{\tau} \exp\left(-\frac{y-Y_0}{\tau}\right) u(y-Y_0) \quad (3.9)$$

with $0 < a < 1$ and $Y_0, \tau > 0$.

Let $Z$ be the sum of $N$ such variables, i.e., $Z = \sum_{i=1}^{N} Y_i$. Since $Z$ is the sum of $N$ independent and identically distributed random variables, the density function of $Z$, $f_Z(z)$, is the $N$-fold convolution function of $f_Y(y)$.

Hence by Theorem 3.1,

$$f_Z(z) = (1 - Na)\delta(y - NY_0) + \frac{Na}{\tau} \exp\left(-\frac{y-NY_0}{\tau}\right) u(y-NY_0) + h(y).$$
Let \( P_Z(z) \) and \( P'_Z(z) \) be the probability distribution of \( Z \) with,

\[
P_Z(z) = \int_0^z f_Z(q) \, dq
\]

and

\[
P'_Z(z) = \int_0^z \left[ (1-Na)\delta(q-NY_o) + \frac{Na}{\tau} \exp\left(-\frac{q-NY_o}{\tau}\right) u(q-NY_o) \right] dq.
\]

Then,

\[
P_Z(z) - P'_Z(z) = \int_0^z h(q) \, dq. \tag{3.10}
\]

By equation (3.8),

\[
P_Z(z) - P'_Z(z) = E(z). \tag{3.11}
\]

Theorem 3.1 gives the bound on \( E(z) \) and hence the bound on the error in the probability distribution function for a given \( Z \) when \( P'_Z(z) \) is used to estimate \( P_Z(z) \).

If \( Y \) is the time separation between two consecutive clock events generated by the clock of a syncro clocked module with \( Y_o = T_S \) and \( a \) is defined by (3.2), then \( Z \) represents the sum of \( N \) such time separations. Hence \( P'_Z(z) \) is an approximation of the distribution function of the time separation between the first and \( N+1 \)-st clock events in a train of \( N+1 \) consecutive clock events. Then \( 1 - P'_Z(z) \) is the probability that the module takes a time longer than \( z \) to generate \( N \) consecutive clock events.

From Property 4, the error bound on \( 1 - P'_Z(z) \) is still \( |E(z)| \). It is interesting to note that for \( z < NT_S \), \( P'_Z(z) = 0 \) since \( N+1 \) clock events cannot be generated within a time interval less than \( NT_S \). For \( z > NT_S \), the difference \( x = z - NT_S \) is the time extension from the theoretical minimum, \( NT_S \), required to generate those pulses. For \( z \) sufficiently small such that \( \frac{ax}{\tau} \ll 1 \), the error bound \( E(z) \) is approximately zero and \( P'_Z(z) \) is a good estimate of \( P_Z(z) \). For our purposes, \( \frac{ax}{\tau} \) is assumed to be much
less than 1 and the density function for $Z$ is,

$$f_Z(t) = (1-Na)\delta(t-NT) + \frac{Na}{\tau} \exp\left(-\frac{t-NT_S}{\tau}\right) u(t-NT_S). \quad (3.11)$$

This density function is used in later chapters to characterize the performance of syncro clocked modules and systems. It is important to note that the area under the exponential tail (i.e., $a$ in equation (3.9) and $Na$ in (3.11)) is the probability that there is an extension (or delay in clock event generation). The coefficient $a$ in the general TED function is a measure of how likely it is for an extension to occur - the larger $a$ is, the more likely it is for the extension to occur. The use of the coefficient $a$ is discussed further in the next chapter.
4. PERFORMANCE OF FIXED PERIOD AND SYNCRO CLOCKED MODULES

4.1 FAULT CONDITION ARISING FROM MARGINAL TRIGGERING

When an M flip-flop is marginally triggered, the flip-flop is set in the metastable state producing a logically undefined output. Sensing of the output takes place later at some finite time interval from the occurrence of the triggering clock event. As there is no guarantee that the flip-flop will resolve by then, the output value can be logically undefined when sensed, in which case a fault condition results. If the output is sensed and processed directly by more than one subsystem (viz. the fan-out is more than one), there is a potential for inconsistency in the interpretation of the output value to result. This can lead on to a system failure due to conflicting actions being taken. Thus marginally triggering the flip-flop can lead to a fault condition and the occurrence of such a fault is termed a glitch hazard. This type of fault is related more to timing than to hardware component failure. The device will perform properly as long as there is no marginal triggering, very much unlike the case of a hardware component failure where the effect is more permanent and more easily detectable. Hence glitch hazard can be classified as a fault, transient in nature, or more precisely, a fault related to synchronization.

Looking at the problem at the module level, marginally triggering its alignment rank, that is, one or more of its M flip-flops is marginally triggered, has a potential of such a fault occurring. For brevity, when one or more of the M flip-flops experiences such a fault, a glitch hazard is said to have occurred in the rank. This can lead to an
erroneous state (i) in the module. Such a state can cause a module failure. It is natural then to use the probability of occurrence of such a failure as a performance measure of a module.

With this form of performance measure, comparison of performance between modules of the same or different types (fixed period or syncro clocked) can be made. However, it is pertinent to note that such a measure is only related to the module's performance in an environment where synchronization is essential. To compute the overall module reliability figure, such a form of failure is only one aspect of the problem. The effect on module reliability of component failures, design errors, fabrication problems, software bugs and the like have to be taken into account. In this thesis, only module failure arising from glitch hazard is treated. With this in mind, and unless otherwise stated, a module failure is assumed to be caused by glitch hazards.

4.2 EFFECT OF A GLITCH HAZARD ON MODULE FAILURE

Whether glitch hazard will cause a module failure or not depends very much on the susceptibility of the module to the fault. In the case of the fixed period clock (FPC) module, the alignment rank's outputs are used after some fixed interval from the occurrence of the triggering clock event. If the sensed output of an M flip-flop in the rank is logically undefined at the end of the interval, there is a potential for module failure. In cases where the flip-flop output is used and processed directly by more than one component at the same time, there exists a possibility that the components may not see a consistent output value - some may see a logical high, others a logical low and still others neither values. Thus unpredictable module behavior can results.
To represent such a possibility of an FPC module failure, the conditional probability of module failure given that a glitch hazard has occurred is used. Such a conditional probability can also be used for syncro clocked modules. However, the effect of a glitch hazard on the performance of a syncro clocked module is not so straightforward.

In a syncro clocked module, clock events subsequent to an occurrence of marginal triggering are only generated after the alignment rank has stabilized i.e. produced stable logically defined values. Hence actions that are initiated by these clock events and those that are dependent on the outputs, are presented with logically defined outputs. There can be no inconsistency in the interpretation of the output values by the components that use them, provided of course, that the components themselves are working correctly. How then can a syncro clocked module fail? The answer to this question is clear once it is realized that the effect of a glitch hazard is to delay the generation of subsequent clock events. An excessive delay can have harmful effects on the module's behavior. As an extreme example, the cummulation of a sequence of such delays over some time can lead to the triggering of a time-out mechanism in the module's environment while engaging in an interlocked communication with the module. Thus failure of a syncro clocked module as a result of a glitch hazard is due to the resultant delay in the generation of clock events.

With the performance measure discussed above, module performance can be analysed. The underlying assumptions and notations used in the analysis are presented next followed by the analysis of the performance of FPC and syncro clocked modules. The results of the analysis are used for comparing performance of the two types of modules.
4.3 BASIC ASSUMPTIONS AND NOTATIONS

The performance analysis to be presented involve the following assumptions.

(i) The arrival of input events (or external events) into each M flip-flop of the module is an integer valued random process with stationary independent increments. This means that the probability of marginal triggering is dependent on the glitch window width, \( w \), as shown in Section 3.1. Even though in practice no more than one input event can occur at any given time instant, the analysis to follow does not require the input process to be a unit jump process i.e. a Poisson process (14).

(ii) All M flip-flops in an alignment rank are independent and identical in characteristics. Each of the flip-flops is characterized by the parameters \( d, w, k \) and \( \tau \).

(iii) The input distribution satisfying (i), of each M flip-flop in the rank is identical. This and the above assumptions lead to the probability of marginal triggering being identical for all M flip-flops and can thus be represented by a single constant, \( P_M \), for a given \( w \).

(iv) The separation between clock events is sufficiently large so that there is no overlapping of the corresponding glitch windows.

(v) The probability of marginal triggering is very much smaller than 1. This assumption holds in many practical cases, for example, those involving Poisson processes with very large interarrival times (i.e. low arrival rates) compared to the glitch window width, \( w \). It also holds for those cases where the arrival of an input event is uniformly distributed over a wide interval enclosing the window.
(vi) The outputs of an alignment rank are used no sooner than the arrival of the next clock event.

(vii) A logically undefined output in any one of the M flip-flops in the alignment rank has a potential of causing a module failure. Furthermore the conditional probability of a module failure given that at least one of the M flip-flops has not yet resolved is nonzero and is not dependent on which of the M flip-flop (or flip-flops) is still unresolved.

Used in conjunction with the above assumptions is the following list of symbols.

