FAULT TOLERANT CLOCKING SYSTEM

by

Tzu-I Jonathan Fan

Thesis submitted to the Graduate Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Electrical Engineering

APPROVED:

F.G.Gray, Chairman

J.R.Armstrong

I.M.Besieris

May, 1978

Blacksburg, Virginia

Acknowledgements

The author would like to express his appreciation to Dr. F. G. Gray, Dr. J. R. Armstrong and Dr. I. M. Besieris for serving on his advisory committee. Special thanks are extended to Professor F. G. Gray, for his encouragement and guidance throughout the research and to Dr. R. A. Thompson for his help with some aspects of the research.

This research was funded by the National Science Foundation under Grant MCS 75-06543 AD1. This funding is hereby gratefully acknowledged.

The author wishes to extend a special note of thanks to his parents for their patience and encouragement and to his wife for her help.

Table of Contents

Chapter 1 General Concept

1.1 Introduction

Since the early 1960's there has developed a great interest in the subject of reliability of digital systems and, with the increased complexity of current digital systems, this subject has become increasingly important.

A clock signal is essential to the operation of most digital systems. Distribution of a correct clock signal in a large system is an important element of reliability. This thesis will propose solutions to the problem of distributing a synchronized fault-tolerant clock signal to all elements of large LSI implementations. A particular application to array structures is described. The first chapter is an introduction to the subject matter and general concepts. The second chapter describes an interesting idea given by Fletcher [7]. The rest of this thesis is devoted to extensions, modifications, and improvements of Fletcher's results.

Any discrepancy between the expected output and actual output of a system is said to constitute an error, the cause of which is said to be a physical fault. Systems which do not produce errors in the presence of physical defects are said to be fault tolerant. A typical clock signal waveform is shown in Fig. 1-1. The parameter T (on) is the amount of time during which the signal is in the 1 state. Meanwhile, T (off) is the amount of time during which the signal is in the 0 state. The definition of synchronization between two different clock signals is that the difference between the time of occurrence of their leading (falling) edge is within the tolerance of the digital system.

1.2 Delay Characteristics

Delay is the essential faulty characteristic of clock signals. Many circuit elements, such as delay lines and multivibrators, are important contributors of delay in the clock signal. We will describe it in this section.

Some devices have significant output response rise (o tc 1) and fall (1 to 0) times which are mainly due to electrical parameters such as capacitance. This kind of delay is called <u>rise-fall delay</u>.

Every element introduces delay to the clock signal propagating through it. The delay through an etched conductor is typically 1 ns/foot. The delay through a NAND gate is 5 ns and the delay through a flip-flop is 15 ns (for high speed logic). This kind of delay is called <u>transport delay</u>.

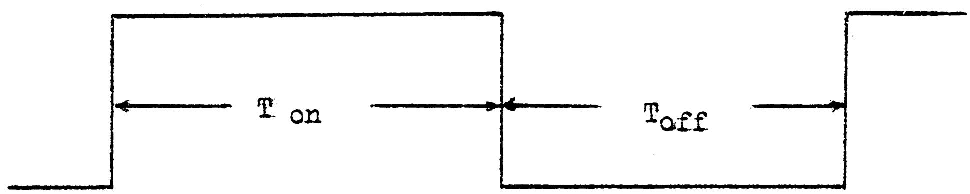


Figure 1-1 Typical Clock Signal

Sometimes the exact transport delay through an element is not known. For example, the delay through a NAND gate may vary from 5 ns to 10 ns, depending on the real situation. This kind of delay is called ambiguity delay.

All elements require energy in order to switch state. The energy in a clock signal is a function of its amplitude and duration. If the duration is too small, the clock signal will not force the element to switch states. The minimum duration for which an input change must persist in order for the element to switch states is called inertial delay.

1.3 General Approach

The general approach to building a fault-tolerant clock system is to have some extra spare clock modules in addition to active modules as shown in Fig. 1-2. There are also switching networks and clock signal checkers in this system. The clock signal checker will monitor the outputs of the active modules. On occasion of incorrect output, the switching network then replaces the faulty module by switching in a spare module. We will introduce some clock signal checkers in this section.

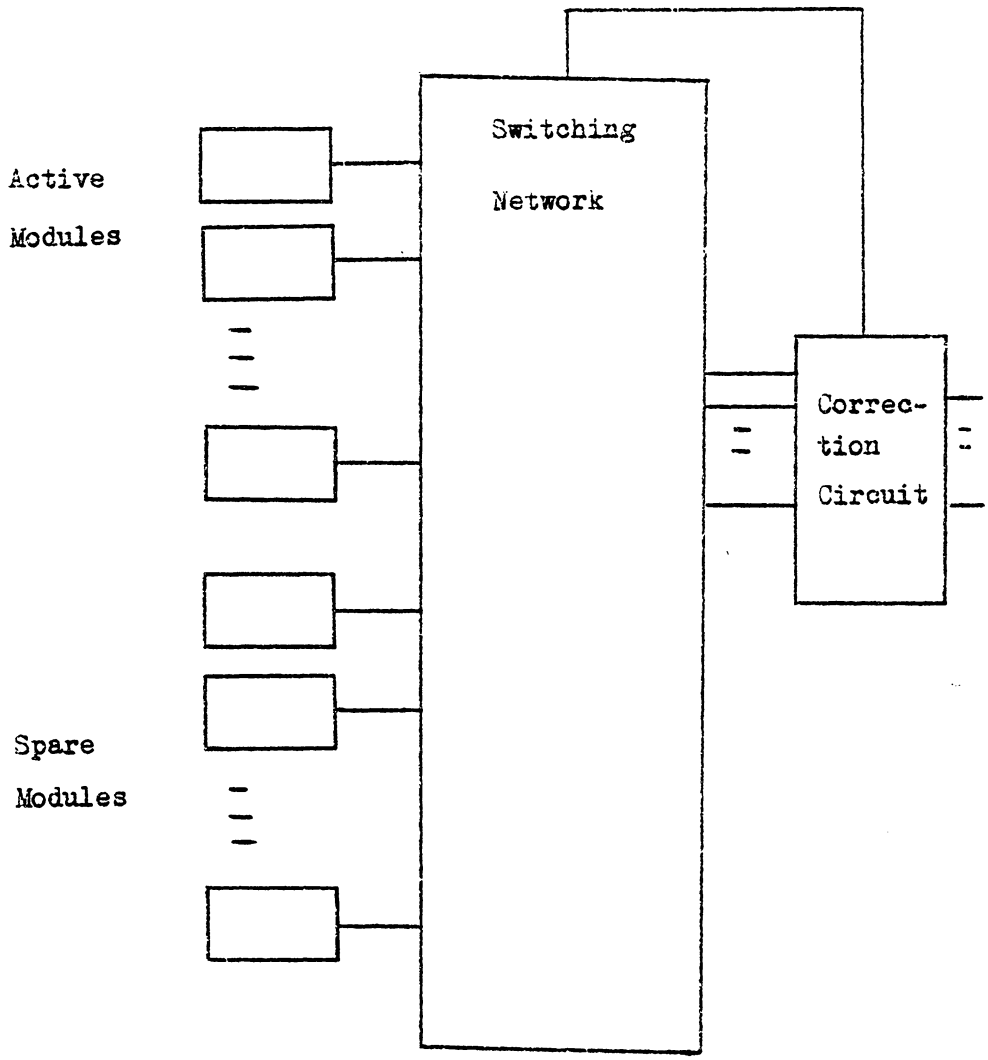


Figure 1-2 General Fault-Tolerant System

Earlier clock signal error detectors used capacitor charge discharge methods [1,2,3]. A typical example is shown in Fig.1-3. The signal to be tested is applied to the input port. When the signal is at 0 , Transistor Q1 is off and Capacitor C can charge through resistor R2 . When the signal is at 1 , Transistor Q1 is on and Capacitor C will be discharged. If the input signal is normal , the output will be at the high state . Otherwise the output will go low . The detection threshold is adjusted by the RC time constant .

Chang et al. [5] designed a detector with an integrator circuit. The detector is shown in Fig. 1-4 . The signal to be tested is connected to the input port of the low-pass RC circuit whose time constant is much larger than the period of the input signal. The dc output voltage of this integrator is the average value of the input which is equal to the product of its duty cycle and its high voltage value. By checking the average value, any variation in the duty cycle will be detected .

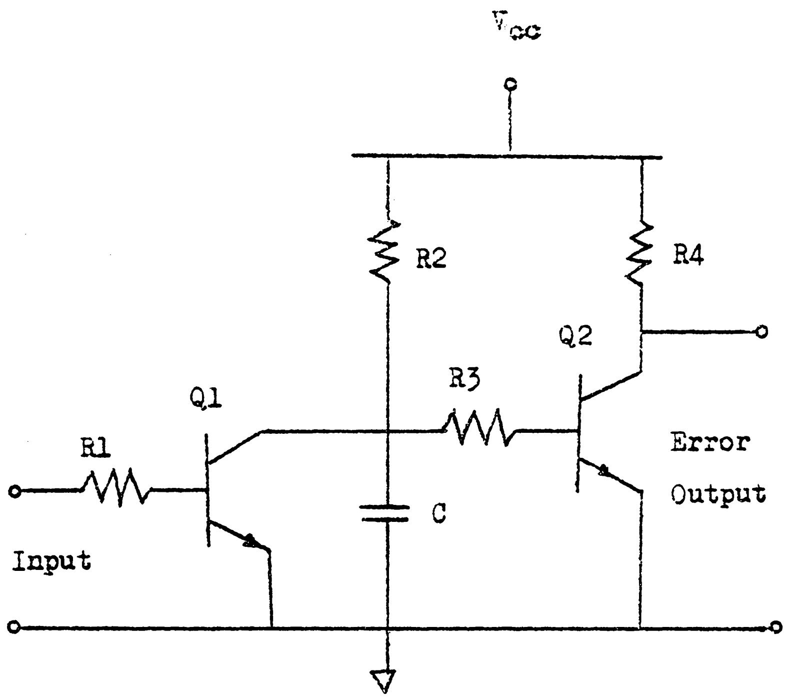


Figure 1-3 Capacitor Discharge Circuit

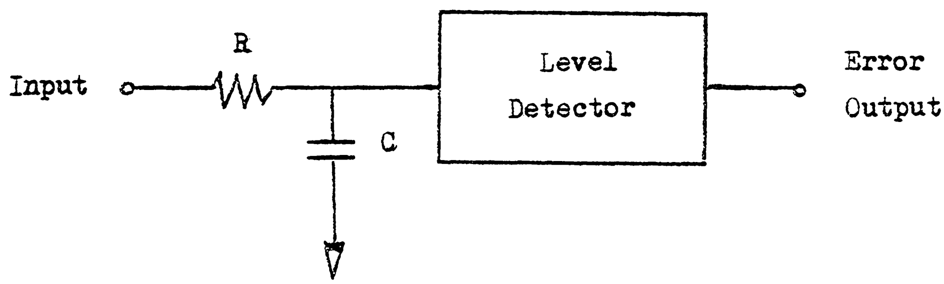


Figure 1.4 Integrator Circuit

Koczela [4] used a digital counter circuit for the detector , which is shown in Fig. 1-5 . The signal to be tested is connected to the reset input of the counter. Another pulse source provides the count signal. If enough count pulses are received between reset operations, a carry output will be produced, indicating an error signal. We can test a specific signal period by using a variable modulus counter.

The methods described above cannot detect errors inside the detector. For example, if the output of the detector has a stuck-at type failure, it will always give the false indication that the signal is correct. Usas [6] suggested another detector which is called a totally self-checking periodic signal checker.

A circuit is fault-secure for a set of faults F , if for any fault in F and any allowable input, the output is a non-code word or the correct code word, not an incorrect code word. A circuit is self-testing for a set of faults F if, for any fault in F, there exists an allowable input which detects it.

