AN ABSTRACT OF THE THESIS OF

Hiro Moriyasu for the Master of Science
(Name) (Degree)

in Electrical and Electronics Engineering presented on May 1, 1969
(Major) (Date)

Title: High Speed Adaptive Logic Circuit

for Adaptive Signal Processing

Redacted for privacy

Abstract approved:
Louis N. Stone

A high speed adaptive signal processing concept and several high speed adaptive logic circuit elements are presented.

The adaptive signal processing system operates in a "goal" oriented mode; the system tries to optimize its characteristics to achieve a given goal in spite of unforeseen variations in the inputs, the system, or lack of detailed bits of instruction.

The adaptive logic circuits presented can perform several Boolean logical operations, such as AND, OR, NAND, NOR, memory, majority as well as arithmetic operations, without changing their circuit configurations. Possible adaptive circuits utilizing tunnel diodes and transistors and higher order adaptive logic circuits are presented. As examples of high speed operations, a 16-bit shift register, a read-only memory, and an adaptive delay line are presented.

What has been presented can offer possibilities for high speed adaptive data processing.

High Speed Adaptive Logic Circuit for Adaptive Signal Processing

by

Hiro Moriyasu

A THESIS

submitted to

OREGON STATE UNIVERSITY

in partial fulfillment of
the requirements for the
degree of

Master of Science

June 1969

APPROVED

Redacted for privacy

Professor of Electrical and Electronics Engineering
in charge of major

Redacted for privacy

Head of Department of Electrical and Electronics Engineering

Redacted for privacy

Dean of Graduate School

Date thesis is presented May 1, 1969

Typed by Pearl Engert for Hiro Moriyasu

ACKNOWLEDGMENTS

The writer is indebted to Professor Louis N. Stone and to the Department of Electrical and Electronics Engineering at the School of Engineering, Oregon State University for the continued guidance and support with this thesis.

Acknowledgment is also extended to Bill Walker and William Velsink of the Engineering Division of Tektronix, Inc. for encouragement and guidance, members of Advanced Circuits and Techniques Development of Tektronix, Inc. for their constructive discussions and aids, and Tektronix, Inc. for the use of printing facilities.

Acknowledgment and appreciation is particularly extended to Pearl Engert for typing the thesis.

TABLE OF CONTENTS

LIST OF FIGURES

Figure Page

HIGH SPEED ADAPTIVE LOGIC CIRCUIT
FOR ADAPTIVE SIGNAL PROCESSING

I. INTRODUCTION

In the past few years, there has been a rapid advancement in digital data processing technology. Such is evidenced by progress in highly sophisticated space guidance control systems, high speed, large scale digital computers for data processing, numerically controlled machinery and digital communication systems. As the tasks become more complex, both hardware and software become highly complex.

To perform complex tasks, very precise step-by-step instructions must be programmed, and the system must operate extremely accurately and reliably. A single bit error in any logic element or programming step cannot be tolerated.

In such a system it is often desirable to provide certain adaptivity to make the system much more flexible for various unforeseen new tasks, in the event of unknown variations in environment, parameters and/or components. For example, the expected input may differ from what was anticipated or include a certain amount of noise, which is quite common if the data is transmitted through microwave, or even telephone lines.

An adaptive signal processor proposed here can operate in a "goal" oriented mode. After the goal and initial instructions are given, it will try to optimize or modify operation to achieve the given goal. In some cases it can give an answer very rapidly with reduced accuracy, or more accurately if sufficient time is allowed.

The system will try to adjust or modify its characteristics to perform different logic operations. The basic adaptive logic performs any one of binary, quinary, or decimal additions; it can perform logic functions such as AND, OR, NAND, NOR or majority, as well as performing as a basic memory storage element.

Development of such flexible adaptive logic can greatly simplify not only the construction of a complex machine but also simplify the software.

There have been several attempts to develop adaptive logic elements. One of these, called "Adalines" (5,17,18), is based on a variable resistance electrode controlled by an electrochemical plating process.