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Number of M flip-flops in alignment rank.</td>
</tr>
<tr>
<td>N</td>
<td>Number of clock events considered.</td>
</tr>
<tr>
<td>$P_M$</td>
<td>Probability of marginally triggering an M flip-flop in the alignment rank.</td>
</tr>
<tr>
<td>$P_G(t,j)$</td>
<td>Probability that the resolution times of a rank of j identical M flip-flops will exceed $t$, $t &gt; k$, for an FPC module.</td>
</tr>
<tr>
<td>$P_S(x,N)$</td>
<td>Probability of the delay in the generation of the $N+1$-st clock event exceeding $x$.</td>
</tr>
<tr>
<td>$P_{FF}(t)$</td>
<td>Probability of an FPC module failure due to a glitch hazard in its alignment rank lasting longer than a time $t$.</td>
</tr>
</tbody>
</table>
Conditional probability that a module will fail given that there is an occurrence of a glitch hazard in its alignment rank. The letter X should be replaced by S for a syncro clocked module and F for an FPC module.

Probability of an FPC module failure, occurring over a time interval covering N consecutive clock events, due to glitch hazards occurring in that interval.

Probability of a syncro clocked module failure due to a delay in the generation of the N-th clock event exceeding an upper bound x.

4.4 PERFORMANCE ANALYSIS OF FPC MODULES

Given that $P_M$ is the probability of marginal triggering for an M flip-flop, then, the probability of not getting a logically defined output by a time $t$, $t > k$, is

$$P_G(t,1) = P_M \int_t^\infty p(y) \, dy.$$ 

Note that in those cases where the output is sensed at every clock event, $t$ is usually less than the clock period. This is because in most cases, there are intervening logic elements inserted between the flip-flop output and the point where the output is actually used. These logic elements introduce propagation delays in the output signal path causing the actual allowable resolution time to be less than the clock period.
Using \( p(y) \) from Definition 2.1, \( P_G(t,1) \) becomes,

\[
P_G(t,1) = P_M \int_t^\infty \frac{1}{\tau} \exp(-\frac{y-k}{\tau}) \, dy
\]

\[
= P_M \exp(-\frac{t-k}{\tau}).
\]

When the number of \( M \) flip-flops in the alignment rank is \( n \), the probability becomes,

\[
P_G(t,n) = 1 - (1 - P_G(t,1))^n,
\]

where \( (1 - P_G(t,1))^n \) is the probability that there is no glitch persisting for a time longer than \( t \).

Hence the probability of module failure due to a glitch in the rank lasting longer than \( t \) is given by:

\[
P_{FF}(t) = P_{FF/G} P_G(t,n)
\]

or,

\[
P_{FF}(t) = P_{FF/G} [1 - (1 - P_G(t,1))^n]
\]

\[
= P_{FF/G} [1 - (1 - P_M \exp(-\frac{t-k}{\tau}))^n].
\]

For \( P_M \ll 1 \),

\[
P_{FF}(t) \approx P_{FF/G} n P_M \exp(-\frac{t-k}{\tau}).
\]

For each clock event the probability of module failure is \( P_{FF}(t) \).

Therefore for \( N \) such clock events, the probability of no module failure is \( (1 - P_{FF}(t))^N \), or, the probability of module failure is,

\[
P_{FF}(t,N) = 1 - (1 - P_{FF}(t))^N.
\]
i.e. \( P_{FF}(t,N) = 1 - (1 - P_{FF/G} n P_M \exp(-\frac{t-k}{\tau}))^N. \)

With \( P_M \ll \frac{1}{n} \exp(-\frac{t-k}{\tau}), \)

\[ P_{FF}(t,N) \approx P_{FF/G} n N \exp(-\frac{t-k}{\tau}). \]  \( \text{(4.3)} \)

Since \( P_{FF/G} \leq 1, \)

\[ P_{FF}(t,N) \leq n N P_M \exp(-\frac{t-k}{\tau}). \]  \( \text{(4.4)} \)

4.5 PERFORMANCE ANALYSIS OF SYNCRO CLOCKED MODULES

As noted in Section 4.2, the failure of a syncro clocked module is mainly due to the delay in the generation of the subsequent clock event each time the alignment rank takes a time longer than prescribed to resolve. In addition, it is the cumulative effect of such delays over some interval that potentially leads to a module failure. Consider the case where the cumulative delay tolerable over \( N+1 \) clock events is \( x. \)

Then the problem of calculating the probability of failure is just that of determining the probability that the \( N+1 \)-st clock event is generated at a time later than \( N T_S + x \) from the instant the first clock event is generated and where \( T_S \) is the nominal clock period of the module.

Each separation between a pair of consecutive clock events is a random variable with a probability density function of the form of equation (3.3) with \( 0 < a < 1 \) and where (3.2) is used for defining \( a. \) The time of generation of the \( N+1 \)-st clock event has a distribution given by the \( N \)-fold convolution of \( f_T(t) \) since the generation time is a random variable which is the sum of \( N \) such separations. For ranges of \( x \) such that \( 0 < ax \ll 1, \) and from Theorem 3.1, the density function for
the generation time can be approximated by the function,

\[ f_T(t,N) = (1-Na)\delta(t-NT_S) + \frac{Na}{\tau} \exp\left(-\frac{t-NT_S}{\tau}\right) u(t-NT_S). \quad (4.5) \]

Hence the probability of a delay in the generation of the \(N+1\)-st clock event by an amount no longer than \(x\), is,

\[ \int_0^{x+NT_S} f_T(t,N) \, dt. \]

The probability that the delay is larger than \(x\), for a given upper bound \(x\), is,

\[ P_S(x,N) = \int_{x+NT_S}^{x+NT_S} f_T(t,N) \, dt \]

\[ = 1 - \int_0^{x+NT_S} f_T(t,N) \, dt \]

\[ = 1 - (1 - Na) - Na(1 - \exp(-\frac{x}{\tau})) \]

\[ = N a \exp\left(-\frac{x}{\tau}\right). \quad (4.6) \]

Hence the probability of module failure, for a given upper bound \(x\) on the tolerable delay, is,

\[ P_{FS}(x,N) = P_{FS/G} P_S(x,N) \quad (4.7) \]

that is,

\[ P_{FS}(x,N) \approx P_{PF/G} d N P_N \exp\left(-\frac{T_S-k}{\tau}\right) \exp\left(-\frac{x}{\tau}\right). \quad (4.8) \]
Letting $C_E = N a$

$$= N \delta_n P_M \exp\left(-\frac{T_S-k}{\tau}\right), \quad (4.9)$$

then,

$$P_F(x, N) \approx P_{FF/G} C_E \exp\left(-\frac{x}{\tau}\right). \quad (4.10)$$

Thus the probability of module failure is proportional to $C_E$, termed the extension coefficient. The larger $C_E$ is, the more likely it is for the system to fail. This has been briefly discussed in Section 3.5.

In general, the coefficient $C_E$ can be used as a single parameter for characterizing the performance of a syncro clocked module. A small value of $C_E$ means a better performance. Note that $C_E$ is also the coefficient of the exponential term in the density function of a syncro clocked module. For the case when $N = 1$, $C_E (= a)$ is the probability that the next clock event will be generated at a time longer than $T_S$ from the current clock event. When $N > 1$, $C_E$ is the probability that the $N+1$-st clock event will be generated at a time longer than $N T_S$ from the first clock event in a train of $N+1$ consecutive clock events. In general, the density function of a syncro clock is of the form:

$$f_T(t) = (1-C_E) \delta(t-NT) + \frac{C_E}{\tau} \exp\left(-\frac{t-NT}{\tau}\right) u(t-NT) \quad (4.11)$$

and the mean extension is $C_E$ which increases linearly with $N$. 

4.6 COMPARING FPC AND SYNCR0 CLOCKED MODULES PERFORMANCE

From the above discussion, it can be seen that FPC and syncro clocked modules employ significantly different synchronization schemes. The intent of the performance comparison to be made in this section is to evaluate the effects of the schemes on module reliability. Using the module failure probability as a performance index, the comparison will allow us to observe changes in module failure probability when one synchronization scheme is replaced by another with everything else in the module remaining the same. The comparison given below looks at the performance of each type of module over a fixed time interval covering some number, \( N \), of clock events. For the FPC module, the prescribed resolution time is assumed to be the same as the clock period. Since the generation of every clock event of the FPC module is never delayed, then \( N \) clock events will always be generated within a time span of \( N \) times the clock period. However for the syncro clocked module, there is a nonzero probability that \( N \) clock events will not be generated within any finite time interval. Since for practical systems, no clock event can be allowed to be delayed forever, there is a finite upper bound on the length of time the module is allowed to generate those \( N \) clock events...

Starting from a general form of module failure probability of both module types, the analysis below gives the ratio of the module failure probabilities. Then restricting the clock period of the FPC module to be the same as the nominal clock period of the syncro clocked module, a numerical example is presented to illustrate the order of magnitude of the parameters involved.
Consider FPC and syncro clocked modules with identical alignment ranks subjected to identical input arrival processes. Let \( x \) be the maximum allowable delay in the generation of the \( N+1 \)-st clock event. The FPC module has a time of \( N(T_F + \frac{x}{N}) \) to generate the \( N \) clock events while the syncro clocked module has to generate the \( N \) clock events within the time interval \( NT_S + x \). Hence the FPC module can run a bit slower with a clock period of \( T_F + \frac{x}{N} \) instead of \( T_F \). This will allow more time for its alignment rank to resolve when marginally triggered. Assuming that a module failure does not stop the clock, the FPC module will always generate \( N \) clock events within \( NT_F + x \) time units. The syncro clock module however has a nonzero probability of not generating \( N \) clock events within \( NT_S + x \) time units.