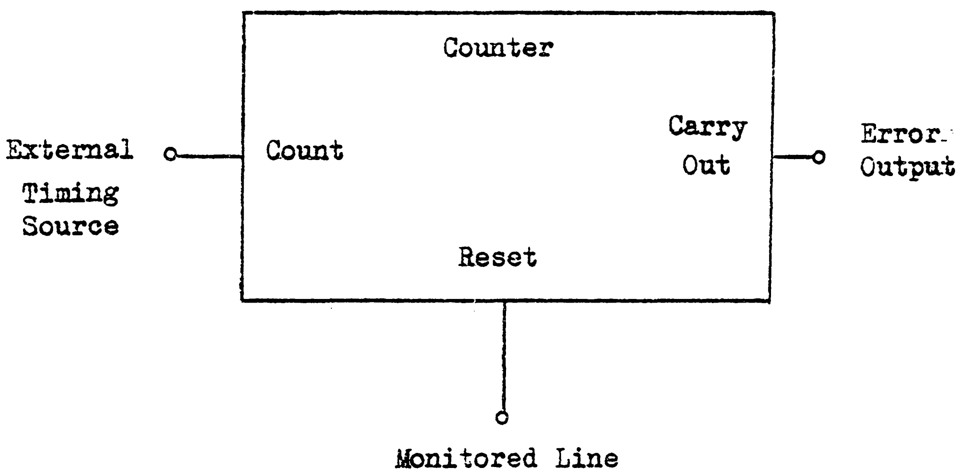


Figure 1.5 Digital Counter Circuit

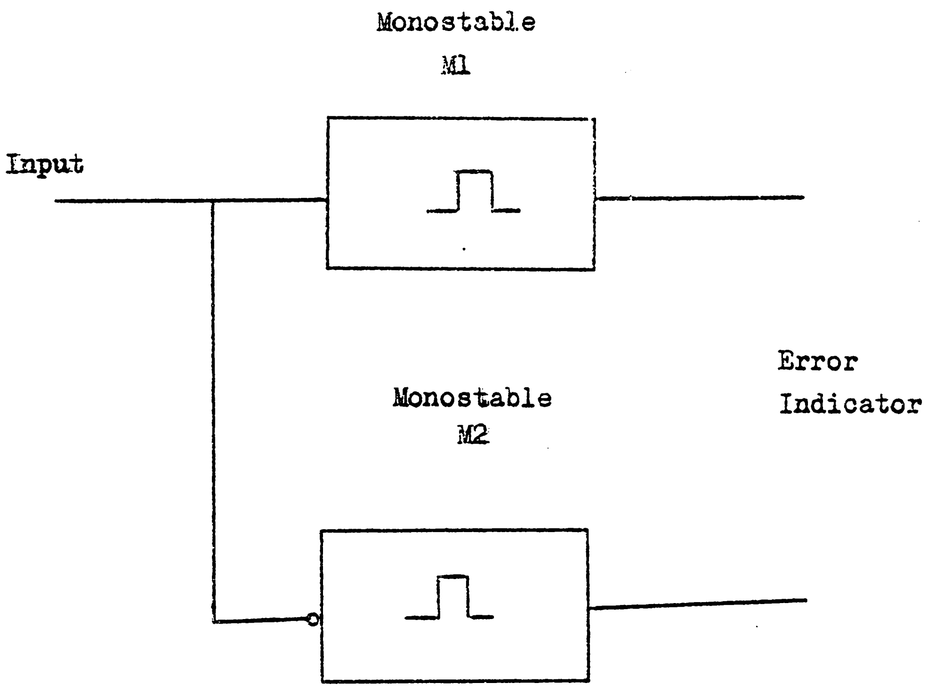


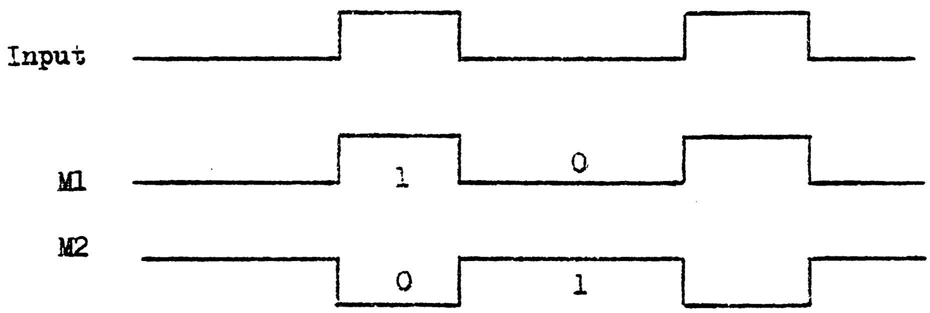
Figure 1.6 Totally Self-Checking Periodic Signal Checker

A totally self-checking circuit is both fault secure and self-testing. Fault-secureness points out that the circuit is operating correctly if the output is a code word. Self-testing insures that any fault can be detected during normal operation. The diagram of this checker is shown in Fig. 1-6. It consists of 2 monostable multivibrators M1 and M2 . Monostable M1 is triggered on the leading edge of the input signal and outputs a pulse of fixed width equal to the expected value of T (on). The second monostable M2 is triggered on the falling edge of the input and produces a pulse of width T (off). The checker used a 1-out-of-2 code . The correct output is 01 or 10. The appearance of a 00 or 11 output indicates that either the input is in error or there is a fault in the checker.

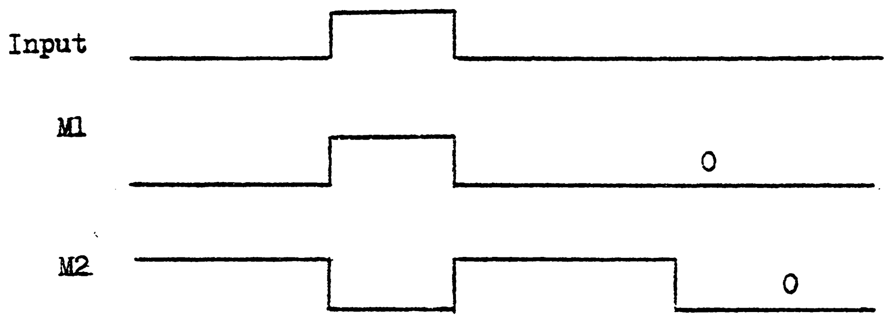
Fig. 1-7 shows how the checker works. Part (a) is the normal operation in which the output of M1 is a regeneration of the input waveform, while M2 is the complement of the input. The checker output is then either 01 or 10 . Part (b) is the case when the input is stuck-at-0. Part (c) is the case when the input is stuck-at-1. In part (d) , there is an abnormal increase in T (off). It can be seen that the checker produces a 00 noncode output which indicates error ocurrence. In part (e) , there is a decrease in T (on) of the input. The checker will generate a 11 noncode output which indicates error occurrence.

By adjusting the monostable pulse width to the values T (on) and T (off) for the clock signal, we can detect any variations in the waveform. The output of the checker could be used to initiate spare elements. This checker can also be used to monitor the synchronization among different clock signals. By combining the signals involved, Chang showed that we could get synchronizatin information for these clock signals.

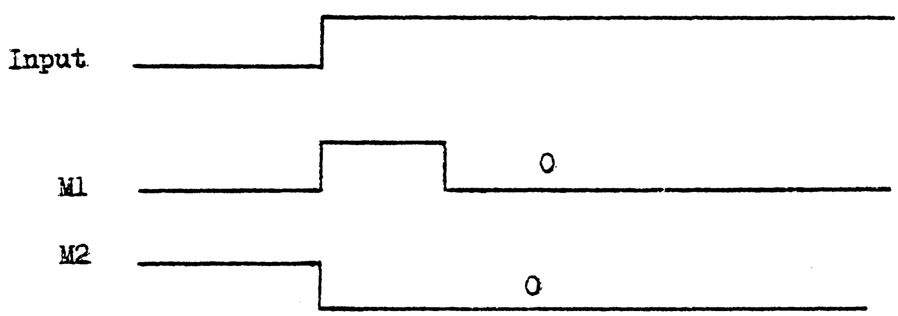
One shortcoming of this fault tolerant clock system is that the switching network in Fig. 1-2 must be highly reliable. Therefore the reliability of the switch has a great effect on the overall system reliability . With this consideration, we will introduce a highly reliable fault tolerant clock system in the next chapter.



a. Normal Operation

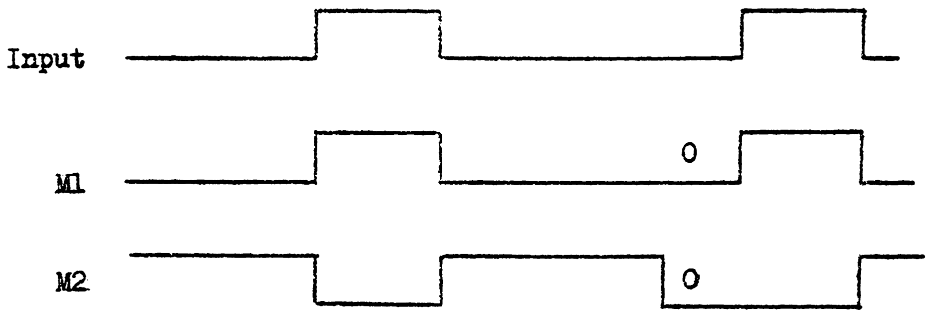


b. Input Stuck-At-0

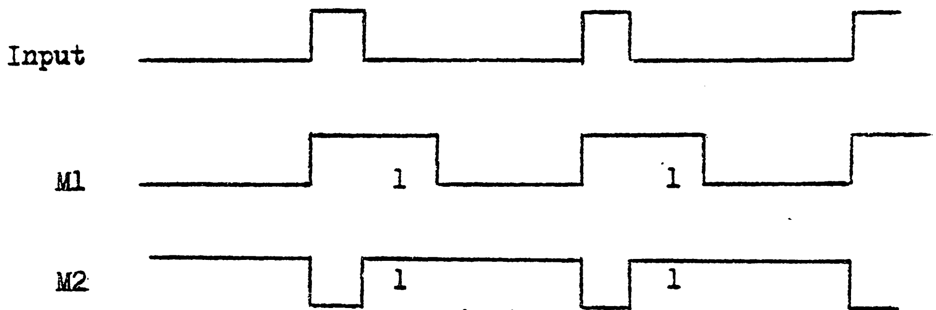


C. Input Stuck-At-1

Figure 1.7 Checker Waveforms



d. Increase in t_{off}



e. Decrease in t_{on}

Figure 1.7 Checker Waveforms (Cont.)

Chapter 2 Fletcher's Clock System

2.1 Introduction

Fletcher's clock patent [7] is an apparatus which provides a clock signal despite failure of individual clock elements (5). This patent has 3f+1 clock elements ,f being any positive integer. It is formed so that the output clock signal is generated notwithstanding failure of any f clock elements. Fig. 2-1 is a circuit diagram of an individual clock element of the patent. Fig. 2-2 is an overall diagram of the patent. It contains 3f+1 clock elements E(1),E(2)...E(3f+1). Each of these clock elements from E(1) through E(3f+1) forms an output clock signal A(1) through A(3f+1) respectively. Each of the clock elements, E(1) through E(3f+1), further receives as input signals the output signals A(i) from each of the other clock elements, as well as its own output signal.

Each of the clock elements includes a quorum logic circuit L (which forms two control signals responsive to the signals received from the individual clock elements), two time delay circuits (Δt), two differentiator circuits (set and reset) and an output circuit. The output circuit of each element is actually a set-reset flip-flop. It will be set and reset in response to the output of the differentiator.