Most of the approaches taken in the past employ a variable analog impedance element (8) as a basic adaptive element which remembers certain previous operations. By use of redundant and majority logic (1,2,9,14) certain adaptive or trainable logic functions are obtained. Although they can perform relatively simple tasks, they are still too slow to be practical in a more general purpose, high efficiency data processor.

The purpose of this study is: (1) to develop a concept for a high speed adaptive signal processor; (2) to investigate some possible adaptive logic elements.

Although what has been proposed here is by no means a total solution, it is hoped that some of the concepts developed here would add to the state of the art of computer technology.

II. HIGH SPEED ADAPTIVE SIGNAL PROCESSING SYSTEM CONCEPT

2.1 Adaptive System

Adaptive signal processing system operates in a "goal" oriented mode; the system tries to optimize its characteristics to achieve a given goal in spite of unavoidable variations in environment. The conventional step-by-step preprogrammed mode can easily halt its operation or give entirely erroneous answer in the event of missing data or instruction.

The proposed high speed adaptive logic signal processor is shown in Figure 2.1.

The adaptive logic system consists of an adaptive input element, adaptive logic circuit, central control, random noise generator, output decision circuit, and reward circuit.

2.2 Adaptive Input Element

The purpose of the adaptive input element is to recognize input signal variations and provide a suitable signal to the following adaptive logic circuit. For example, it can adapt to changes in signal amplitude, incoming information rate, and provide signals for following elements either in series form or parallel form, depending on the nature of the input signal.

Another possible adaptive operation may be added to the input element to perform certain statistical analysis, such as amplitude probability distribution, autocorrelation and crosscorrelation. After making simple statistical analysis, the input element can be adjusted