Hence from equations (4.3) and (4.8), the probabilities of failure for the modules are,

\[
P_{FF}(T_F + \frac{x}{N}, N) \approx P_{FF/G} n N P_M \exp(-\frac{T_F + \frac{x}{N} - k}{\tau}) \quad (4.12)
\]

and,

\[
P_{FS}(x, N) \approx P_{FS/G} d_n N P_M \exp(-\frac{T_S + x - x}{\tau}). \quad (4.13)
\]

Let

\[
K = \frac{P_{FF}(T_F + \frac{x}{N}, N)}{P_{FS}(x, N)} \quad (4.14)
\]

then from (4.12) and (4.13),

\[
K = \frac{P_{FF/G}}{P_{FS/G}} \exp(-\frac{T_F - T_S}{\tau}) \exp(-\frac{x(\frac{1}{N} - 1)}{\tau}) \frac{n}{d_n} \quad (4.15)
\]
For the FPC and syncro clocked modules considered, \( \frac{P_{FF/G}}{P_{FS/G}} \exp\left(-\frac{T_F-T_S}{\tau}\right) \) is fixed. Therefore,

\[
K = K_0 \exp\left(\frac{\tau}{\tau} (1 - \frac{1}{N}) \right) \frac{F}{d_n}
\]

where \( K_0 = \frac{P_{FF/G}}{P_{FS/G}} \exp\left(-\frac{T_F-T_S}{\tau}\right) \).

Taking the logarithm of \( K \),

\[
Y = \ln K = \ln K_0 + \frac{X}{\tau} (1 - \frac{1}{N}) + \ln\left(\frac{F}{d_n}\right) \tag{4.17}
\]

A plot of \( Y \) versus \( \frac{\tau}{\tau} \) is given in Figure 4.1 for the case where there is no delay in the feedback paths, i.e. \( d_n = n \). For the case when \( K_0 > 1 \), \( K \) is greater than one for any \( N \). This means that the FPC module is more likely to fail than the syncro clocked module. For \( N = 1 \), \( K \) is a constant and \( K_0 \) controls which module is more likely to fail. If \( N > 1 \) and \( K_0 \leq 1 \), there is a cross-over value of \( x, x_c \), beyond which the syncro clocked module is less likely to fail, i.e. \( K > 1 \). The plots for \( K_0 = \frac{1}{K_0} \), where \( \alpha \) is less than one, are also included in Figure 4.1. The value of \( x_c \), in general, is given by,

\[
x_c = \frac{\tau}{1 - \frac{1}{N}} \ln L \tag{4.18}
\]

where \( L = \frac{1}{K_0} \geq 1 \), and \( d_n = n \).

When \( d_n \neq n \),

\[
x_c = \frac{\tau}{1 - \frac{1}{N}} (\ln L + \ln\left(\frac{d_n}{n}\right)) \tag{4.19}
\]
$Y = \ln K$

$K_0 = 1, N \to \infty$

$K_0 = 1, N = 1$

$x_c' = -\tau \ln z$

$0 < z < 1$

**Figure 4.1**: Plot of $Y$ versus $\frac{x}{\tau}$
When \( N \) is large, \( x_c \) approaches \( \tau \ln L \). For a given value of \( \tau \), the smaller the value of \( L \) and hence the larger the value of \( K_0 \), the closer is the cross-over point to the origin of the axes. This means that the FPC module deteriorates in performance when compared with the syncro clocked module as \( x \) exceeds \( x_c \).

If an FPC module is "converted" to a syncro clocked module by changing the clock and introducing the feedback paths for the \( M \) signals with the nominal clock period set to \( T_P \), \( K_0 \) is unity. Thus \( x_c \) is zero for this case and the module performance is better for \( N \) larger than one and \( x \) greater than zero. If the delays \( D_M \) and \( D_W \) are included in the analysis, then, for the case when the delays in the path are equal, i.e. \( D_j = D_M \) for all \( j \), \( d_n \) (see Section 3.1) becomes,

\[
d_n = n \exp\left(-\frac{D_W + D_M}{\tau}\right),
\]

and (4.17) becomes,

\[
Y = \ln K_0 + \frac{x}{\tau} \left(1 - \frac{1}{N}\right) - \frac{D_W + D_M}{\tau}.
\]

Thus the plots are shifted downwards by an amount \( \frac{D_S + D_M}{\tau} \). This has the effect of increasing \( x_c \), as can be seen from (4.19) which becomes,

\[
x_c = \frac{\tau}{1 - \frac{1}{N}} \left(\ln L + \frac{D_W + D_M}{\tau}\right)
\]

\[
= \frac{D_W + D_M}{1 - \frac{1}{N}} + \frac{\tau}{1 - \frac{1}{N}} \ln L.
\]

Thus \( x_c \) is increased by \( \frac{D_W + D_M}{1 - \frac{1}{N}} \) and for \( N \) large, the increase in \( x_c \) is approximately \( D_M + D_W \). Similarly if there is a predominant delay in
the feedback paths so that for some \( i \), \( \sum_{j \neq i} \exp\left(\frac{D_i}{\tau}\right) \ll \exp\left(\frac{D_i}{\tau}\right) \), then,
\[
d_n \approx \exp\left(\frac{D_N + D_i}{\tau}\right)
\] (4.23)

and (4.17) becomes,
\[
Y = \ln K_0 + \ln n - \frac{D_N + D_i}{\tau} + \frac{K}{\tau} (1 - \frac{1}{N}).
\] (4.24)

Hence for this case, the plots are shifted downwards by \( \frac{D_N + D_i}{\tau} - \ln n \),
and (4.19) becomes,
\[
x_c = \frac{\tau}{1 - \frac{1}{N}} \left( \ln L - \ln n + \frac{D_N + D_i}{\tau} \right)
\] (4.25)

For \( N \) large, the increase in \( x_c \) over that derived from Figure 4.1 is
approximately \( D_N + D_i - \tau \ln n \).

As an example to illustrate the order of magnitudes of the values
of \( x_c \), consider the case where \( T_F = T_S \), that is, \( K_0 = \frac{P_{FF/G}}{P_{FF/G}} \) and there
is no delay in the feedback paths. For the FPC module, the delay \( x \) is
distributed over \( N \) clock events for the FPC module. Instead of
delaying the generation of each clock event by a fixed amount, \( \frac{x}{N} \), as in
the FPC module, the synchro clocked module allows the clock to adjust its
rate of generation of clock events according to whether a glitch hazard
has occurred or not. Table 4.1 gives the values of \( \frac{x_c}{\tau} \) for various
\( \frac{P_{FF/G}}{P_{FF/G}} \) and \( N \). Figures 4.2 and 4.3 are plots of \( \frac{x_c}{\tau} \) versus \( N \) and versus
\( \ln\left(\frac{P_{FS/G}}{P_{FF/G}}\right) \), respectively. Note that for a given \( \frac{P_{FF/G}}{P_{FF/G}} \), \( \frac{x_c}{\tau} \) is inversely
proportional to \( 1 - \frac{1}{N} \) and for a given \( N \), \( \frac{x_c}{\tau} \) is proportional to \( \ln\left(\frac{P_{FS/G}}{P_{FF/G}}\right) \).
Table 4.1: Values of \( \frac{X_C}{\tau} \) for various \( \frac{P_{FF/G}}{P_{FS/G}} \) and N

<table>
<thead>
<tr>
<th>N</th>
<th>( 10^{-1} )</th>
<th>( 10^{-2} )</th>
<th>( 10^{-3} )</th>
<th>( 10^{-6} )</th>
<th>( 10^{-10} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>2.56</td>
<td>5.12</td>
<td>7.68</td>
<td>15.35</td>
<td>25.58</td>
</tr>
<tr>
<td>( 10^2 )</td>
<td>2.33</td>
<td>4.65</td>
<td>6.98</td>
<td>13.96</td>
<td>23.26</td>
</tr>
<tr>
<td>( 10^3 )</td>
<td>2.30</td>
<td>4.61</td>
<td>6.91</td>
<td>13.83</td>
<td>23.05</td>
</tr>
<tr>
<td>( 10^4 )</td>
<td>2.30</td>
<td>4.61</td>
<td>6.91</td>
<td>13.82</td>
<td>23.03</td>
</tr>
</tbody>
</table>

\( \frac{X_C}{\tau} \) decreases very rapidly from a very large value when \( N = 1 \) to a value close to 1 for increasing \( N \) (for example, \( \frac{X_C}{\tau} = 1.01 \) when \( N = 100 \) and \( \ln(\frac{P_{FF/G}}{P_{FS/G}}) = 1 \) and as \( N \) is increased to 1000, \( \frac{X_C}{\tau} = 1.001 \), see Figure 4.2).

For a given \( N \), \( \frac{X_C}{\tau} \) increases logarithmically with \( \frac{P_{FS/G}}{P_{FF/G}} \). When \( \frac{P_{FS/G}}{P_{FF/G}} \) increases by \( 10^9 \) (from 10 to \( 10^{10} \)), \( \frac{X_C}{\tau} \) only increases by a factor of 10 (from 2.3 to 23) for \( N = 10^4 \). Since \( \tau \) is of the order of nanoseconds (see Table 2.1), \( X_C \) ranges from nanoseconds to tens of nanoseconds. For a typical clock period of the order of tens of nanoseconds, the distributed delay \( (\approx \frac{X}{N} \approx 23 \times 10^{-4} \text{nsec.} \text{ for an extreme case of} \frac{P_{FS/G}}{P_{FF/G}} = 10^{10}) \), is negligible and the tolerable delay is only 23 nsec. over a time period of the order of milliseconds \( (N \text{ clock period} \approx \text{tens of nanoseconds} \times 10^4). \)

Many computations require more than a few milliseconds of processor time, hence a value of \( N = 10^4 \) is not large. Furthermore a tolerable delay of 23 nsec. is fairly small for a time span of the order of milliseconds.
\[ \frac{x_c}{\tau} = \frac{1}{1 - \frac{1}{N}} \quad \text{for } \ln\left(\frac{P_{FF/G}}{P_{FS/G}}\right) = 1, T_S = T_F, \]

and zero feedback delays.