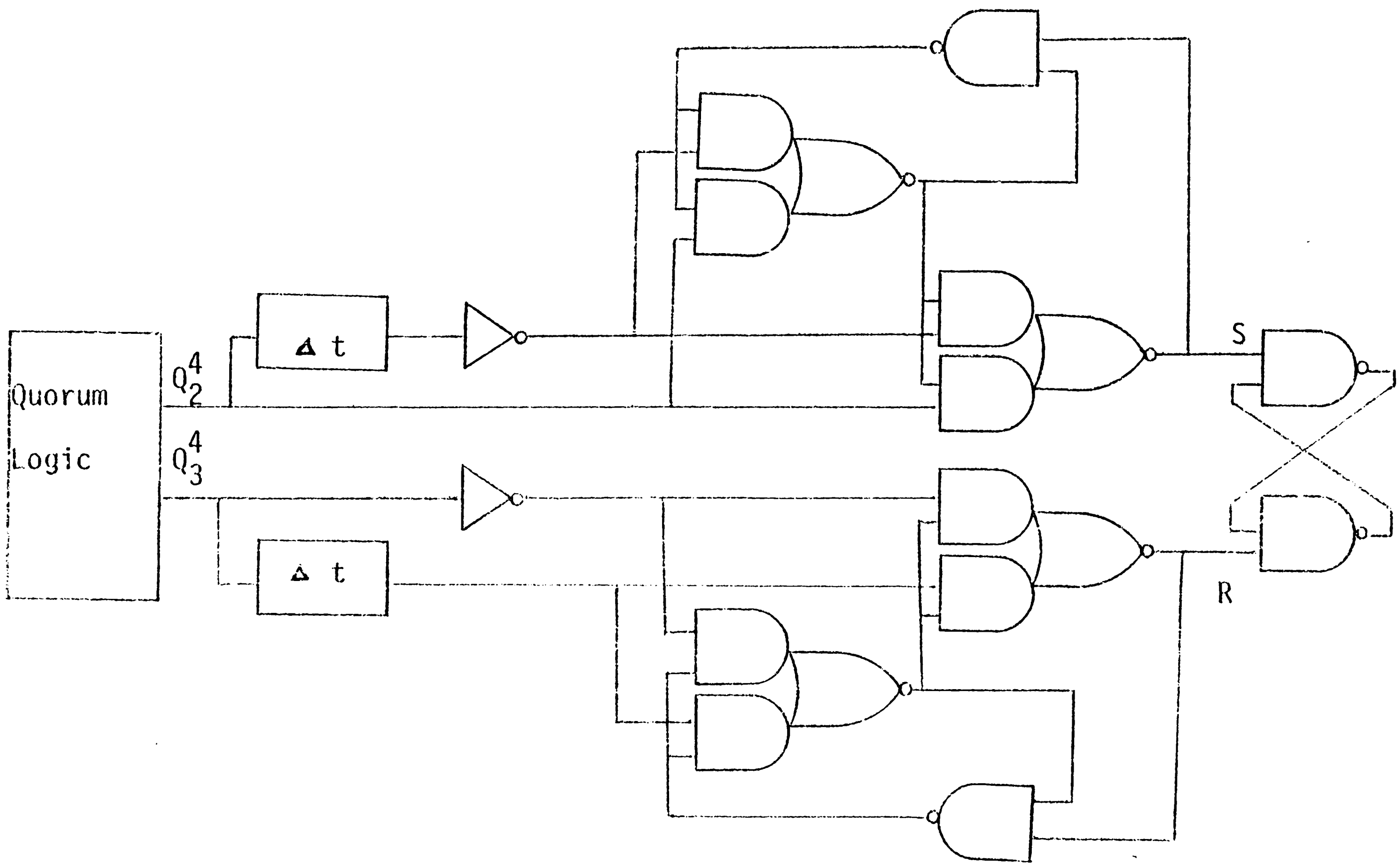


Figure 2-1 Clock Element (letting f be 1)

(patent number 3900741)

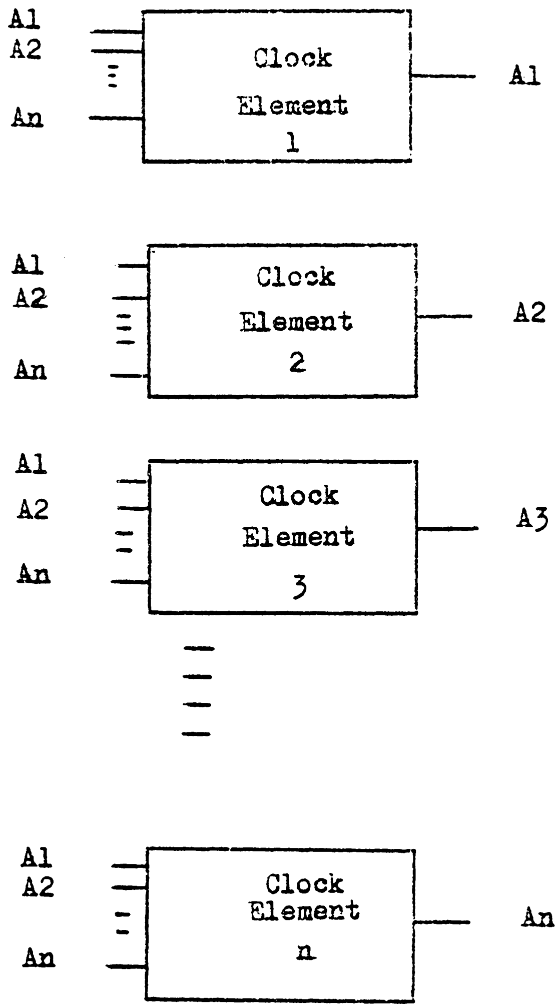


Figure 2-2 Overall Diagram

Definition: A <u>quorum logic function Q(n,i)</u> is a function of n inputs. If less than i of those n inputs are logic level 1 , then this function is logic level 0 ; otherwise it is logic level 1.

The quorum logic circuit L of each element will produce two quorum logic functions, Q(3f+1,f+1) and Q(3f+1,2f+1). If less than f+1 of the 3f+1 output signals A(1),A(2)...A(3f+1) are logic level 1, the function Q(3f+1,f+1) is logic level 0. If less than 2f+1 of the 3f+1 output signals A(1),A(2)...A(3f+1) are logic level 1 , the function Q(3f+1,2f+1) is logic level 0. Those two signals will pass through the time delay circuits and differentiators to set or reset the flip-flop at the proper time. Those two time delay circuits within the clock elements are used to determine the frequency of the clock signal. The top one will determine the length of the time period during which the clock signal is in the 0 state. The bottom one determines the length of the time period during which the clock signal is in the 1 state.

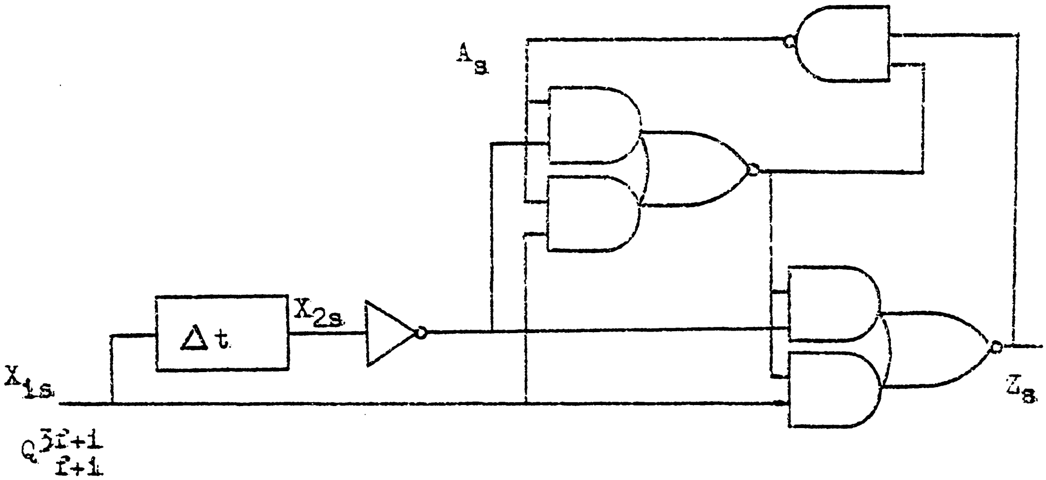
2.2 Circuit Description

Fig.2-3 is the circuit diagram of a set differentiator which is part of the circuit of each clock element.

$X(1s)X(2s)$ are two inputs of the circuit. $A(s)$ is the internal state variable. $Z(s)$ is the output variable. $X(1s)$ is actually the output of the function $Q(3f+1,f+1)$. $X(2s)$ is also the output of the function $Q(3f+1,f+1)$, except that there is a time delay Δt between $X(2s)$ and $X(1s)$.

The state diagram of this circuit is also shown in Fig.2-3 . There are 4 stable states which are enclosed within circles. When inputs $X(1s)X(2s)$ are 00,11,or 10, the internal stable state $A(s)$ will be 1. When $X(1s)X(2s)$ is 01, $A(s)$ will be 0.

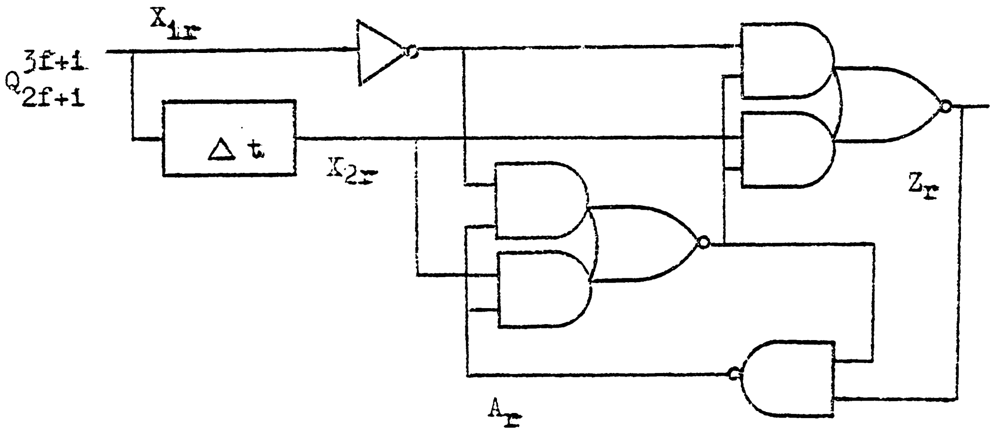
When $X(1s)X(2s)$ go from 01 to 00, the internal state $A(s)$ will go from 0 to 1, the output $Z(s)$ will go from 1 to 0, then return to 1, generating a negative pulse. This is the case when $Q(3f+1,f+1)(t)$ is still 0, but $Q(3f+1,f+1)(t-\Delta t)$ undergoes a 1 to 0 transition. Since $Q(3f+1,f+1)=0$ means less than f+1 clock elements are in 1 state, therefore the clock elements will be set to the 1 state Δt time units after more than 2f clock elements make a 1 to 0 transition.



a_s \ $x_{1s}x_{2s}$	00	01	11	10
0	1/0	0/1	1/0	1/0
1	1/1	0/1	1/1	1/1

A_s/Z_s Asynchronous Flow Table

Figure 2-3 Set Circuit Description



a_r \ $x_{1r}x_{2r}$	00	01	11	10
0	1/0	1/0	1/0	0/1
1	1/1	1/1	1/1	0/1

A_r/Z_r Asynchronous Flow Table

Figure 2-4 Reset Circuit Description

When X(1s)X(2s) go from 01 to 11, A(s) will go from 0 to 1, Z(s) will go from 1 to 0, then return to 1, generating a negative pulse. Since X(1s) is the output of the function Q(3f+1,f+1), X(1s)=0 means less than f+1 clock elements are in the 1 state, and X(1s)=1 means more than f clock elements are in the 1 state. Therefore, whenever more than f clock elements make a 0 to 1 state transition, this clock element will also make a 0 to 1 transition.

Fig. 2-4 is the circuit diagram of the reset differentiator. It is also part of the circuit diagram of each clock element. This figure also shows the state table of the circuit. Z(r) is the output variable. A(r) is the internal state variable. X(1r) and X(2r) are the two input variables of the circuit. There are 4 stable states enclosed within the circles. When input X(1r)X(2r) is 00,01, or 11, the stable state value of A(r) is 0. Otherwise it is 1.

When X(1r)X(2r) goes from 10 to 00, A(r) goes from 0 to 1, output Z(r) goes from 1 to 0, then returns to 1, generating a negative pulse, which will be used to reset the respective flip-flop. This is the case when Q(3f+1,2f+1) goes from 1 to 0. This is the case when Q(3f+1,2f+1)(t-Δt) was still 0, but Q(3f+1,2f+1) goes from 1 to 0. Since

$Q(3f+1,2f+1)=0$ means less than $2f+1$ clock elements are in the 1 state, it must be concluded that the clock element will be reset to the 0 state when more than f clock elements go from 1 to 0.