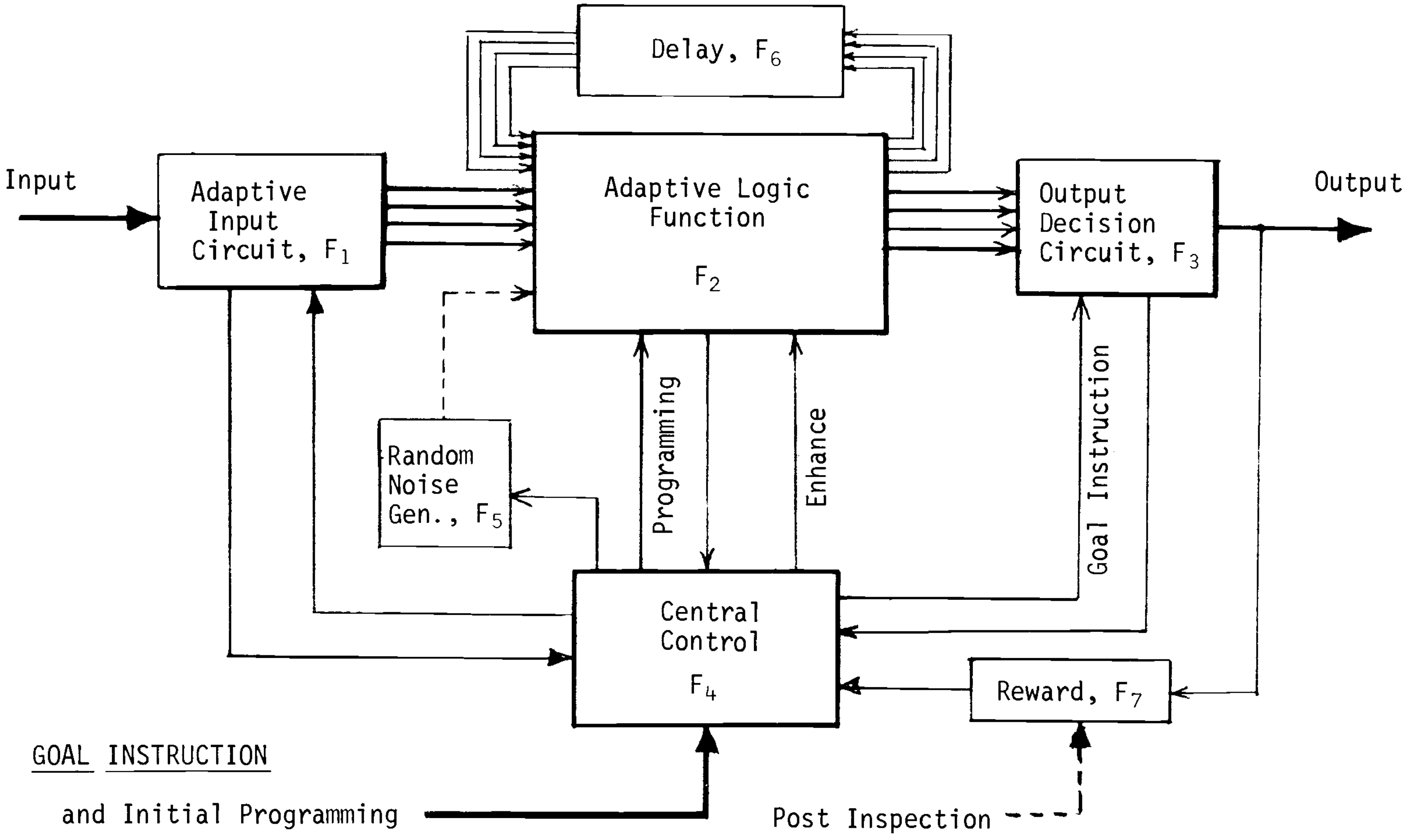


Figure 2.1. Adaptive signal processing system block diagram.

for optimum conditions. This type of input circuit can greatly improve reliability of system operation since it can tolerate variations in the incoming signal such as amplitude, data transmission rate, and noise components.

2.3 Adaptive Logic Functions

The key element of a total system is an adaptive logic circuit, which can perform a variety of logic functions such as AND, OR, NAND, NOR, memory, majority as well as arithmetic operations. It can also accept redundant logic path, and it can perform in a recurrent logic processing mode.

2.4 Output Decision Circuit

The output from the adaptive logic function may consist of many desired and undesired outputs; the output decision circuit sorts out undesirable output by comparing to the initial goal instructions. Only the pertinent outputs are allowed through the output decision circuits. If the output tends to deviate from the goal instruction, it will give information back to central control for modification of programming.

2.5 Central Control

Initial programming of a goal instruction is given to the central control. The purpose of the central control is to study the overall performance of the system and provide the required instructions, such as programming, sequence of logic functions, and modifying storage. It also serves as the controller for input and output.

2.6 Random Noise Generator

The random noise generator may be used for two purposes. It may be used to preset logic and program functions in statistically unbiased conditions. The secondary purpose of the random noise generator is to test noise immunity or stability of the system against random variations in environment or components variations.

2.7 Data Processing by Recirculation of Data

A typical data processing system with multiple input-output cascaded logic blocks is shown in Figure 2.2. To solve certain higher order data processes, a number of logic blocks are cascaded. Each block performs a given function. Since each block has a certain time delay, at time t_1 the data is processed at block one and at time t_2 the data will be processed in block two and likewise at t_n data will arrive at function block n. During one clock period each block performs only one given function. At any given time only one block may be in use and the rest of the blocks are not.