Figure 4.2: Plot of \( \frac{x_c}{\tau} \) versus \( N \) with no delay and \( \ln\left(\frac{P_{FF/G}}{P_{FS/G}}\right) = 1 \)
\[ \frac{x_0}{\tau} = \ln \left( \frac{P_{FS/G}}{P_{FF/G}} \right) \]

\[ T_F = T_S \]

\[ N \rightarrow \infty \]

Figure 4.3: Plot of \( \frac{x_0}{\tau} \) versus \( \frac{P_{FS/G}}{P_{FF/G}} \) with no delay
The tolerable delay is usually larger than $x_c$ resulting in the syncro
clocked module having a lower probability of module failure. Note that
this is despite the fact that the syncro clocked system is $10^{10}$ times more
susceptible (i.e. $\frac{P_{PS/G}}{P_{FF/G}} = 10^{10}$) than the FPC module to a glitch hazard!
However it must be borne in mind that the results have some limitations
due to the assumptions used. These assumptions may not be valid for all
applications. The limitations are discussed in Chapter 6.
5. CHARACTERIZATION OF PERFORMANCE OF CLOCKED SYSTEMS

5.1 INTRODUCTION

A clocked system in general is made up of one or more modules. Its overall performance is dependent on the performance of each of the modules performing its role in the system. This chapter addresses the problem of characterizing such system performance. The performance index to be used is the same as that introduced previously, that is, the probability of a system failure due to glitch hazards in its modules. Due to the complexity of the problem, the analysis is restricted to homogeneous systems, or, systems composed of identical modules. It is shown that a system's performance can be characterized by a set of parameters.

5.2 ROLE OF A MODULE IN A CLOCKED SYSTEM

A module is essentially a processing unit generating and using clock events from a single clock. Each clock event is used to initiate actions which contribute to the system's operation in providing some specified service to its environment. The amount of clock events used (or "consumed") by the module can be viewed as a measure of its contribution to the system's goal in providing the service. It is convenient to specify the goal in terms of the number of clock events required.

Taking the view that clocked systems are used to implement algorithms, then the goal of such systems is to successfully carry out such implementations. An implementation of one such algorithm on a specific system is termed a computation. The number of clock events required by a computation is termed computation cost. As an algorithm must terminate (18), the computation cost is finite and only a finite number of actions need be
initiated. Note that the computation cost is both dependent on the implementation of the algorithm as well as on the system's architecture. Exactly how such factors affect the cost is not of concern here, however, it is important to realize that there is always such a cost for a given computation and a given system.

Within the system, the realization of the computation may be carried out by division of labor - splitting the computation into parts and letting the modules handle one or more of such parts. The role of a module is to ensure that the parts allocated to it are successfully carried out. The collection of parts of a computation that is allocated to a module is termed a job. Correspondingly, the job cost is the number of clock events required for the job. Since only homogeneous systems are considered, the job cost is the same for a given job allocated to any module within the system. Moreover, the computation cost is the sum of job costs of all jobs related to the computation. The overhead involved in the allocation of jobs within the system can be included in the computation cost.

Whether a system successfully completes a computation or not depends very much on the success of every module in the system completing its job... The unsuccessful completion of a computation leads to the system failing to meet its goal. For our purposes, a computation is deemed to be successfully implemented if the time it takes to perform it is less than some specified upper bound.

5.3 PERFORMANCE MEASURE FOR CLOCKED SYSTEMS

Using the term job execution time to mean the time elapsed between the start and end of a job, one can establish the relationship between the time bounds on the job's execution time and the probability of exceeding
it. A module is said to have failed if the time bound is exceeded. The probability of module failure then is very much the same as that developed in Chapter 4.

Defining a system failure as the occurrence of one or more module failures within the system, the probability of system failure can be expressed in terms of the probability of module failure if the job cost and upper bound of the extension of the job execution time for each module are known.

Fundamental cases of system failure are analysed in the following sections. The analysis is carried out for two types of clocked systems - those composed of FPC modules only and those composed of syncro clocked modules only. The assumptions used in Section 4.3 apply together with those discussed in the next section.

5.4 ASSUMPTIONS AND NOTATIONS FOR ANALYSIS OF CLOCKED SYSTEMS

The following are the assumptions used in the analysis:

(i) A clocked system is not reconfigurable i.e. if a system module or its subpart fails, the system does not reconfigure itself in an attempt to recover from the failure.

(ii) Modules within a system form a connected network.

(iii) The arrival processes to the modules are of identical characteristics. Each of the arrival processes is an integer valued independent increment stochastic process.

(iv) A module processes a job without interruptions, that is, there is no module sharing.

(v) A system is made up of identical modules which are either FPC or syncro clocked. No mixture of such modules is allowed.
### Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_i$</td>
<td>Job cost for module i.</td>
</tr>
<tr>
<td>$T_F$</td>
<td>Clock period for homogeneous FPC system.</td>
</tr>
<tr>
<td>$T_S$</td>
<td>Nominal clock period for homogeneous syncro clocked system.</td>
</tr>
<tr>
<td>$T_C$</td>
<td>Computation time.</td>
</tr>
<tr>
<td>$T_{Ji}$</td>
<td>Job execution time for the i-th job.</td>
</tr>
<tr>
<td>$P_{FF_i}(t)$</td>
<td>Probability of failure for the i-th module in an FPC system.</td>
</tr>
<tr>
<td>$P_{FXi/G}$</td>
<td>Conditional probability that module i will fail given that it experiences a glitch hazard.</td>
</tr>
<tr>
<td>$P_{FS_i}(x,N_i)$</td>
<td>Probability that an extension, x, in the job execution time will be exceeded by module i of a syncro clocked system executing a job of job cost $N_i$.</td>
</tr>
<tr>
<td>$F_{X}$</td>
<td>Probability of system failure.</td>
</tr>
<tr>
<td>$M_X$</td>
<td>No. of modules in system.</td>
</tr>
<tr>
<td>$T_{Fi}$</td>
<td>Clock period of module i in FPC system.</td>
</tr>
</tbody>
</table>

* The letter X should be replaced by F or S for FPC or syncro clocked modules respectively.
$T_{S_1}$ Nominal clock period of module $i$ in syncro clocked system.

$C_{E_1}$ Extension coefficient for module $i$ in syncro clocked system.

$X_i$ Maximum tolerable extension in the job execution time for module $i$ of a syncro clocked system.

5.5 PROBABILITY OF FAILURE OF FIXED PERIOD CLOCKED SYSTEMS

For an FPC system with $M$ modules with the $i$-th module assigned a job of job cost $N_i$, the probability that module $i$ will fail due to a glitch hazard lasting longer than $t$ is given by equation (4.3), that is,

$$P_{FF_i}(t) = P_{FF_i}/G_n N_i P_M \exp(-\frac{t-k}{\tau}).$$ (5.1)

Hence the probability of system failure is given by,

$$F_F = Pr[\text{at least one module fails}]$$

$$= 1 - Pr[\text{no module fails}]$$

$$= 1 - \prod_{i=1}^{M_F} (1 - P_{FF_i}(t)).$$

Let the flip-flops be triggered by every clock event generated and $t$ be the net allowable resolution time after each triggering, then, for $P_{FF_i}(t) \ll 1$ for all $i$,

$$F_F \approx \sum_{i=1}^{M_F} P_{FF_i}(t)$$
\[ F_F = \sum_{i=1}^{M_F} P_{FF_i/G} N_1 n P_M \exp\left(-\frac{t-k}{\tau}\right) \]

or,

\[ F_F = \left[\sum_{i=1}^{M_F} P_{FF_i/G} N_1 \right] n P_M \exp\left(-\frac{t-k}{\tau}\right) \tag{5.2} \]

Rewriting equation (5.2),

\[ F_F = P_{IN} C_{SYS} P_R(t) \tag{5.3} \]

where, \[ P_{IN} = P_M, \]

\[ C_{SYS} = n \sum_{i=1}^{M_F} P_{FF_i/G} N_1, \]

and, \[ P_R(t) = \exp\left(-\frac{t-k}{\tau}\right). \]

In equation (5.3), \( P_{IN} \) represents the input arrival process while \( C_{SYS} \) is strictly dependent on the system and module structure, and \( P_R(t) \) is a function of the resolution time, \( t \).