When $X(1r)X(2r)$ goes from 10 to 11, $A(r)$ goes from 0 to 1, output $Z(r)$ goes from 1 to 0, then returns to 1, generating a negative pulse, which will also be used to reset the respective flip-flop.

Since $X(1r)$ represents the value of $Q(3f+1,2f+1)$, $X(1r)=1$ means more than 2f clock elements are in the 1 state. Because $X(2r)$ represents the value of $Q(3f+1,2f+1)$ with Δt time delay, $X(2r)=0$ means less than $2f+1$ clock elements were in the 1 state (prior to $t-\Delta t$). Therefore, t time units after more than 2f clock elements go from 0 to 1, the output variable $Z(r)$ will give out a negative pulse to reset the respective clock element.

2.3 Operation Rule

From the above description, we can draw the following conclusion about the circuit operation rule of the clock element: The clock element will be set to the 1 state Δt time units after more than 2f clock elements go low or when more than f clock elements go high. The clock element will be reset to the 0 state Δt time units after more than 2f clock elements go high, or when more than f clock elements go low.

With the above operation principle, this system can keep all clock elemenmts in a synchronized condition. It can generate 3f+1 synchronized clock signals. The time when each clock element go high (low) is completely determined by the two functions $Q(3f+1,f+1)$ and $Q(3f+1,2f+1)$. Whenever the number of failed clock elements is less than f+1, regardless of the failure mode, these two functions still work effectively. The rest of the system still generates synchronized clock signals. It is an f-fault tolerant clock system.

In order to demonstrate this property, consider the case where f=1. Since 3f+1=4, there are 4 clock elements synchronized and, if any one of them fails, the remaining 3 are still to be correct synchronized clock signals.

Each clock element will generate a clock signal. This signal will be connected to the input port of each other element and its own input port. So each element will have 4 input signals. Each element has quorum logic which will generate the 2 functions Q(4,2) and Q(4,3), which are the functions of the 4 output signals of all 4 clock elements. When more than 1 element is in the 1 state, Q(4,2) will be 1. Otherwise it is 0. When more than 2 clock elements are in the 1 state, Q(4,3) will be 1. Otherwise it is 0.

Whenever more than 1 clock element goes high (Q(4,2)=X(1s) goes from 0 to 1), all 3 other elements will go high. All clock elements will also go high Δt time units after more than 2 clock elements go low, (Q(4,2)=X(1s) goes from 1 to 0). Whenever more than 1 clock element goes low (Q(4,3)=X(1r) goes from 1 to 0),all 3 other elements will go low. All elements will go low Δt time units after more than 2 clock elements go high (Q(4,3)=X(1r) goes from 0 to 1).

Let Fig. 2-5 be the output waveforms of these 4 clock elements with this diagram. We can give a detailed description of the relation among functions Q(4,2),Q(4,3), variables X(1r), X(2r), A(r), A(s) Z(r), X(s) and the state transition of the clock elements.

E(1) goes low before time T(1), E(2) goes low at time T(1). So does X(1r) (Q(4,3)). Right after T(1),A(r) of E(3) and E(4) will go high, Z(r) of E(3) and E(4) will give out a negative pulse. E(3) then goes low at T(2). The difference between T(1) and T(2) is pretty small (just due to an internal gate delay.). At this time, 3 elements have already gone low; therefore, Q(4,2) and X(1s) will go low. Q(4,2),Q(4,3),X(1r),X(1s) are all 0 at t(3).

At t(2)+Δt, all clocks are scheduled to go high. Let X(2s) of E(3) go low first. Then A(s) of E(3) goes high, and Z(s) gives out a negative pulse to set E(3) to 1 state. Assume E(2) goes high next in the same way at T(4). Then Q(4,2) and X(1s) go high. Shortly thereafter, E(1) and E(4) go high and Z(s) gives out a negative pulse. Assume E(1) goes high at T(5) (The difference between T(4) and T(5) is pretty small). Then Q(4,3) and X(1r) go to 1 because 3 clock elements have already gone high. Δt time units after T(5), X(2r) of E(2) goes high, which causes A(r) go high and Z(r) to give out a negative pulse and reset E(2).

Let E(1) be stuck at 0. At T(2), both E(2) and E(3) are in the 0 state. Then Q(4,3) will go low, which will cause E(4) to go low immediately. Δt time later, E(3) will go high. At T(4), both E(2) and E(3) are in the 1 state, thus Q(4,2) will go high, which will cause E(4) to go high immediately.

Therefore, even if E(1) fails, the other 3 elements

E(2), E(3), E(4) will still be synchronized. They still generate 3 synchronized clock signals with the same frequency as before. This is called a 1-fault tolerant system.

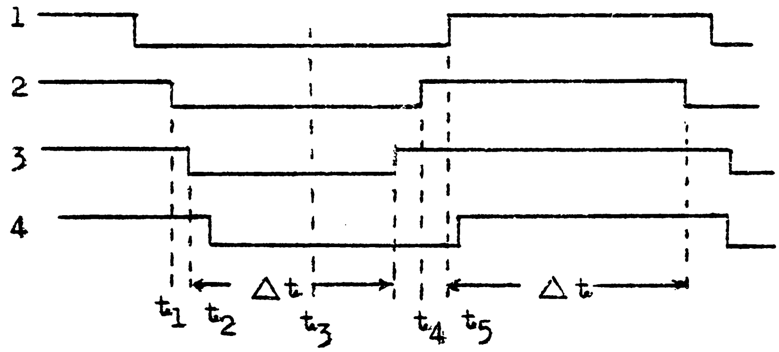


Figure 2-5 Circuit Operation

Chapter 3 Multi-order Clock

3.1 1st order clock

Fletcher's clock patent can generate 3f+1 synchronized clock signals with 3f+1 clock elements. Also, it has the f-fault tolerant capability. For this patent, the expression for the number of inteconnections among different clock elements is (3f+1)*(3f+1) if 3f+1 clock elements are to be generated. For f=1, 16 interconnections will provide 4 clock signals. for f=5, 256 interconnections will provide 16 clock signals. For f=33, 10000 interconnections will provide 100 clock signals. Notice that the number of interconnections is exponentially increasing.

A different concept of interconnection is now suggested. In Fletcher's patent, every clock element is connected to every other element, which is the main cause of the tremendous number of interconnections. If the system is divided into several groups, each group being internally connected, and every group being connected to every other group, then synchronized clock signals with fewer interconnections can be obtained. One kind of connection is stated as follows: Suppose we have (3f+1)*(3f+1) clock elements. Then if we divide them into (3f+1) groups, each group will have 3f+1 clock elements. Every clock element is connected with every other clock element within its own group. Each group will generate 3f+1 clock signals. There are 3f+1 groups. Each clock signal will be connected to a different one of the 3f+1 groups. The circuit structure of each clock element is shown as Fig. 3-1. It is similar to the circuit of Fletcher's patent with the following modifications: Additional quorum logic, additional differentiators for local signals, and additional AND gates between differentiators and flip-flops. Each clock element has two types of input signals, one type coming from clock elements in its own group and the other type coming from other groups. The signals coming from their own group are termed 'local signals ' and the signals coming from other groups are termed 'global signals'. 'Local quorum logic' is quorum logic operating on local signals coming from the 3f+1 clock elements within the same group. 'Global quorum logic' is quorum logic for global inputs. Each of the 3f+1 inputs comes from a different group. There is a differentiator between the global quorum logic and the flip-flop. Its circuit and state table are shown in Fig. 3-2. Depending on the state transitions at its input, it will generate a negative pulse to set or reset the flip-flop. Local quorum logic generates the Q(3f+1, f+1) and Q(3f+1,2f+1) functions as before.

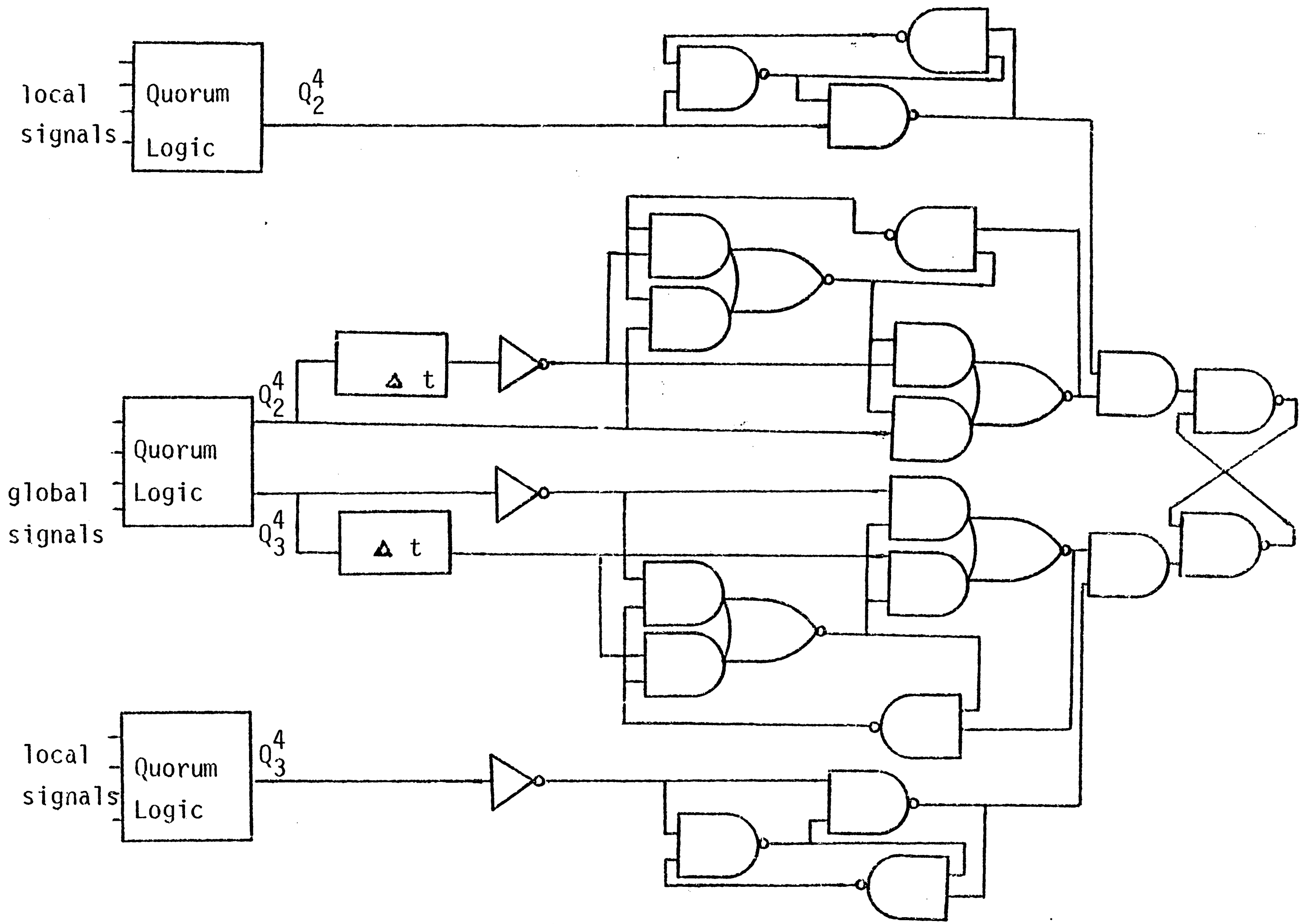
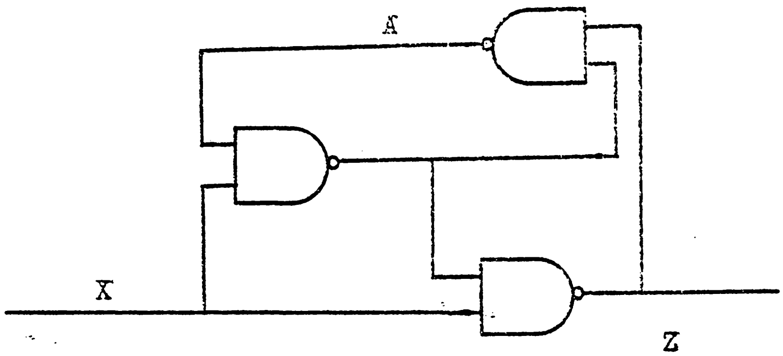


Figure 3-1 Suggested Clock Element (letting f be 1)



a \ x	0	1
0	0/1	1/0
1	0/1	1/1

Figure 3-2 Local Synchronizer

When more than f clock elements go logic high, Q(3f+1,f+1) will go logic high, and the respective differentiator will generate a negative pulse, which will pass through an AND gate and set the trailing flip-flop. When more than f clock elements go logic low, Q(3f+1,2f+1) will go logic low, and the respective differentiator will generate a negative pulse, which will pass through an AND gate and reset the trailing flip-flop. Therefore, each group will be locally synchronized as in the original patent. Global quorum logic will also generate Q(3f+1,f+1) and Q(3f+1,2f+1) functions. When more than f group outputs go logic high, Q(3f+1,f+1) will go logic high, and the respective differentiator will then generate a negative pulse, which will pass through an AND gate to set the trailing flip-flops. When more than f group outputs go logic low, Q(3f+1,2f+1) will go logic low, then the respective differentiator will generate a negative pulse which will pass through a respective AND gate to reset the trailing flip-flop. When more than 2f group outputs go logic low, Q(3f+1,f+1) will go logic low; the transition will pass through a Δt delay circuit; then the respective differentiator will generate a negative pulse, which will pass through the respective AND gate to reset the flip-flop. When more than 2f group outputs go logic high, Q(3f+1,2f+1) will go logic high; the state transition will pass through Δ

t time delay, then the respective differentiator will generate negative pulse, which will pass through the respective AND gate to set the flip-flop.