A possible recirculating type adaptive logic function is shown in Figure 2.3. In this recirculating, adaptive logic, the data is recirculated n times to perform n cascaded operations. To perform various functions, the adaptive logic block changes its logic functions for each recirculation of data. Such a method greatly reduces the number of components necessary. This recirculation may be possible with adaptive logic functions. To improve reliability of data processing, various techniques are used such as programming by

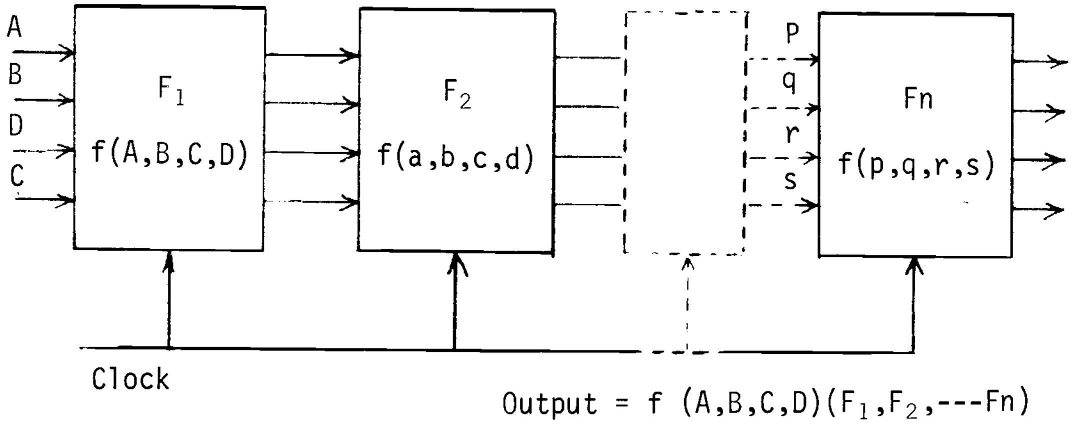


Figure 2.2. Conventional Data Processing.
Multi input-output logic blocks are cascaded for n stages.

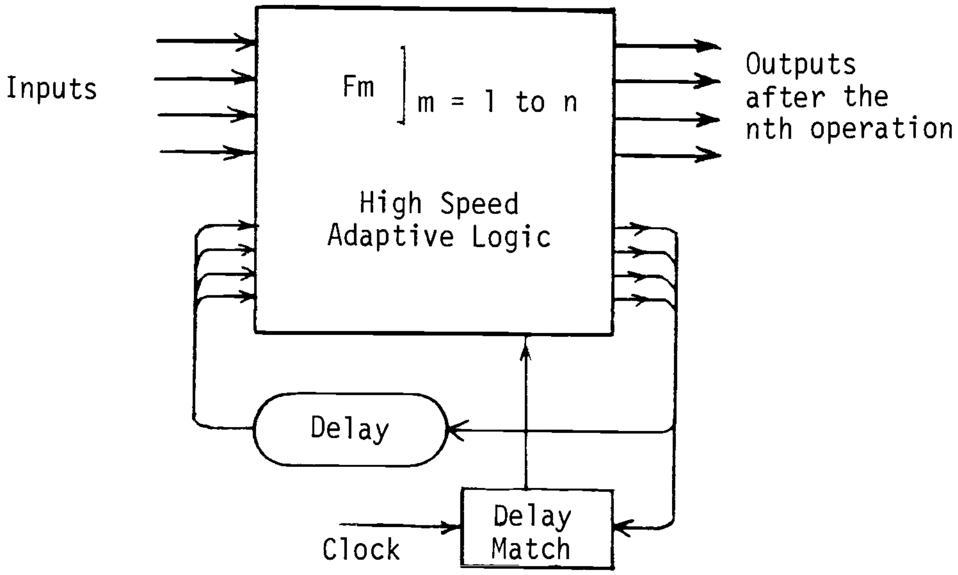


Figure 2.3. Adaptive Recirculating Data Processing.
Adaptive logic block replaces n cascaded conventional logic blocks. The input data is recirculated around the block n times during one clock pulse time. The logic function of the block is varied for each circulation of data.

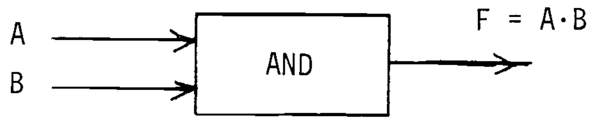


Figure 2.4. Conventional AND logic.