5.6 PROBABILITY OF FAILURE OF SYNCRO CLOCKED SYSTEMS

For module \( i \), the extension coefficient is given by equation (4.9), that is,

\[ C_{Ei} = N_i d_{ni} P_M \exp\left(-\frac{T_S-k}{\tau}\right) \tag{5.4} \]

With \( X_i \) as the upper bound on the extension in the job execution time for module \( i \), the probability that module \( i \) will fail is \( P_{FSi}(X_i, N_i) \), where,

\[ P_{FSi}(X_i, N_i) = P_{FSi/G} C_{Ei} \exp\left(-\frac{X_i}{\tau}\right) \cdot \tag{5.5} \]
Hence the probability of system failure is given by,

\[ F_S = \Pr[\text{at least one module fails}] \]

\[ = 1 - \Pr[\text{no module failure in system}] \]

\[ = 1 - \prod_{i=1}^{M_S} (1 - P_{FSi}(X_i, N_i)). \]

For \( P_{FSi}(X_i, N_i) \) small,

\[ F_S = \sum_{i=1}^{M_S} P_{FSi}(X_i, N_i). \]

That is,

\[ F_S = \sum_{i=1}^{M_S} \frac{P_{FSi/G} C_{Ei}}{C_{Ei}} \exp\left(-\frac{X_i}{\tau}\right) \quad (5.6) \]

or,

\[ F_S = \left[ \sum_{i=1}^{M_S} \frac{P_{FSi/G}}{C_{Ei}} \exp\left(-\frac{X_i - X}{\tau}\right) \right] \exp\left(-\frac{X}{\tau}\right) \quad (5.7) \]

where \( X \) can be the minimum, mean or maximum of \( X_i \) over all \( i = 1, \ldots, M_S \).

That is,

\[ F_S = P_M\left[ \sum_{i=1}^{M_S} \frac{P_{FSi/G}}{C_{Ei}} N_i d_{ni} e^{-\frac{TS}{\tau}} \exp\left(-\frac{X_i - X}{\tau}\right) \right] \exp\left(-\frac{X-K}{\tau}\right) \quad (5.8) \]

\( F_S \), like \( F_F \), can be written in terms of the three parameters, \( P_{IN} \), \( C_{SYS} \) and \( P_R(X) \), that is, with \( P_{IN} = P_M \), \( P_R(X) = \exp\left(-\frac{X-K}{\tau}\right) \), and,

\[ C_{SYS} = \sum_{i=1}^{M_S} \frac{P_{FSi/G}}{C_{Ei}} N_i d_{ni} \exp\left(-\frac{TS}{\tau}\right) \exp\left(-\frac{X_i - X}{\tau}\right). \]
In general,

$$F_S = \left[ \sum_{i=1}^{M_S} P_{FSi/G} N_i \exp\left(-\frac{T_{Si-T_S}}{\tau}\right) d_n \exp\left(-\frac{X_i-X}{\tau}\right) \right] P_M \exp\left(-\frac{T_{S-k}}{\tau}\right) e^{-\frac{X}{\tau}} \quad (5.9)$$

for some fixed constant $T_S$.

With $a' = P_M \exp\left(-\frac{T_{S-k}}{\tau}\right)$,

$$F_S = \left[ \sum_{i=1}^{M_S} P_{FSi/G} N_i \exp\left(-\frac{T_{Si-T_S}}{\tau}\right) d_n \exp\left(-\frac{X_i-X}{\tau}\right) \right] a' \exp\left(-\frac{X}{\tau}\right). \quad (5.10)$$

For the special case where $T_{Si} = T_S$, $d_n = n$ and $X_i = X$ for all $i$,

$$F_S = \sum_{i=1}^{M_S} P_{FSi/G} N_i \ n \ a' \exp\left(-\frac{X}{\tau}\right). \quad (5.11)$$

Equation (5.11) holds for the case when there is no delay in the feedback paths of all of the modules. Rewriting (5.11) and substituting for $a'$,

$$F_S = P_M \left[ \sum_{i=1}^{M_S} P_{FSi/G} N_i \exp\left(-\frac{T_S}{\tau}\right) \right] \exp\left(-\frac{X-k}{\tau}\right), \quad (5.12)$$

and writing in terms of $P_{IN}$, $C_{SYS}$ and $P_R(X)$,

$$F_S = P_{IN} C_{SYS} P_R(X), \quad (5.13)$$

where,

$$P_{IN} = P_M,$$

$$C_{SYS} = \sum_{i=1}^{M_S} P_{FSi/G} N_i \exp\left(-\frac{T_S}{\tau}\right), \quad (5.14)$$

and,

$$P_R(X) = \exp\left(-\frac{X-k}{\tau}\right).$$
Unlike the case of a single clocked module, the probability density function for a syncro clocked system with more than one module cannot be directly derived from equation (5.6). The density function is dependent on the internal structure of the system. This problem is discussed in the next section.

5.7 DENSITY FUNCTIONS FOR SYNCR0 CLOCKED SYSTEMS

The use of density functions for syncro clocked system is convenient if it can be expressed in the form of a TED function. Such a form involves the extension coefficient \( C_E \). Two special cases are considered in deriving expressions for \( C_E \).

CASE I: Modules executing jobs in a sequential manner.

For this case, the jobs are executed one after the other in a sequential manner. For homogeneous systems, this is equivalent to a single module executing the same sequence of jobs. Hence \( N \) in equation (4.9) is replaced by \( \sum_{i=1}^{M_S} N_i \), i.e.,

\[
C_E = \left( \sum_{i=1}^{M_S} N_i \right) a. \tag{5.15}
\]

CASE II: Modules execute jobs in parallel.

Assume that all jobs start at the same time. Let \( N_{max} = \max_i N_i \), \( T_{j_i} \) be the job execution time for module \( i \) and \( T_C \) the computation time (i.e., the time it takes to perform the computation completely). The minimum value for \( T_C \) is \( N_{max} T_S \) and that for \( T_{j_i} \) is \( N_i T_S \).

\[
Pr[T_C < N_{max} T_S] = 0
\]
\[ \Pr[T_C = \max_{i} T_{J_i}] = \Pr[T_{\text{J1}} \leq \max_{i} T_{S_i} \text{ for all } i] \]
\[ = \prod_{i=1}^{M_S} \left(1 - P_{FSi/G} C_{Ei} \exp\left(- \frac{(\max_{i} T_{S_i})}{\tau}\right)\right) \]
\[ \approx 1 - \sum_{i=1}^{M_S} P_{FSi/G} C_{Ei} \exp\left(- \frac{(\max_{i} T_{S_i})}{\tau}\right). \quad (5.16) \]

For any extension \( x \) of \( T_C \) over \( \max_{i} T_{S_i} \),

\[ \Pr[T_C \geq \max_{i} T_{S_i} + x] = 1 - \Pr[T_{J1} \leq \max_{i} T_{S_i} + x \text{ for all } i] \]
\[ = 1 - \prod_{i=1}^{M_S} \left(1 - P_{FSi/G} C_{Ei} \exp\left(- \frac{(\max_{i} T_{S_i})}{\tau}\right) \exp\left(- \frac{x}{\tau}\right)\right) \]
\[ \approx \left[ \sum_{i=1}^{M_S} P_{FSi/G} C_{Ei} \exp\left(- \frac{(\max_{i} T_{S_i})}{\tau}\right) \right] \exp\left(- \frac{x}{\tau}\right). \quad (5.17) \]

With \( C_{EP} = \sum_{i=1}^{M_S} P_{FSi/G} C_{Ei} \exp\left(- \frac{(\max_{i} T_{S_i})}{\tau}\right), \quad (5.18) \]
the density function of \( T_C \) is,

\[ f_{T_C}(t) = (1 - C_{EP}) \delta(t - N_c T_S) + \frac{C_{EP}}{t} \exp\left(- \frac{t - N_c T_S}{\tau}\right) u(t - N_c T_S), \quad (5.19) \]

where \( N_c = \max_i \).

Furthermore, the system failure probability is given by,

\[ F_S = \prod_{i=1}^{M_S} P_{FSi/G} N_1 \exp\left(- \frac{T_{S_i}}{\tau}\right) \exp\left(- \frac{(\max_{i} N_1 T_{S_i})}{\tau}\right) \exp\left(- \frac{x - k}{\tau}\right) \]
\[ = P_{IN} C_{SYS} P_R(x), \]

where \( P_{IN} \) and \( P_R(x) \) are identical to those in (5.13) and \( C_{SYS} \) is the sum involving \( P_{FSi/G} N_1 \), etc. in the equation above.
5.8 PARAMETERS FOR CHARACTERIZING PERFORMANCE OF CLOCKED SYSTEMS

From the various forms of probability of system failure derived above, it can be seen that in general, the probability of system failure, \( P_{SYS} \), is of the form,

\[
P_{SYS} = P_{IN}(w) C_{SYS} P_{R}(t)
\]

where,

\( P_{IN}(w) \) = probability of marginal triggering for an appropriate input process;

\( C_{SYS} \) = a parameter related to the module characteristics, the computation and job costs;

\( P_{R}(t) \) = probability that the output of an M flip-flop does not settle to a logically defined level after a specified time \( t \) from the time of occurrence of the triggering clock event.

For the case of a single M flip-flop after being marginally triggered, \( C_{SYS} = 1 \), \( P_{IN}(w) = P_{M} \) for an integer valued, independent increment process, and \( P_{R}(t) = \exp\left(-\frac{t-k}{\tau}\right) \).