From the above circuit description it is known that: When more than f groups go logic high (low), the other groups will go logic high (low) immediately. No group can go logic high (low) until Δt time units after more than 2f groups have gone low (high).

Now suppose f=1, then 3f+1=4, and (3f+1)*(3f+1)=16. There are 16 clock elements, which are divided into 4 groups with 4 elements in each group. Each element is connected to 4 elements in its own group and one different group. So each element has 4 'local input signals' and 4 'global input signals'. Local quorum logic will generate Q(4,3) and Q(4,2) functions of the 4 local input signals. Global quorum logic will generate Q(4,2) and Q(4,3) functions of the 4 global input signals.

According to the circuit operation principle, whenever more than 1 local clock element goes logic high the local function Q(4,2) will go logic high, and the respective differentiator will generate a negative pulse to set the trailing flip-flops within the group. Whenever more than 1 local clock element goes logic low, the local function Q(4,3) will go logic low, and the respective differentiator will generate a negative pulse to reset the trailing flip-flops within the group. Whenever more than 1 group output goes logic high (low), the other groups will go logic high (low). No group can go logic high (low) until Δt time units after more than 2 groups have gone low (high).

As shown in Fig. 3-3, suppose groups 1,2,3,4 are not synchronized initially, with group 1 leading group 2, group 2 leading group 3, and group 3 leading group 4. Group 1 goes logic high at t(1). Group 2 goes logic high at t(2). Q(4,2) goes logic high at t(2). So groups 3 and 4 are forced to go logic high at t(2) (after some gate delay). Group 1 intends to go logic low at t(3) ($t(3)-t(1)=\Delta t$.). But since Q(4,3) goes logic high at t(2) and $t(3)-t(2)<\Delta t$, $t(4)-t(2)=\Delta t$, group 4 will not go logic low until t(4). The same thing happens to groups 2,3 and 4. And from here we can see how they are synchronized.

For a system with 3f+1 groups, the functions which make state transition decisions are Q(3f+1,f+1) and Q(3f+1,2f+1). Whenever fewer than f+1 groups fail, those two functions still drive the system properly. Thus it is f-group fault tolerant.

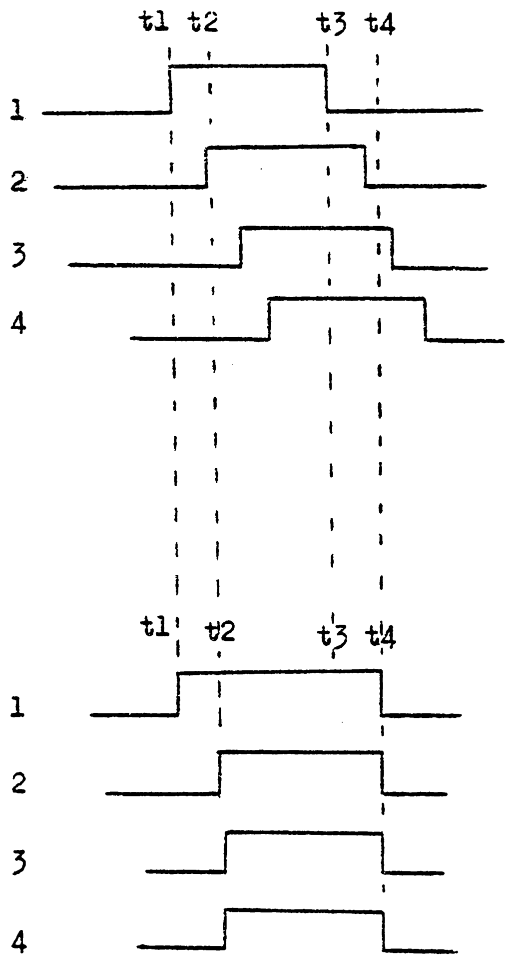


Figure 3-3 Group Output Waveform

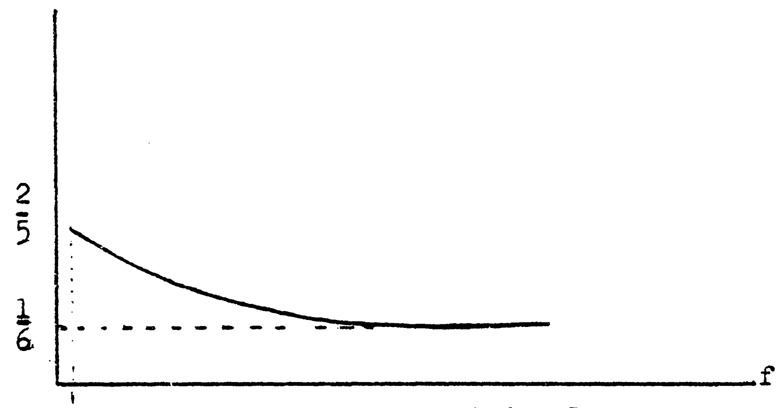
Table 3-1 Comparison between Fletcher's Type and 1st-Order Group

F	No. of Signals	Inputs per element		No. of interconnections	
		Fletcher's Type	1st-order Group	Fletcher's Type	1st order Group
1	16	16	8	256	128
2	49	49	14	2401	686
3	100	100	20	10000	2000
4	169	169	26	28561	4394
5	256	256	32	65536	8192
6	361	361	38	130321	13718
7	484	484	44	234256	21296
8	625	625	50	390625	31250
9	784	784	56	614656	43904
10	961	961	62	923521	59582
11	1156	1156	68	1336336	78608
12	1369	1369	74	1874161	101306
13	1600	1600	80	2560000	128000
14	1849	1849	86	3418801	159014
15	2116	2116	92	4477456	194672
16	2401	2401	98	5764801	235298
17	2704	2704	104	7311616	281216
18	3025	3025	110	9150625	332750
19	3364	3364	116	11316496	390224
20	3721	3721	122	13845841	453962

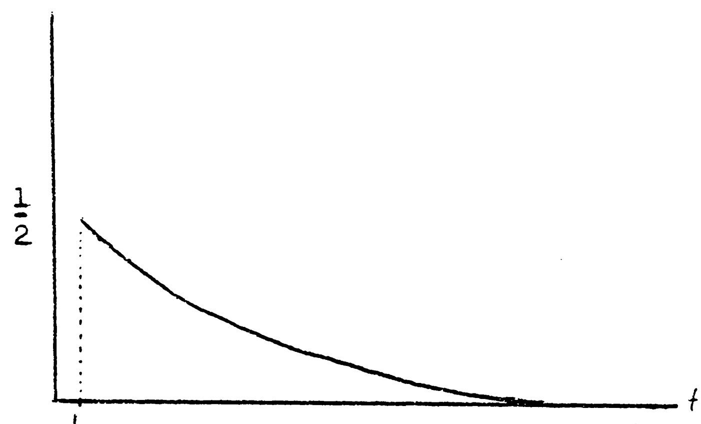
Table 3-1 (Continued)

F	No. of Signals	Fault Tolerant Capability (Element)		
		Fletcher's Type	1st Order Group	
			Worst Case	Best Case
1	16	5	2	7
2	49	16	5	24
3	100	33	9	51
4	169	56	14	88
5	256	85	20	135
6	361	120	27	192
7	484	161	35	259
8	625	208	44	336
9	784	261	54	423
10	961	320	65	520
11	1156	385	77	627
12	1369	456	90	744
13	1600	533	104	871
14	1849	616	119	1008
15	2116	705	135	1155
16	2401	800	152	1312
17	2704	901	170	1479
18	3025	1008	189	1656
19	3364	1121	209	1843
20	3721	1240	230	2040

Table 3-1 (Cont.)



Fault Tolerance in 1st Order Group
Fault Tolerance in Fletcher's Type



No. of Interconnections in 1st Order Group
No. of Interconnections in Fletcher's Type

For the patent connection, a system with 3f+1 elements will have (3f+1)*(3f+1) interconnections (each element has 3f+1 input lines). For the current connections, there are 3f+1 groups with 3f+1 elements in each group. There are (3f+1)*(3f+1) elements in total. Each element has 2*(3f+1) inputs. The total number of interconnections is 2*(3f+1)**3. In order to generate (3f+1)**2 clock signals, (3f+1)**4 interconnection are needed for the patent and 2*(3f+1)**3 for the current connection pattern; the difference being (3f-1)(3f+1)**3. A comparison is shown in Table. 3-1.

3.2 2nd-order clock

In the former section, the clocking system has 3f+1 groups. Each group has 3f+1 clock elements. This idea is now extended by building a system with (3f+1)**2 groups. Each group having 3f+1 subgroups. And each subgroup having 3f+1 clock elements. In other words, a system is built with (3f+1)**2 groups which itself is a clocking system described in the former section. There is some slight addition in the element structure. It consists of a local synchronizer, a group synchronizer and a global synchronizer. The local synchronizer has local quorum logic generating local functions Q(3f+1,f+1) and Q(3f+1,2f+1) of the 3f+1 local input signals coming from all clock elements within its own subgroup. A 'group synchronizer' has a 'group quorum logic' which will generate 'group functions' Q(3f+1,f+1) and Q(3f+1,2f+1) where the 3f+1 input signals each come from a different one of the 3f+1 subgroups of the regional group. A 'global synchronizer' has a 'global quorum logic' which generates 'global functions' Q(3f+1,f+1) and Q(3f+1,2f+1) of 3f+1 input signals each coming from a different one of the 3f+1 groups. Every element is connected to every other element within the subgroup and every subgroup is connected to every subgroup within the group in the same way as described in the former section. Also, every element knows when other elements will go logic high or go low within its own subgroup, every subgroup knows when other subgroup will go logic high or go logic low within its own group, and every group knows when other groups will go logic high or go logic low.

With the element circuit structure and circuit operation described above, the system will operate as follows: When more than f elements go high (low) in a subgroup, the other elements within the subgroup will go high.

When more than f subgroups in a group go logic high (low), the other subgroups within the group will go logic high (low). When more than n (at most f*(3f+1)+f+1, at least

$(f+1)*(f+1))$ groups go logic high (low), the other groups will go logic high (low), No groups can go logic high (low) until Δt time units after more than n (at least $(2f+1)(2f+1)$, at most $2f*(3f+1)+f+1)$ groups go logic low (high). With the first 3 rules, the trailing elements can be speeded up. With the last rule, the leading elements can be slowed down. Finally the system will be synchronized.