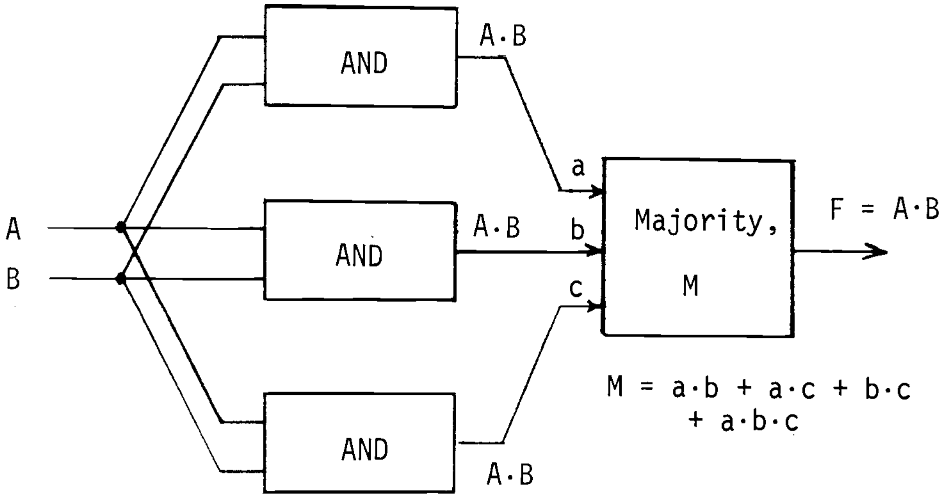


Figure 2.5. Redundant logic circuit allows reliable AND operation in event of component failure.

redundant operation and self checking programming.

2.8 Redundant Logic

In Figure 2.4, nonredundant logic is shown. It will be seen that if there is any failure in input or output logic, the logic becomes nonfunctional.

In Figure 2.5, redundant logic circuit is shown. Providing three redundant paths the reliability is improved in the event of component failure. If any one of the AND circuits fail, the total system still functions as normal. The redundant logic circuit could be incorporated in adaptive logic systems.

2.9 Reward Circuit

After the operation is completed the operator will make a post-inspection; if the results of the output are satisfactory, a reward is given. Thus certain correct operations will be enhanced.

2.10 Adaptability to Output Conditions

In conventional logic elements, the output condition is precisely determined by the logic function and the input. But in some cases it may be necessary to consider the output conditions and adapt accordingly. In Figure 2.6, a logic circuit that adapts to the output conditions is shown.

2.11 Adaptability to Bidirectional Signal Transmission

Conventional unidirectional logic and bidirectional logic circuits are shown in Figure 2.7. If only unidirectional logic is used in a data transmission flow, a failure at any transmission point will

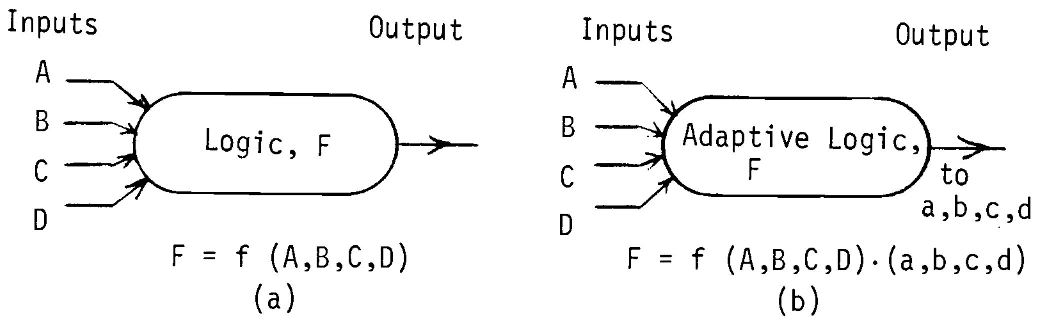


Figure 2.6. Adaptability to fan-out, output conditions
a) Conventional logic
b) Adaptive logic readjusts its characteristics to meet a change in the load condition.