For an FPC system, \( P_{R}(t) \) is the probability that the output of an M flip-flop is logically undefined after a fixed delay \( t \) from the occurrence of a triggering clock event. Hence \( t \) may be \( T_{P} \), the clock period, or any fixed time delay used in the synchronizer. However in a syncro clocked system, \( t \) is a specified upper bound on the extension of the execution time of a computation. Table 5.1 lists the expressions for \( C_{SYS} \) for various types of clocked systems.
Table 5.1: Expressions for $C_{SYS}$

<table>
<thead>
<tr>
<th>System Type</th>
<th>Module Type</th>
<th>$C_{SYS}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>FPC</td>
<td>$P_{FF/G} N_n$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$P_{PS/G} N d_n \exp(-\frac{T_S}{\tau})$</td>
</tr>
<tr>
<td>II</td>
<td>FPC</td>
<td>$\sum_{i=1}^{M_F} n P_{FFi/G} N_i$</td>
</tr>
<tr>
<td></td>
<td>Syncro clock</td>
<td>$\sum_{i=1}^{M_S} P_{PSi/G} d_{ni} N_i \exp(-\frac{T_S}{\tau})$</td>
</tr>
<tr>
<td>III</td>
<td>FPC</td>
<td>$\sum_{i=1}^{M_F} P_{FFi/G} N_i n$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\sum_{i=1}^{M_S} P_{PSi/G} d_{ni} N_i \exp(-\frac{T_S}{\tau}) \exp[-\frac{N_{max}-N_i}{\tau} T_S]$</td>
</tr>
</tbody>
</table>

System types:  
  I. Single module with job cost of $N$,  
  II. $M$ modules with $X_1 = X$ and jobs executed sequentially,  
      and,  III. $M$ modules with $X_1 = X$ and jobs executed in parallel.

From the above table, it can be seen that except for syncro clocked systems of system type III, $C_{SYS}$ has the same form for both FPC and syncro clocked systems i.e. for system of the same system and module types. $C_{SYS}$ takes into account the effect of the size of the alignment rank (in terms of number of $M$ flip-flops), the conditional probability of module failure given that a glitch hazard has occurred in the rank and the job costs.
It is obvious that when \( n, P_{F1/G} \) or \( N_1 \) increases, a larger \( F_{SYS} \) results. Thus the system performance degrades with increasing \( C_{SYS} \).

It is found that the form for \( F_{SYS} \) in equation (5.20) is convenient for characterizing performance of clocked systems. Consider the case of an FPC system with \( M \) identical modules with \( P_{F1/G} = P_{FF/G} \) for all \( i = 1, \ldots, M \) and all modules are subjected to identical Poisson input processes of rate \( \lambda \). Then, \( P_{IN}(\lambda) = e^{-\lambda \lambda_{m}}(e^{\lambda \lambda_{m}} - 1) \) and \( P_{R}(t) = \exp(-\frac{t-k}{\tau}) \) for some fixed \( t \). From Table 5.1, \( C_{SYS} = P_{FF/G} n \sum_{i=1}^{M} N_i \), where there are \( n \) \( M \) flip-flops in each alignment rank. Alternately, \( C_{SYS} = P_{FF/G} n N_C \), where \( N_C = \sum_{i=1}^{M} N_i \) is the computation cost. For \( \lambda \lambda_{m} \ll 1 \),

\[
F_{SYS} \approx \lambda \lambda_{m} P_{FF/G} n N_C \exp(-\frac{t-k}{\tau}).
\]

If the input arrival rate is increased to \( 2\lambda \), then the new \( F_{SYS}' \), \( F_{SYS} \) is given by,

\[
F_{SYS}' = e^{-2\lambda \lambda_{m}}(e^{2\lambda \lambda_{m}} - 1) P_{FF/G} n N_C \exp(-\frac{t-k}{\tau})
\approx 2\lambda \lambda_{m} \left[ P_{FF/G} n N_C \exp(-\frac{t-k}{\tau}) \right]
= 2 F_{SYS}'.
\]

The probability of system failure now becomes twice the original \( F_{SYS} \). On the other hand if the arrival rate stays the same and the tolerable resolution time is increased from \( t \) to \( t' = t + x \) for some \( x \), real, then the probability of system failure \( F_{SYS}'' \) becomes,

\[
F_{SYS}'' = \left[ 1 - \exp(-\lambda \lambda_{m}) \right] P_{FF/G} n N_C \exp(-\frac{t-k}{\tau})
= \exp(-\frac{x}{\tau}) \left[ \left( 1 - \exp(-\lambda \lambda_{m}) \right) P_{FF/G} n N_C \exp(-\frac{t-k}{\tau}) \right]
= \exp(-\frac{x}{\tau}) F_{SYS}'.
\]
If \( x > 0 \) then increasing the tolerable resolution time by an amount \( x \) results in an exponential decrease in the probability of system failure.

In both of the cases in the above example, only one parameter is affected - \( P_{IN}(w) \) for the first case and \( P_{R}(t) \) for the second. Other uses of the three parameters include observing changes in \( P_{M} \) or \( P_{R}(t) \) when different \( M \) flip-flops are used, thus changing \( w, k \) and \( \tau \), or for performance comparison between systems of identical and different types. Chapter 6 has two more sophisticated examples.

Another important use of \( F_{SYS} \) is in performance characterization at a higher level, that is, that of a network of clocked systems. Each clocked system is described by its own \( F_{SYS} \) and if there are \( J \) clocked systems in the network then the probability of failure (using a performance index analogous to those described earlier) of the network, \( F_{NET} \), is given by,

\[
F_{NET} = 1 - \prod_{i=1}^{J} (1 - F_{SYSi})
\]

\[
\approx J \sum_{i=1}^{J} F_{SYSi} \quad (5.21)
\]

for \( F_{SYSi} \ll 1 \) for all \( i \), and where \( F_{SYSi} \) is the probability of failure of the \( i \)-th system.

With this general form, one can evaluate the performance of a complex system where its subsystems are not all of identical types or are subjected to different input processes. For example the complex system may be made up of both PPC and syncro clocked subsystems, the \( M \) flip-flops used may be different from subsystem to subsystem and the Poisson input processes for the subsystems may have different arrival rates.
6. CONCLUSION AND FUTURE RESEARCH

6.1 INTRODUCTION

Before concluding the thesis, examples illustrating the use of the results in Chapter 5 are presented. Following the examples is a discussion on the limitations of the results. These limitations are imposed by the assumptions used in the derivation of the results. The conclusion of the thesis is then presented, followed by suggestions for future research.

6.2 APPLICATIONS OF THE RESULTS - EXAMPLES

The following two examples illustrate how the results of Chapter 5 can be used. The first deals with FPC systems while the other is for syncro clocked systems. Some aspects of the first example can also be used for syncro clocked systems.

6.2.1 Determining Maximum Resolution Time For FPC Systems

Consider the situation where the system designer is interested in determining the maximum effective resolution time that is required for a given system with a specified failure probability. The computation cost of a typical computation that the system performs is $10^9$ clock events. Thus for a clock period of the order of tens of nanoseconds, the typical computation time is of the order of tens of seconds, if there is no parallelism in the system when performing the computation. The designer wishes to ensure that the system failure probability will not exceed $10^{-9}$. This means that a typical computation should not experience a system failure rate of more than one in a billion. Input events have a Poisson distribution with a mean arrival rate of $10^5$ sec.\(^{-1}\). Hence each input path to a module in the system is subjected to a hundred thousand input events per second, on the average, or equivalently, the
average time between input event arrivals is 10 microseconds. The system has 5 identical FPC modules, each with 50 independent input paths. All the modules cannot tolerate any glitch hazard at all i.e. a module failure results whenever it experiences a glitch hazard. From equation (5.20),

$$F_{\text{SYS}} = P_{\text{IN}}(w) C_{\text{SYS}} P_{R}(t).$$

The constraint on $F_{\text{SYS}}$ is,

$$F_{\text{SYS}} \leq 10^{-9}.\]$$

For Poisson arrival processes with a rate of $\lambda$,

$$P_{\text{IN}}(w) = 1 - e^{-\lambda w} \approx \lambda w, \text{ for } \lambda w \ll 1.$$  

From Table 5.1 for system type II,

$$C_{\text{SYS}} = \sum_{i=1}^{M_{F}} P_{F_{1}/G} N_{1} n_i.$$  

As, $P_{F_{1}/G} = 1$ (the system is highly susceptible to glitch hazards),

$$M_{F} = 5, \text{ } n = 50, \text{ and hence, } \sum_{i=1}^{5} P_{F_{1}/G} N_{1} = \sum_{i=1}^{5} N_{1} = 10^9, \text{ since } \sum_{i=1}^{5} N_{1} \text{ is the computation cost.}$$

Thus,

$$C_{\text{SYS}} = 10^9 \times 50 = 5 \times 10^{10}.\]$$

With $P_{R}(t) = \exp\left(- \frac{t-k}{\tau}\right)$,

$$F_{\text{SYS}} = \lambda w C_{\text{SYS}} \exp\left(- \frac{t-k}{\tau}\right),$$

(6.1)

i.e. $w \exp\left(- \frac{t-k}{\tau}\right) = \frac{F_{\text{SYS}}}{\lambda C_{\text{SYS}}}.$
That is, \[ \frac{1}{w} \exp\left(\frac{t-k}{\tau}\right) = \frac{\lambda C_{\text{SYS}}}{F_{\text{SYS}}} \] (6.2)

With \( F_{\text{SYS}} \leq 10^{-9} \),

\[ \frac{1}{w} \exp\left(\frac{t-k}{\tau}\right) \geq \frac{\lambda C_{\text{SYS}}}{10^{-9}} = \frac{10^5 \times 5 \times 10^{10}}{10^{-9}} = 5 \times 10^{24}. \]

Taking logarithma,

\[ \frac{1}{\tau}(t-k) - \ln w \geq \ln(5 \times 10^{24}) = 56.87. \]

Let \( P_{\text{SYS}} = \frac{C_{\text{SYS}}}{F_{\text{SYS}}} \), then,

\[ \ln P_{\text{SYS}} = \frac{1}{\tau}(t-k) - \ln w \geq 56.87. \]

Curves of \( \ln P_{\text{SYS}} \) versus the effective resolution time, \( t \), are plotted in Figure 6.1 for the three types of flip-flops given in Table 2.1.