There are $(3f+1)**4$ elements in this system. Each element has $3*(3f+1)$ input signals. The total number of interconnections is $3*(3f+1)**5$. With the patent's connection scheme, in order to generate $(3f+1)**4$ clock signals, each element will have $(3f+1)**4$ input signals. The total number of interconnections will be $(3f+1)**8$. The difference is $((3f+1)**3-3)*(3f+1)**5$. Table.3-2 is a comparison of their difference.

Let f equal 1. Then there are $(3f+1)**2=16$ groups in this system. Each group has $3f+1=4$ subgroups. Each subgroup has 4 elements. Each element has 12 inputs, including 4 local inputs, 4 subgroup inputs and 4 group inputs.

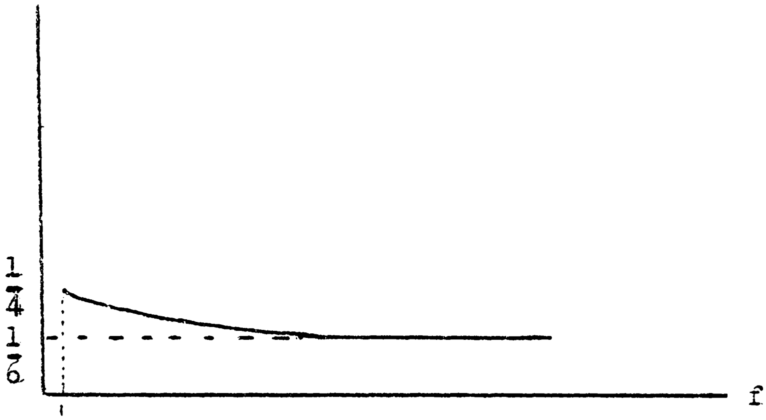
Table 3-2 Comparison between Fletcher's Type and 2nd-Order Group

F	No. of Signals	Inputs per element		No. of interconnections	
		Fletcher's Type	2nd Order Group	Fletcher's Type	2nd-order Group
1	256	256	12	65536	3072
2	2401	2401	21	5764801	50421
3	10000	10000	30	100000000	300000
4	28561	28561	39	815730721	1113879
5	65536	65536	48	4294967296	3145728
6	130321	130321	57	16983563041	7428297
7	234256	234256	66	54875873536	15461896
8	390625	390625	75	152587890625	29296875
9	614656	614656	84	377801998336	51631104
10	923521	923521	93	852891037441	85887453
11	1336336	1336336	102	1785793904896	136306272
12	1874161	1874161	111	3512479453921	208031871
13	2560000	2560000	120	6553600000000	307200000
14	3418801	3418801	129	11688200277601	441025329
15	4477456	4477456	138	20047612231936	617888928
16	5764801	5764801	147	33232930569601	847425747
17	7311616	7311616	156	53459728531456	1140612096
18	9150625	9150625	165	83733937890625	1509853125
19	11316496	11316496	174	128063081718016	1969070304
20	13845841	13845841	183	191707312997281	2533788903

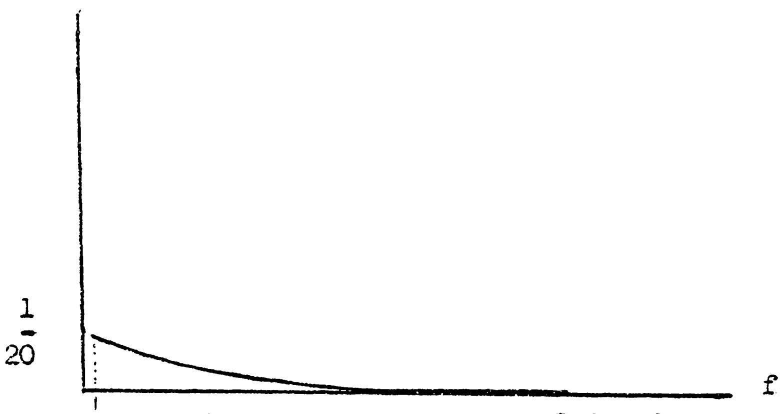
Table 3-2 (Continued)

F	No. of Signals	Fault Tolerant Capability (Element)		
		Fletcher's Type	2nd Order Group	
			Worst Case	Best Case
1	256	85	20	135
2	2401	800	152	1312
3	10000	3333	594	5511
4	28561	9520	1652	15792
5	65536	21845	3740	36295
6	130321	43440	7380	72240
7	234256	78085	13202	129927
8	390625	130208	21944	216736
9	614656	204885	34452	341127
10	923521	307840	51680	512640
11	1336336	445445	74690	741995
12	1874161	624720	104652	1040592
13	2560000	853333	142844	1421511
14	3418801	1139600	190652	1898512
15	4477456	1492485	249570	2486535
16	5764801	1921600	321200	3201600
17	7311616	2437205	407252	4060807
18	9150625	3050208	509544	5082336
19	11316496	3772165	630002	6285447
20	13845841	4615280	770660	7690480

Table 3-2 (Cont.)



Fault Tolerance in 2nd Order Group
Fault Tolerance in Fletcher's Type



No. of Interconnections in 2nd Order Group
No. of Interconnections in Fletcher's Type

The local ,subgroup and group quorum logic of each element will produce local functions Q(4,2), Q(4,3), group functions Q(4,2), Q(4,3) and global functions Q(4,2) and Q(4,3). These functions are used to determine when to set the element and when to reset the element.

In accordance with the operation rule described above, whenever more than one element goes high (low), the other elements in the same subgroup within the same group will go high (low). Whenever more than 1 group goes high (low), the other groups will go high (low). Δt time units after more than 2 groups have gone high (low), all groups will begin to go low (high).

This system can generate 256 synchronized clock signals, and is 1-group fault tolerant. There are 256 elements. Each element has 12 inputs. The total number of interconnections is 12*256=3072. With the patent's interconnection pattern, in order to generate 256 clock signals, we need 256 elements, each of which has 256 inputs. The total number of interconnections is 256*256=65536. A large difference in circuit complexity is seen by looking at the number of interconnections. Also,the quorum logic functions are only functions of 4 variables, compared to quorum logic with 256 inputs for the patent.

3.3 Generalization

From the previous sections, we know that we can build a large clocking system systematically, First, a 3f+1-element 0th order group is built. then a 1st order group with (3f+1) 0th order groups, then a 2nd order group with (3f+1)(3f+1) 1st order groups, then a 3rd order group with (3f+1)**(2**2) 2nd order groups... Until finally an mth order group with (3f+1)**(2**(m-1)) m-1st order groups. A 0th order group has 3f+1 clock elements. A 1st order group has (3f+1)**(2) elements. An mth order group has (3f+1)**(2**m) elements. Fletcher's patent is a system consisting of one 0th order group. The system we described in the first section is a 1st order group. The system we described in the last section is a 2nd order group. For a system with 0th order group, each element has a 0th order quorum logic with 3f+1 inputs. For a system with a 1st order group, each element has a 0th order quorum logic and a 1st order quorum logic. For a system with an mth order group, each element has a 0th order, a 1st order..., and an mth order quorum logic. Within each 0th order group, when more than f elements go high (low), the others will go high (low). Within each 1st order group, when more than f 0th order groups go high (low), the others will go high (low). Within each 2nd order group, when more than n (at most f*(3f+1)*(f+1), at least (f+1)**2)

1st order groups go high (low), the others will go high (low).... Within each mth order group, when more than n m-1st order groups go high (low),(n is a function of m) the others will go high (low).

There are (3f+1)**(2**m) elements in an mth order system. Each element has (m+1)(3f+1) inputs. There are m+1 identical quorum functions each with 3f+1. Thus, the quorum functions are only functions of 3f+1 variables. The total number of interconnections is (m+1)(3f+1)(3f+1)**(2**m). This system is f-(m-1)-level group fault tolerant. With the patent's interconnections pattern, in order to generate (3f+1)**(2**m) inputs, the total number of interconnections is (3f+1)**(2**m+2**m). Each element has a quorum function with (3f+1)**(2**m) inputs. Thus, a huge improvement in cell complexity results, a huge improvement in interconnection complexity, and a huge improvement in fault-tolerance.

Chapter 4 Iterative Type Connection

4.1 Cascaded Type

Fletcher's patent is actually an asynchronous circuit. One common technique for dealing with asynchronous circuits is to develop an iterative array combinational circuit model. With this in mind, we come up with the idea of an iterative array of clock elements which has the possibility of greatly reducing the number of interconnections. Obviously, these elements must be synchronized. Their fault tolerant capability is also important. Consider now two simple cases, as shown in Fig. 4-1. The upper circuit contains 2 boxes A and B. The lower circuit contains box C. Each box contains 3f+1 clock elements. The circuit structure of each clock elements is the same as described in Fig. 2-1. Those three boxes A,B, and C are completely the same. The lower connection is the same as the connection in the patent. We know that it can generate 3f+1 synchronized clock signals with f-fault tolerant capability. For the left circuit, the 3f+1 output signals of box A are connected to the inputs of box B, the 3f+1 output signals of box B are connected to the inputs of box A.

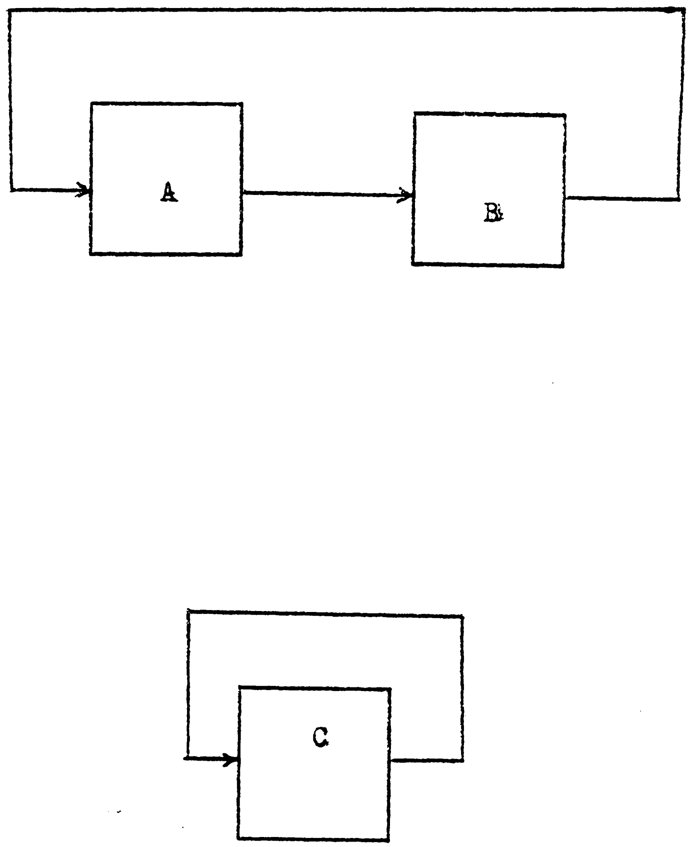


Figure 4-1 Two Stage Case

Let us analyze the two-box system. Whenever more than f clock elements of box A go high, the function Q(3f+1,f+1) of all 3f+1 clock elements of box B will go high, and all clock elements of box B will go high; then the function Q(3f+1,f+1) of all 3f+1 clock elements of box A will get a 0 to 1 transition, which will set all other clock elements to the 1 state.

Whenever more than f clock elements of box A go low, the function Q(3f+1,2f+1) of all 3f+1 clock elements of box B will go low, and all clock elements of box B will go low. Then the function Q(3f+1,2f+1) of all 3f+1 clock elements of box A will get a 0 to 1 transition which will reset all other clock elements in box A to the 0 state.

Therefore, whenever more than f clock elements of box A go high, the other clock elements of box A will follow. Similarly, we can say that whenever more than f clock elements of box B go high, the other clock elements of box B will go high. Up to now we have demonstrated that all clock elements of boxes A and B are synchronized to within gate delays in the individual circuits.

No clock elements of box A can go low until Δt time units after more than 2f clock elements of box B go high. No clock elements of box B can go low until Δt time units after more than 2f clock elements of box A go high.

Besides, all clock elements of box A and B are synchronized. In conclusion, boxes A and B can each generate 3f+1 synchronized clock signals, whose frequency is determined by Δt.