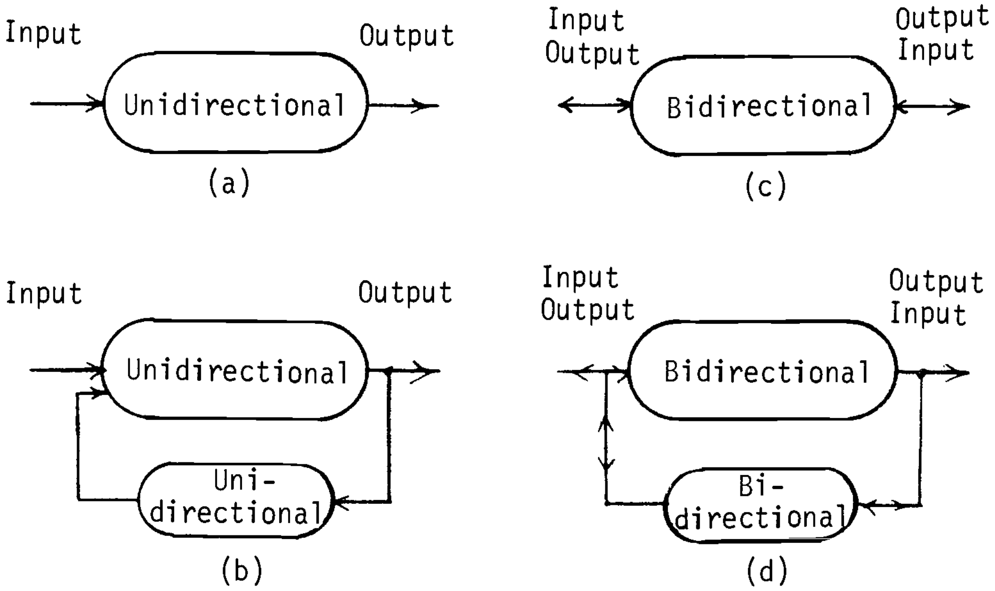


Figure 2.7. Adaptability to bidirectional signal transmission
a,b) Conventional unidirectional logic
c,d) Adaptive bidirectional logic may perform bidirectional signal transmission.
In some instances, it can reduce complexities of interconnections.

cause the whole system to fail. When a logic function is made to have bidirectional characteristics if one element becomes nonfunctional it could be bypassed by some transmission path. With a bidirectional system, it could be made more reliable in case of certain failures. This is somewhat analogous to the behavior of a neuron (3, 40-53) that can send or receive commands in either direction.

III. NEGATIVE RESISTANCE ADAPTIVE LOGIC CIRCUITS

A tunnel diode has three modes of unique operating characteristics. It can operate in switching, linear amplification, and oscillation mode. In the switching mode, it may be used as a logic element. Since it possesses low impedance in both "ON" and "OFF" states, subnanosecond switching speed can be achieved. This fast switching speed is very appealing for use in high speed logic circuits.

Although a tunnel diode may not be chosen in a practical adaptive logic realization (because of difficulty in large-scale integration and lack of input-output isolation), it will be considered here since it is one of the simplest and fastest devices.

By proper design of the load line, and by using proper signal, tunnel diodes can perform various logic functions, such as AND-gate, OR-gate, memory, majority, flip-flop, shift register, pulse countdown, and Schmidt trigger, as well as solving certain higher order Boolean functions.