It can be seen from the plots that for \( \ln P_{\text{SYS}} = 56.87 \), \( t = 58 \) nsec. for the SN74S74, 74 nsec. for the MC10131 and 85 nsec. for the MC1016. Hence if the designer has a choice of using any of the three types of flip-flops, the SN74S74 gives the minimum value of \( t \) to achieve the same system failure probability. For \( t \geq 59 \) nsec., the SN74S74 flip-flop gives system failure probability that is better than the value specified for the system. It is interesting to note that the plots in Figure 6.1 completely characterize the flip-flop types. The slope of the straight line for a given flip-flop type is \( \frac{1}{\tau} \). The plots can be used for determining which flip-flop type is best for a given \( t \). For this example, the SN74S74 flip-flop is the best type to use. If \( \ln P_{\text{SYS}} \) is 20, then the MC10131 should be used instead as the resolution time needed is only about 12 nsec. while \( t \) for the SN74S74 is 18 nsec. and that for the
Figure 6.1: Plots of $\ln P_{SYS}$ versus $t$ for three flip-flop types
MC1016 is 20 nsec. For \( \ln P_{\text{SYS}} = 17.5 \) the MC10131 is still the best followed by the MC1016 and then the SN74S74. Hence the type of flip-flop to be used varies with \( P_{\text{SYS}} \) for a given \( t \).

Summarizing the above discussion, plots similar to those in Figure 6.1 can be used to evaluate the effect of flip-flop type on system performance. Hence the designer can select the appropriate flip-flop type given \( P_{\text{SYS}} \) and \( t \), or, the minimum value of \( t \) required for a given flip-flop type to achieve a given value of \( P_{\text{SYS}} \). The plots can be used to minimize \( t \), that is to select the best flip-flop type for the lowest value of \( t \) that will give the desired \( P_{\text{SYS}} \). In this case, \( t \) and the flip-flop type are not specified. An example is the case where the designer is interested in minimizing \( t \) so that the clock rate can be as high as possible.

For syncro clocked systems, a similar approach can be used to determine the best flip-flop type to use. However, instead of using \( t \) on the horizontal axis of Figure 6.1, the maximum allowable extension, \( X \), can be used. \( P_{\text{SYS}} \) for syncro clocked systems is of the same form as equation (5.20) except that \( P_{R}(X) \) is used instead of \( P_{R}(t) \). From Table 5.1, it is observed that the nominal clock period \( T_S \) is embedded in \( C_{\text{SYS}} \) and hence \( T_S \) has to be specified before Figure 6.1 can be used. Note that \( X \) is the maximum delay over the minimum computation time that can tolerated. Very often \( X \) is specified and the designer has to determine the value of \( T_S \) to be used to achieve a given system failure probability. The value of \( X \) may be fixed by the refresh time of dynamic logic elements used inside the system, or the environment may have a threshold value for \( X \), such that if it is exceeded, the environment would declare that the system is malfunctioning. The latter occurs in the case where the system is part of a larger network of systems which frequently perform checks on the status.
of each other in the network. The next example illustrates the use of $F_{SYS}$ to determine an optimal value of $T_S$.

6.2.2 Determining The Value of $T_S$ For Syncro Clocked Systems

As before an upper bound on $F_{SYS}$ is specified together with the arrival rate $\lambda$. The computation cost and module structure are also specified. The designer has to ensure that the system will not fail more often than the specified failure rate. The intent here is to design a syncro clock for the system with an optional value of $T_S$. For the syncro clocked system, equation (5.20) becomes,

$$F_{SYS} = P_{IN}(w) C_{SYS} P_{A}(X).$$

Note that from Table 5.1, only $C_{SYS}$ is dependent on the value of $T_S$.

For the first two system types, $C_{SYS}$ can be written in the form:

$$C_{SYS} = K_{SYS} \exp\left(- \frac{T_S}{\tau}\right), \quad (6.3)$$

where,

$$K_{SYS} = \begin{cases} 
  P_{FS/G} \, N \, d_n, & \text{for system type I,} \\
  M_3 \\
  \sum_{i=1} P_{FS_i/G} \, d_{ni} \, N_i, & \text{for system type II.} \quad (6.4)
\end{cases}$$

For system type III with $N_i$ identical for all $i$, $C_{SYS}$ is of the same form as (6.3) with $K_{SYS}$ having the same form as for system type II.

Hence,

$$F_{SYS} = \lambda w \, K_{SYS} \exp\left(- \frac{T_S}{\tau}\right) \exp\left(- \frac{X-k}{\tau}\right), \quad (6.5)$$

i.e.,

$$w \exp\left(- \frac{T_S + X-k}{\tau}\right) = \frac{F_{SYS}}{\lambda K_{SYS}}, \quad (6.6)$$
Letting \( u = T_S + X \),
\[
\frac{1}{w} \exp \left( \frac{u-K}{\tau} \right) = \frac{\lambda K_{SYS}}{F_{SYS}}.
\] (6.7)

Note that (6.7) is of the same form as (6.2). Hence letting
\[
P'_{SYS} = \frac{\lambda K_{SYS}}{F_{SYS}},
\]
the plots of \( \ln P'_{SYS} \) versus \( u \) are similar to those in Figure 6.1. The designer needs to know the value of \( \lambda \), \( K_{SYS} \) and \( F_{SYS} \) to compute \( P'_{SYS} \) which is then used to determine the optimum value of \( u \), or for selecting the best flip-flop type for a given \( u \), or for determining the best value of \( u \) for a given flip-flop type. The procedures for all three are the same as those described earlier for the FPC system. Knowing \( u \), \( T_S \) can be determined since \( X \) is usually specified. For most applications involving system type III of Table 5.1, there is usually a dominant job which has a large job cost and hence a large job execution time. Since jobs are executed in parallel, all other jobs have a higher probability of completing within the specified time. Figure 6.2 illustrates the case for 3 jobs executed in parallel.

\[\text{JOB 1} \quad \longrightarrow \quad (N_2-N_1)T_S + X \quad \longrightarrow \]
\[\text{JO}B \quad 2 \quad \longrightarrow \quad X \quad \longrightarrow \]
\[\text{JOB} \quad 3 \quad \longrightarrow \quad (N_2-N_3)T_S + X \quad \longrightarrow \]

\[N_{\text{max}} = N_2 \]
Computation time \( \leq N_2 T_S + X \)

Figure 6.2: Job execution times for three parallel jobs
The effective allowable extensions in the job execution times for jobs 1 and 3 are large and hence the probability of completing the jobs within the time bound of $N_2T_5X$ is very close to 1. Thus the system can be treated as though it has only one module, the dominant module, which affects the system reliability. This case is similar to that for system type I and thus the same procedures can be used for determining $T_S$.

The above examples have served to illustrate the use of $F_{SYS}$ for system design. It must be noted that there are limitations to the use of $F_{SYS}$. These are discussed in the next section.

6.3 LIMITATIONS OF THE RESULTS

One of the limitations of the results is that the input processes have to be stationary independent increment processes. This means that the case where input arrivals are not independent, is not handled. An example of such a case is the transmission of data in bit-parallel, byte-serial fashion i.e. the 8 bits of a byte are transmitted in parallel and the bytes are transmitted one after another in a sequential manner. Even though the 8 bits of a byte are sent simultaneously, their arrival times at the destination clocked system vary due to differences in path delays. If the bits are clocked into an alignment rank in parallel, then the input arrival process to the clocked system is no more an independent increment process. There is a special situation which can be dealt with. When the bytes are sent in a manner described by a stationary independent increment process, the arrival process of each byte is stochastic. If a transmission scheme is used such that only the arrival of a byte is sensed instead of the arrival of individual bits, then the input process to the alignment rank is a stationary independent increment process.
Even when the arrival of data or input events, which are not necessarily independent, can be mapped into a single input arrival event to the alignment rank, the results still assume that all of the input or data events thus obtained are identical stochastically i.e. they are all described by the same stochastic process. This is not always true in practice for example in cases where the input arrival processes are all Poisson but have different arrival rates. In many instances, it is possible to obtain a realistic system failure probability. For example in using the worst case design strategy, the Poisson arrival rate used could very well the worst case arrival rate.

It is assumed in the derivation of the results that the flip-flops in the alignment rank have identical characteristics i.e. identical $k$, $\tau$, $w$, and $d$. This may not necessarily be true in complex systems which use many flip-flop types. Furthermore, the characteristics vary from flip-flop to flip-flop of the same type. Very often an approximation is made, in which case the values of $k$, $\tau$, $w$, and $d$ used are typical values for the flip-flop type.

The TED function is used in the analysis of syncro clocked system performance. The N-fold convolution function of the TED function is approximated by another TED function. There is an error bound on the approximate convolution function used (see Theorem 3.1). One must always be aware of the error bound and its effect on the results. For many cases, the error introduced may be negligible but in other cases, the error may be significant. If the error is found to be too large, one should
use the exact form of the convolution function. The approximate function is convenient to use and manipulate. It is useful for many cases where the error involved is tolerable.