Whenever the number of failed clock elements of box A is less than f+1 and the number of failed elements of box B is less than f+1, the functions Q(3f+1, f+1) and Q(3f+1,2f+1) still have correct outputs. Therefore boxes A and B are f-fault tolerant. Together boxes A and B have 6f+2 clock elements. This system can generate 6f+2 synchronized signals. The number of interconnections in this system is 2*(3f+1)*(3f+1). But with the patent connection, in order to generate 6f+2 clock signals, the number of interconnections is (6f+2)*(6f+2). The difference is 2*(3f+1)*(3f+1). At the same time, they have the same fault-tolerant capability.

To illustrate this,let f be 1. Then both box A and box B have 4 clock elements. The output signal from each element of box A is connected to the input of each element of box B. The output signal of each element of box B is also connected to the input of each element of box A. At the time when 2 clock elements of box A go high, Q(4,2) of each element of box B will go high. Thus box B's elements which stay at 0 will make a 0 to 1 transition, which will cause the other elements of box A to go high immediately. At the time when 2 clock elements of box B go high, Q(4,2) of each element of box A will go high. Then box A's elements which stay at 0 will make a 0 to 1 transition, which will cause the other elements of box B to go high immediately. Therefore, whenever more than 1 clock element of box A goes high (low), all other elements of boxes A and B will also go high (low) immediately. For the present example, no clock element can go low until Δt time units after Q(4,3) goes high. No clock element can go high until Δ t time units after Q(4,2) goes low. Therefore, no element of box B can go low (high) until Δt time later after more than 2 elements of box A go high. No element of box A can go low (high) until Δt time units after more than 2 elements of box B go high (low). From the above explanation, we can say that this system can generate 8 synchronized clock signals whose common frequency is determined by Δt. The number of interconnections is 2*4*4, which is equal to 32. Each element only has 4 inputs. However, if we use the patent's connection type, the number of interconnections will be 64. Each element will need 8 inputs.

As we consider the general case, we assume that there is an array of boxes A(1), A(2)...A(n). Each box consists of 3f+1 clock elements. The 3f+1 output signals of box A(1) are connected to the input of box A(2). 3f+1 output signals of box A(2) are connected to the input port of all elements of box A(3)...; finally, the 3f+1 output signals of box A(n) are connected to the input port of all elements of box A(1) (Fig. 4-2).

At first, let us look into box A(1). Whenever more than f clock elements of box A(1) go high, all clock elements of box A(2) will go high. As a result, all clock elements of box A(3) will go high..., all clock elements of box A(n) will go high, then all other clock elements of box A(1) will go high. (Time delay grows around the loop.) Similarly, whenever more than f clock elements of box A(i) go high, all clock elements of box A(i+1) will go high.... All clock elements of box A(i-1) will go high, then all other clock elements of box A(i) will go high.

On the other hand, except for the above situation, no clock elements of box A(1) can go high until Δt time units after more than 2f clock elements of box A(n) go low. No clock elements of box A(2) can go high until Δt time units after more than 2f clock elements of box A(1) go low. No clock elements of box A(i) can go high until Δt time units after more than 2f clock elements of box A(i-1) go low.

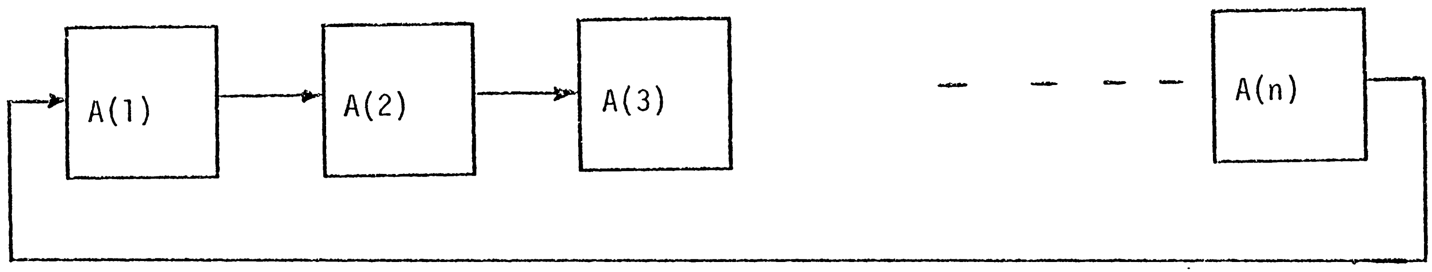


Figure 4-2 Cascaded Type Connection

Table 4-1 Comparison between Fletcher's Type and Cascade Type (Letting n be 10)

F	No. of Signals	Inputs per elements		No. of interconnection	
		Fletcher's Type	Cascade Type	Fletcher's Type	Cascade Type
1	40	40	4	1600	160
2	70	70	7	4900	490
3	100	100	10	10000	1000
4	130	130	13	16900	1690
5	160	160	16	25600	2560
6	190	190	19	36100	3610
7	220	220	22	48400	4840
8	250	250	25	62500	6250
9	280	280	28	78400	7840
10	310	310	31	96100	9610
11	340	340	34	115600	11560
12	370	370	37	136900	13690
13	400	400	40	160000	16000
14	430	430	43	184900	18490
15	460	460	46	211600	21160
16	490	490	49	240100	24010
17	520	520	52	270400	27040
18	550	550	55	302500	30250
19	580	580	58	336400	33640
20	610	610	61	372100	37210

From the above points, we can conclude that either when more than f clock elements of box A(i) go high, all other clock elements of box A(i) go high, or Δt time units after more than 2f clock elements go low, all other clock elements of box A(i) will go high. Meanwhile all clock elements of all other boxes will go high; consequently all clock elements of all boxes are synchronized. They can generate n*(3f+1) synchronized clock signals whose common frequency is determined by Δt. This system comprises n*(3f+1) clock elements. The number of interconnections within the system is n*(3f+1)*(3f+1). It can generate n*(3f+1) synchronized clock signals. But with the patent connection, in order to generate n*(3f+1) clock signals we need n*n*(3f+1)(3f+1) inerconnctions among those clock elements. The difference is (n*n-n)(3f+1)(3f+1). A comparison is shown in Table. 4-1. Even if f clock elements of each box fail, the system still functions correctly.It still generates n*(2f+1) syrchronized clock signals. Therefore, each box is f-fault tolerant.

What we should notice is that there is actually some amount of time delay between the time at which clock elements of box A(i) make state transitions and the time at which clock elements of box A(i+1) make state transitions. Suppose it is δt; then, the maximum possible time delay would be $n\delta t$. If the maximum allowable time delay among different clock signals in a computer system is max t, then the maximum number of allowable stages is max $t/\delta t$.

4-2 Scattering Type

If we use the connection pattern described in the last section, then the number of interconnections will be greatly reduced. However, if more than f clock elements of a single box fail, the whole system will fail. To improve the fault-tolerant property, consider the following interconnection pattern.

Suppose we have an array of boxes A(1), A(2)...A(n). Box A(i) has 3f+1 output signals. Instead of connecting all of these to box A(i+1), we will connect each one to boxes A(i+1), A(i+2),...A(i+3f+1), where the subscripts are computed modulo n. For example, box A(i) has 3f+1 signals. It will supply one signal to each one of the boxes A(2), A(3)...A(3f+2). Box A(2) will supply a signal to each one of boxes A(3), A(4)...A(3f+3). Box A(n) will supply signals to each one of boxes A(1), A(2)..A(3f+1). In other words, box A(i) will get one signal from A(i-3f-1), one from A(i-3f-2)...one from A(i-1), with subscripts modulo n. Each box will get its 3f+1 input signals from the 3f+1 preceding boxes. (Fig. 4-3)

When more than f boxes among boxes A(1), A(2)..A(3f+1) go high, than f input signals to box A(3f+1) will go high. The function Q(3f+1,f+1) of all clock elements of box A(3f+2) will go high, then all clock elements of box A(3f+2) will go high. As the clock elements of box (3f+2) go high, the number of boxes which go high among boxes A(2), A(3)..., A(3f+2) is also greater than f. As a result of this event, the function Q(3f+1,f+1) of all clock elements of box A(3f+3) will go high; therefore, these elements will go high. Similarly, all clock elements of boxes A(3f+4)...A(n)...A(1)...A(3f+1) will go high. Therefore, whenever, more than f boxes among boxes A(1), A(2)...A(3f+1) go high, all other boxes will go high. With the same criteria, whenever more than f boxes among A(2), A(3)... A(3f+2) go high, all other boxes will go high. Undoubtedly, whenever more than f boxes among A(n), A(1),...A(3f) go high , all other boxes will go high.

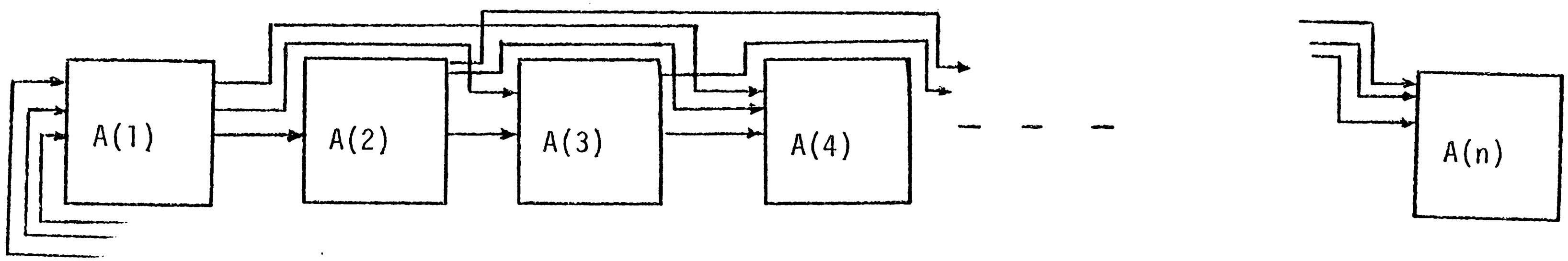


Figure 4-3 Scattering Type Connection

Table 4-2 Comparison between Fletcher's Type and Scattering Type

(Letting n be 10)

F	No. of Signals	Inputs per element		No. of interconnections	
		Fletcher's Type	Scattering Type	Fletcher's Type	Scattering Type
1	40	40	8	1600	320
2	70	70	14	4900	980
3	100	100	20	10000	2000
4	130	130	26	16900	3380
5	160	160	32	25600	5120
6	190	190	38	36100	7220
7	220	220	44	48400	9680
8	250	250	50	62500	12500
9	280	280	56	78400	15680
10	310	310	62	96100	19220
11	340	340	68	115600	23120
12	370	370	74	136900	27380
13	400	400	80	160000	32000
14	430	430	86	184900	36980
15	460	460	92	211600	42320
16	490	490	98	240100	48020
17	520	520	104	270400	54080
18	550	550	110	302500	60500
19	580	580	116	336400	67280
20	610	610	122	372100	74420

Except for the above situation, no clock element of box A(3f+2) can go high until t time later after more than 2f boxes among box A(1), A(2),... A(3f+1) go low. In general, except for the above situation, no clock elements of box A(i) can go high until t time units after more than 2f boxes among box A(i-3f-1), A(i-3f)...A(i-1) go low. When more than 2f boxes of its 3f+1 preceeding boxes go low, the function Q(3f+1,2f+1) of all clock elements of box A(i) will go low. t time later, a negative pulse will be generated to set the clock elements of box A(i).

From the above discription, we can see that this system can generate n*(3f+1) synchronized clock signals whose frequency is determined by Δt. The number of interconnections required in the system is 2n*(3f+1)*(3f+1). A comparison between this type of connection and the patent connection is shown in Table. 4-2. But its fault-tolerant capability is much better. The failure of all clock elements of a single box will not cause the whole system to fail. In fact, the system will not fail until more than f boxes among 3f+1 consecutive boxes fail.

4.3 2-dimensional Type

Another way to improve reliability is to use a twodimensional structure, as shown in Fig.4-4. There are 3f+1 boxes in each column and n boxes in each row. Each box contains 3f+1 clock elements. In the system described in the first section, each box supplied 3f+1 clock signals to its succeeding box. Now each box will supply one clock signal to each of the 3f+1 boxes in the succeeding column. Each box will have 3f+1 input signals coming from the 3f+1 boxes in the preceding column. The very last column of boxes will be wrapped around to the first column of boxes.