3.1 Tunnel Diode Logic Circuits

A unique tunnel diode characteristic is its negative resistance region. A typical tunnel diode in a switching mode is shown in Figure 3.1. For a tunnel diode to operate as a switch the external resistance must be greater than the negative resistance of the tunnel diode. In this mode the tunnel diode operates as a bistable

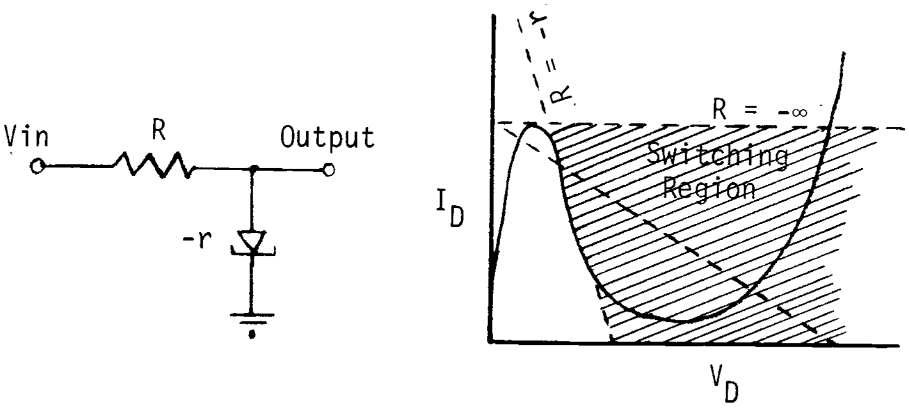


Figure 3.1. Tunnel diode in switching mode.

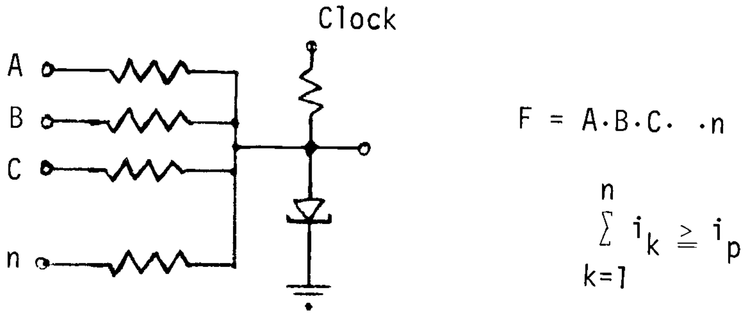

Clock
A
B
C
n
$F = A \cdot B \cdot C \cdot \ \cdot n$
$\sum_{k=1}^{n} i_k \geqq i_p$

Figure 3.2. Tunnel diode AND-gate.

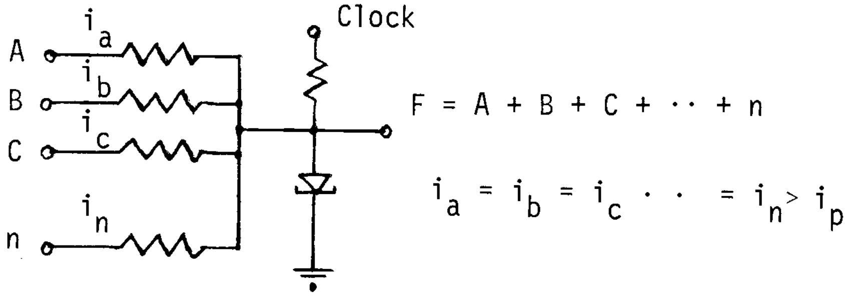

Clock
i_a
i_b
i_c
i_n
A
B
C
n
$F = A + B + C + \cdots + n$
$i_a = i_b = i_c \cdot \cdot = i_n > i_p$

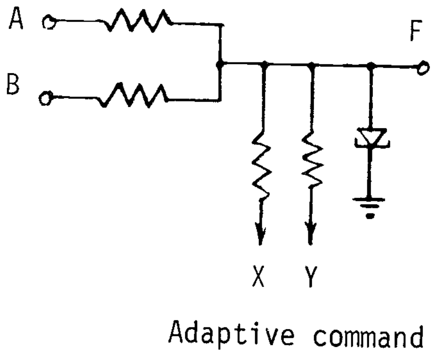
Figure 3.3. Tunnel diode OR-gate.

device; it will switch from low level to high level if the supplied current is greater than the peak tunnel diode current. It, therefore, can be recognized as a "0" or "1" state.