One factor that leads to a small error bound in the N-fold convolution function of a TED function is the low probability of marginal triggering. For a Poisson input arrival process, a low probability of marginal triggering means a low arrival rate compared to the clock rate. Thus the results can only be used when this is true. In cases where the arrival rate is high, the results should not be used.

In the analysis, systems are assumed to be homogeneous. This restricts the applicability of the results to such systems only. The analysis of more complex systems with a mixture of FPC and syncro clocked modules would involve treating the system as a network of clocked systems in a manner similar to that discussed in the latter part of Section 5.8.

In addition to restricting clocked systems to be homogeneous systems, the analysis assumes that there is no module sharing i.e. a job is assigned exclusively to a module which executes the job to completion without interruptions. In systems with the capability of performing multiple computations concurrently e.g. in a multiprogramming system, a module can be assigned more than one job from different computations. The results of this thesis cannot be used for such systems.

Apart from the above limitations, there are a number of other limitations which are discussed in Section 6.5 containing suggestions for future research. Before going into the areas for future research, the conclusions of the thesis are presented.
6.4 CONCLUSION

The thesis has dealt with the problem of characterizing the performance of clocked systems in an environment where the synchronization of external events is essential. The first of the two main conclusions of the thesis is concerned with the comparison of the performance of syncro clocked and FPC modules. The second conclusion is related to the characterization of the performance of clocked systems.

It is found that the performance of a syncro clocked module degrades as the delays in the feedback paths (from the M outputs of the alignment rank to the WAIT input of the syncro clock) increase. Furthermore when compared with the FPC module of a similar internal structure, there is a critical value of the extension in the computation time such that if the upper bound on the extension that can be tolerated is greater than the critical value, the syncro clocked module has a lower failure probability. For those cases where the delays in the feedback paths of the syncro clocked module can be represented by a single delay, the critical value of the extension in computation time increases with the magnitude of the representative delay.

With regard to the performance characterization of clocked systems, it is found that a clocked system's performance is affected by three factors:

(i) the stochastic process describing the input arrival to the alignment ranks of the system,

(ii) the computation cost and system characteristics,

and, (iii) the behavior of a flip-flop when it is marginally triggered.
The first of the factors gives the probability of marginal triggering in the system while the third involves the use of the flip-flop parameters $k$ and $\tau$ to describe the probability of having a logically undefined output after some fixed interval from the triggering clock event (or the probability of exceeding some upper bound on the computation time in the case of the syncro clocked module or system). The second factor gives the effect of the system structure expressed in terms of the computation cost, system size (in terms of the number of modules in the system) and number of flip-flops in each alignment rank, on system performance. It is found that increasing any of these three system parameters results in a degradation of system performance as it would mean that there is more opportunity for marginal triggering to occur leading to a higher probability of system failure. The effects of all the factors can be conveniently grouped into three parameters allowing one to see variation in system performance as the parameters change.

Another important result of the thesis is the use of the coefficient of extension, $C_E$, to characterize performance of syncro clocked systems. A larger $C_E$ means a poorer performance. In general, $C_E$ is the area under the exponential tail of the truncated exponential density function for the system. A larger area or $C_E$ means that the probability of an extension in the computation time is higher. It is found that $C_E$ takes into account the first two factors described above including the effect of nominal clock period of the syncro clock used. The expression for $C_E$ for two special cases of syncro clocked system has been derived in the thesis. It is found that increasing the computation cost, or number of modules (or flip-flops) results in a larger $C_E$. Furthermore, if the system is homogeneus,
the mean extension over the nominal computation time is \( C_{\text{r}} \) where \( r \) is the time constant of the exponential probability function for the flip-flop used.

The performance parameters obtained are useful for estimating system performance. They can be used in the design and performance evaluation of clocked systems (see Subsections 6.2.1 and 6.2.2) and networks of such systems especially in VLSI systems. There are however still some areas that need to be explored further. These are discussed in the next section.

6.5 FUTURE RESEARCH

Some areas for future research are discussed below:

(i) It is felt that the probability distribution function for no resolution of a bistable device should be generalized further. Due to the use of the exponential approximation to the function, the resolution time cannot be less than \( k \), a device dependent constant. A more complete distribution function is found in (10) covering all values of resolution times. It would be helpful if there are experimental curves that cover the whole range of resolution times. The resultant density function will be more complex and analysis of system performance made more difficult.

(ii) The thesis has dealt with the performance of non-reconfigurable (static) systems. It would be interesting to compare the performance of a reconfigurable system with that of a static system. The effect of additional clock events required to reconfigure the system as a result of a module failure should be included in the analysis. There are a number of problems in this area. One needs
to have a good understanding of the effect of scheduling policy on system behavior. For example, if a module fails, how is its job going to be distributed among the remaining healthy modules. Is the added communication as a result of this redistribution of the job (or jobs) going to affect the rest of the system? How much of the uncompleted job (or jobs) should be redistributed? Does the failure of a module change the tolerable extension in the computation time, that is, does the user accept a degraded system performance automatically once it is known that there is a module failure in the system? It may be possible to analyse a specific case given sufficient details but the analysis of a more general system is not so straightforward.

(iii) As cautioned in (19), general purpose simulators are not appropriate for simulating bistable devices operating near the metastable region. A special purpose simulator designed for such an application would be extremely useful. Such a simulator should allow the user to specify the internal structure of the bistable device in sufficient detail and generate results of acceptable accuracy. Perhaps special numerical techniques can be used. These may require data structures similar to those in FORMAC (20) which is used for formula manipulation and which allows numbers of up to 2000 (approx.) digits in length to be used. The simulator may be designed to generate probability functions instead of just mere numbers. Such functions would be useful for experimental verification purposes. A further extension in the simulator may include capabilities for computing probability of failure for systems. The results of this thesis may be useful in this respect. If the simulator has some capability of dealing
with functions of variables like FORMAC, the generation and manipulation of TED functions will be greatly facilitated. Curves for the error bound used in Theorem 3.1 can be generated over the range of extension of interest. One can then get a more accurate error bound. Other desired capabilities are facilities for specifying input processes to bistable devices, simulating system clocks (especially syncro clocks) and describing bistable devices probabilistically instead of through its internal structure. Perhaps a data base of models for commonly used synchronization techniques and strategies should be included. Such a simulator may have the capability for handling reconfigurable systems.

As speeds of computer systems increase, their clock speeds may be forced to increase, leading to more stringent constraints on synchronization. The results presented here would become useful as an aid to the design of faster and more reliable systems. With more research in the above areas, the techniques described can become useful for the design of even more complex systems.
7. ACKNOWLEDGEMENT

The author is indebted to his thesis advisor, Professor J. R. Cox, Jr., for his invaluable advice and able supervision of the thesis work. He has on countless occasions served as a patient listener, as well as an observant critic. The author also wishes to thank the other members of the thesis committee, namely, Professors D. L. Elliott, T. Kimura, and D. F. Wann, for their suggestions and criticisms in the preparation of the thesis draft. A special word of thanks is reserved for Mr. T. J. Chaney and Mr. M. J. Stucki for their help and for the numerous discussion sessions which have been very useful in the development of the thesis. Finally, the author is very indebted to his wife, Ellen, for her unfailing support and encouragement.
8. APPENDIX
APPENDIX 8.1

Derivation of the Glitch Window Width from Experimental Data

Let \( P_G(t) \) be the probability of the resolution time being longer than \( t \). From (7,13), \( P_G(t) \) is given by,

\[
P_G(t) = \frac{T_0}{\tau} \exp(-\frac{t'}{\tau}), \quad t' > k,
\]

\[
= \frac{T_0}{\tau} \exp(-\frac{k}{\tau}) \exp(-\frac{t-k}{\tau}) u(t-k), \quad (8.1)
\]

where \( T_0, k, \tau, \) are constants and \( T \) is the clock period.

From Definition 2.1 and equations (2.4) and (2.5),

\[
P_G(t) = P_M \int_{t}^{\infty} p(t) \, dt \quad (8.2)
\]

where \( P_M \) is the probability of marginal triggering for a given clock event.

Assuming that the time of occurrence of a data (or external) event is uniformly distributed within a clock period,

\[
P_M = \frac{W}{T}.
\]

Equating (8.1) and (8.2),

\[
\frac{W}{T} u(t-k) \exp(-\frac{t-k}{\tau}) = \frac{T_0}{\tau} \exp(-\frac{k}{\tau}) \exp(-\frac{t-k}{\tau}) u(t-k),
\]

i.e.,

\[
w = T_0 \exp(-\frac{k}{\tau}).
\]
2. BIBLIOGRAPHY


10. VITA

Biographical items on the author of the thesis, Mr. Y-P. Lim,

1) Born in Malaysia on October 24, 1952.

2) Attended the University of Malaya, Kuala Lumpur, Malaysia from May, 1972 to June, 1976. Received the degree of Bachelor of Engineering in Electrical Engineering with First Class Honors in June, 1976.

3) Process Engineer, Motorola Malaysia (Semiconductor Group) in Malaysia, from April, 1976 to August, 1976.

4) Awarded first prize in the IEEE Region X Student Paper Contest, 1976.

5) Attended Washington University from September, 1976 to the present date. Received the degree of Master of Science in Electrical Engineering in May, 1978. Research assistant, Department of System Science and Mathematics from September 1976 to December 1976 and Department of Computer Science from January 1977 to June 1979.

6) Membership in Professional Societies: Student Member of I.E.E.E. and A.C.M.; Associate Member of the Institute of Electrical Engineers, England.