For any column j, if more than f boxes in this column go high, then the function Q(3f+1,f+1) of the clock elements of all boxes of column j+1 will go high, these boxes will go high, the boxes of column j+2 will go high..., and, finally all other boxes of column j will go high. Therefore, if more than f boxes in any column go high, all boxes in all columns will go high.

When more than 2f boxes of column j go high, the function Q(3f+1,2f+1) of the clock elements of all boxes j+1 will go high.

Δt time later, these clock elements will be reset to the 0 state, and then all clock elements will be reset to the 0 state. When more than 2f boxes of column j go low, the function q(3f+1,f+1) of the clock elements of all boxes of column j+1 will go low.

Δt time later, these clock elements will be set to the 1 state and all clock elements of the boxes of all columns will be set to the 1 state.

As mentioned before, each row has n boxes, each column has 3f+1 boxes, and each box contains 3f+1 clock elements. This system can generate n*(3f+1)(3f+1) synchronized clock signals whose frequency is determined by Δt.

When fewer than f boxes of column j fail, the clock elements of all boxes of column j+1 still get more than 2f input signals. Their quorum functions Q(3f+1,2f+1) and Q(3f+1,f+1) still function correctly. These clock elements can still be set and reset at the proper time. So can the successive clock elements. Therefore, whenever the number of failed boxes in each column is smaller than f, the rest of the system still generates synchronized clock signals. Its fault tolerant capability is much better than that of the connection pattern described in the first section of the chapter. Each element has 3*(3f+1) inputs. The number of interconnections among these boxes is n*(3f+1)(3f+1). The total number of interconnection of the whole system is 3*(3f+1)(3f+1)(3f+1)*n. There is still a big difference between the number of interconnection of this connection pattern and that of the patent. (Table. 4-3)

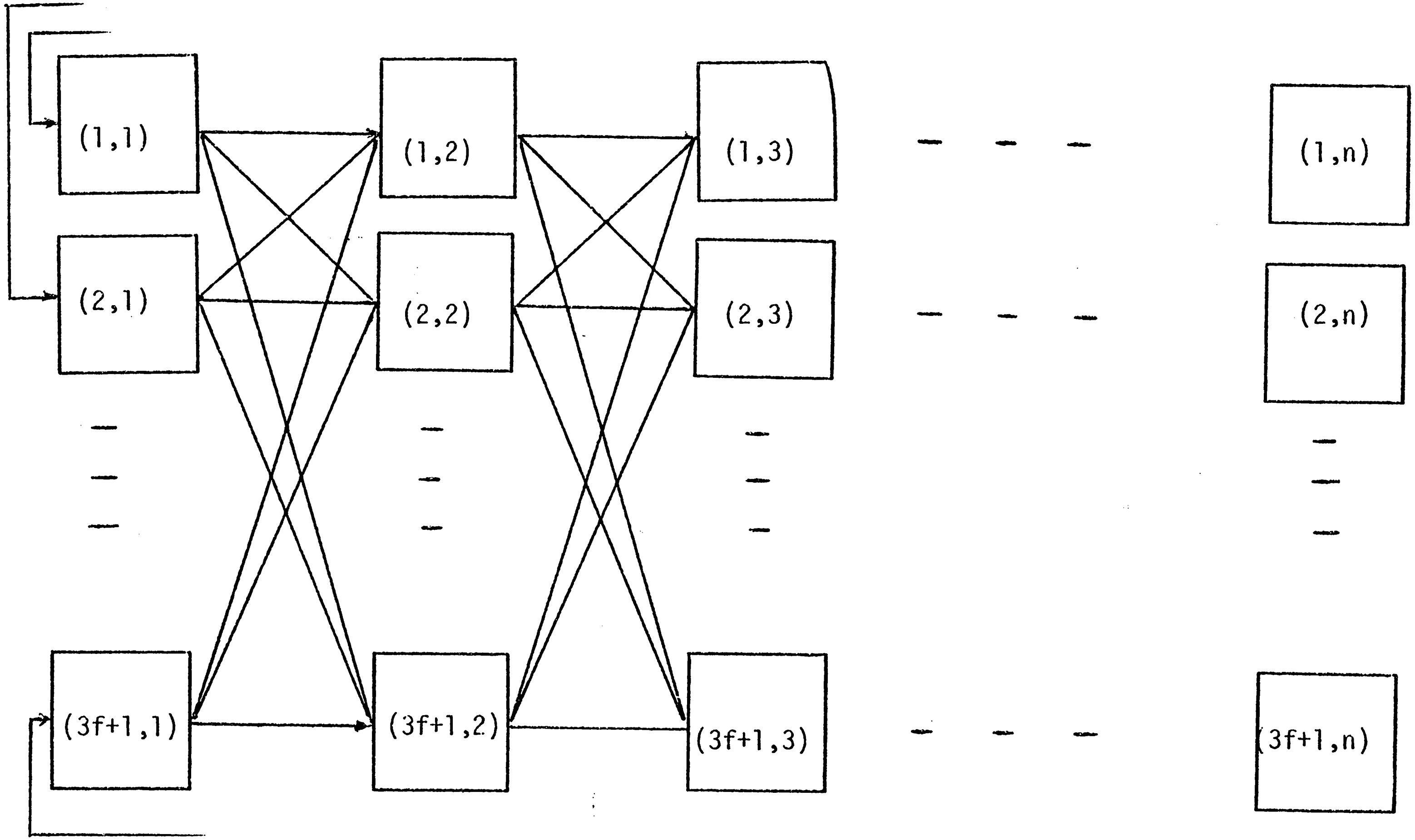


Figure 4-4 2-dimensional Type Connection

Table 4-3 Comparison between Fletcher's Type and 2-Dimensional Type (Letting n be 10)

F	No. of Signals	Inputs per element		No. of interconnections	
		Fletcher's Type	2-dimensional Type	Fletcher's Type	2-dimensional Type
1	160	160	12	25600	1920
2	490	490	21	240100	10290
3	1000	1000	30	1000000	30000
4	1690	1690	39	2856100	65910
5	2560	2560	48	6553600	122880
6	3610	3610	57	13032100	205770
7	4840	4840	66	23425600	319440
8	6250	6250	75	39062500	468750
9	7840	7840	84	61465600	658560
10	9610	9610	93	92352100	893730
11	11560	11560	102	133633600	1179120
12	13690	13690	111	187416100	1519590
13	16000	16000	120	256000000	1920000
14	18490	18490	129	341880100	2385210
15	21160	21160	138	447745600	2920080
16	24010	24010	147	576480100	3529470
17	27040	27040	156	731161600	4218240
18	31250	30250	165	915062500	4991250
19	33640	33640	174	1131649600	5853360
20	37210	37210	183	1384584100	6809430

Let f be 1, n be 4. Then we have 4 boxes in each row, 4 boxes in each column, and each box has 4 elements. We now name these boxes (i,j), where i,j=1,2,3,4. For example, (1,2) represents box located at row 1, column 2. At the time when 2 boxes of column 1 go high, say (1,1) and (2,1), then Q(4,2) of all elements of boxes (1,2) and (2,1), then Q(4,2) of all elements of boxes (1,2), (2,2), (3,2) (4,2) will go high, these elements will go high. So do the elements of the boxes of columns 3 and 4. Thus, Q(4,2) of the elements of boxes (3,1), (4,1) will go high and these elements will go high. The same thing happens to the other columns. Thus whenever more than 1 box of column j goes high (low), all other boxes of this column and all other columns will go high (low) immediately. (Actually there is some delay whose maximum value is equal to 4 times the amount of delay between the leading flip-flop and the trailing flip-flop in the patent.) In other words, they are always synchronized, going high at the same time and going low at the same time.

In this system, no clock elements of column 2 can go high (low) until

Δt time units after more than 2 boxes of column 1 go low (high). No clock elements of column 3 can go high (low) until Δt time units after more than 2 boxes of column 2 go low (high). The same criteria applys to column 4. This system can therefore generate 64 synchronized clock signals whose common frequency is determined by Δt. The required number of interconnections is 64*12=768. Each element has 12 inputs. If we use the patent's connection pattern, then the required number of interconnections will be 64*64=4096. Each element has 64 inputs. It is not too difficult to imagine how hard it is to build a function of 64 inputs. Therefore a significant improvement in circuit complexity has been achieved.

Chapter 5 Conclusion

It has been shown that Fletcher's clock patent can be modified to generate the same number of synchronized clock signals with greatly reduced number of interconnections. Two approaches have been proposed to achieve the goal.

According to the definition given in chapter 3, Fletcher's clock patent is a 0th-order group. An mth-order group is a clock system consisting of $(3f+1)**(2**(m-1))$ $(m-1)$th-order groups. It has $(3f+1)**(2**m)$ clock elements. The required number of interconnections among different elements is $(m+1)(3f+1)**(2**m+1)$.

Another approach is to use an iterative type connection. An n-stage cascaded clock circuit can generate $n*(3f+1)$ synchronized clock signals with $n*(3f+1)*(3f+1)$ interconnections. Each stage has f-element fault tolerant capability. The maximum value of n is the allowable delay between different clock signals divided by the amount of delay through each individual clock element. A similar type of connection is called scattering type. It can generate $n*(3f+1)$ clock signals with $2n*(3f+1)(3f+1)$ interconnections. The failure of a whole stage will not cause the system to fail. Actually the system will not fail until the number of failed stages in 3f+1 consecutive stages exceeds f. Another type connection whose number of groups grows in two directionsis called 2-dimensional type. This circuit can generate n*(3f+1)(3f+1) clock signals. The required number of interconnections is 3n*(3f+1)**3. With this type of connection,each stage is f-group fault tolerant.

Bibliography

[1] C.Gertson,'Missing-pulse detector for narrow pulses' EEE,vol. 12,pp. 72-74,Aug. 1964.

[2] H.S.Reichard,'Missing pulse detector' ,EEE,vol.10, p.35, June 1962.

[3] F.F.Sellers, et al., 'Error Detecting Logic for Digital Computers', pp. 244-246.

[4] L.J.Koczela,'A three failure tolerant computer system' in Dig. 1971 Intl. Symp. Fault-Tolerant Computing, Pasadena,Ca. pp. 101-104.

[5] H.Y.Chang, et al., 'Maintenance techniques of a microprogrammable self-checking control complex of an electronic switching system' IEEE Trans.Comp., vol.C-22,pp. 501-512, May 1973.

[6] A.M.Usas , 'The detection of errors in periodic signals' ,IEEE Computer Society R-75-10.

[7] J.C.Fletcher,W.M. Daly and J.F.Mckenna,'Fault Tolerant Clock Apparatus Utilizing a Controlled Minority of Clock Elements', U.S.patent number 3,900,741.

[8] M.A.Breuer and A.D.Friedman , 'Diagnosis and reliable design of digital systems', Computer Science Press Inc..

[9] B.A.Prasad and F.G.Gray , 'Research Initiation: A Theoretical Investigation of Diagnosable Logic Systems', Final Report,NSF,Oct. 1973.

The vita has been removed from the scanned document

FAULT TOLERANT CLOCKING SYSTEM

By

Tzu-I Jonathan Fan

(ABSTRACT)

The distributions of synchronized clock signals to all elements of a computing system is very important. Fletcher's clock patent is a good solution to this problem. The difficulty with implementation is the tremendous number of interconnections among different clock elements. Two methods are proposed to reduce the number of interconnectons without loss of synchronization and fault-tolerant capability. An mth order clock is a circuit consisting of (3f+1)**(2**(m-1)) (m-1)th order clocks. Fletcher's clock patent is a 0th order clock. Starting with 0th order clocks, an mth order clock circuit can be built systematically. An mth order clock circuit can generate (3f+1)**(2**m) synchronized clock signals with (m+1)(3f+1)**(2**m) interconnections instead of (3f+1)**(2*2**m). Its fault-tolerant capability is also considered. Another type of connection is called Iterative Type, which is further classified into Cascaded Type, Scattering Type and 2-dimensional Type. Each type has its own characteristics.