Since the tunnel diode peak current point is very stable over temperature, aging, and relatively free from radiation effects, a stable tunnel diode logic circuit such as AND-gate, OR-gate is possible. To perform AND-gate, additional input resistors are provided to the diode as shown in Figure 3.2. For given signal voltage the input resistors are chosen such that summation of all the current in the input resistor is slightly greater than the tunnel diode peak current. Therefore, output results only when all the input signals are in coincidence. To perform OR-gate operation, additional bias current is provided to the tunnel diode so that any current supplied by the input signal is sufficient to overcome the tunnel diode peak current point. Thus, any signal at input can flip the diode to the "ON" state. These basic tunnel diode logic circuits are used for developing adaptive logic circuits.

3.2 Simple Adaptive Logic Circuit

The basic adaptive logic circuit is shown in Figure 3.4. In addition to the input A and input B adaptive command signal X and Y are provided. These two adaptive command controls can have four possible combinations. This will make the basic logic circuit an AND-gate, OR-gate, inhibit or memory as determined by these adaptive command controls. This allows the logic to be adaptive to its own



1) AND gate with $(X \cdot Y)$

$F = A\ B$

2) OR gate, $(X \cdot Y)$

$F = A + B$

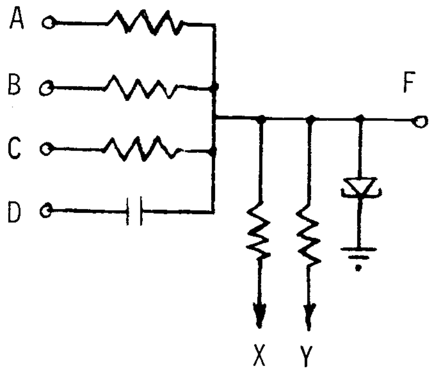
3) Inhibit, $(\bar{X} \cdot \bar{Y})$

$F = 0$

4) Memory,

$F = (1)\ /\ a + b$

Figure 3.4. Basic Adaptive Logic Circuit.



1) AND, $(X \cdot \bar{Y} \cdot \bar{C} \cdot \bar{D})$

$F = A\ B$

2) OR, $(X \cdot Y \cdot \bar{C} \cdot \bar{D})$

$F = A + B$

3) Inhibit, $(\bar{X} \cdot \bar{Y} \cdot \bar{C} \cdot \bar{D})$

$F = 0$

4) Memory, $(X \cdot \bar{Y})$

$F = (1) \rbrack a + b + c + d$

5) Majority, $(X\ Y\ D)$

$F = A \cdot B + B \cdot C + A \cdot D$

6) Flip-Flop, $(\bar{X} \cdot Y)$

$F = D$

Figure 3.5 Basic four-input Adaptive Logic Circuit

logic functions for various operations. With additional input to the basic adaptive circuit some other functions can be performed, such as majority operation and flip-flop operation, as shown in Figure 3.5. These basic adaptive logic circuits are used in higher order adaptive circuits.

3.3 Higher-order Adaptive Logic Circuits

The basic adaptive logic circuit is combined to form a higher-order adaptive logic circuit as shown in Figure 3.6a. All the logic circuits are tied to the neighboring logic circuits by resistors. There are three different operations incorporated in this circuit. It consists of adaptive logic nodes, memory nodes, and programming nodes. Each of these nodes is an adaptive resistor-tunnel diode logic circuit. Since the logic circuits are connected to various memory circuits and programming circuits, the logic circuit must take into account the neighboring logic nodes. During the logic operation the memory circuit may be made to detect certain favorable conditions which are initially sent to the programming node. After several logical operations, the adaptive logic can learn limited operations with the help of the memory units which are placed around the logic circuit. The learning of memory is aided by the programming node.

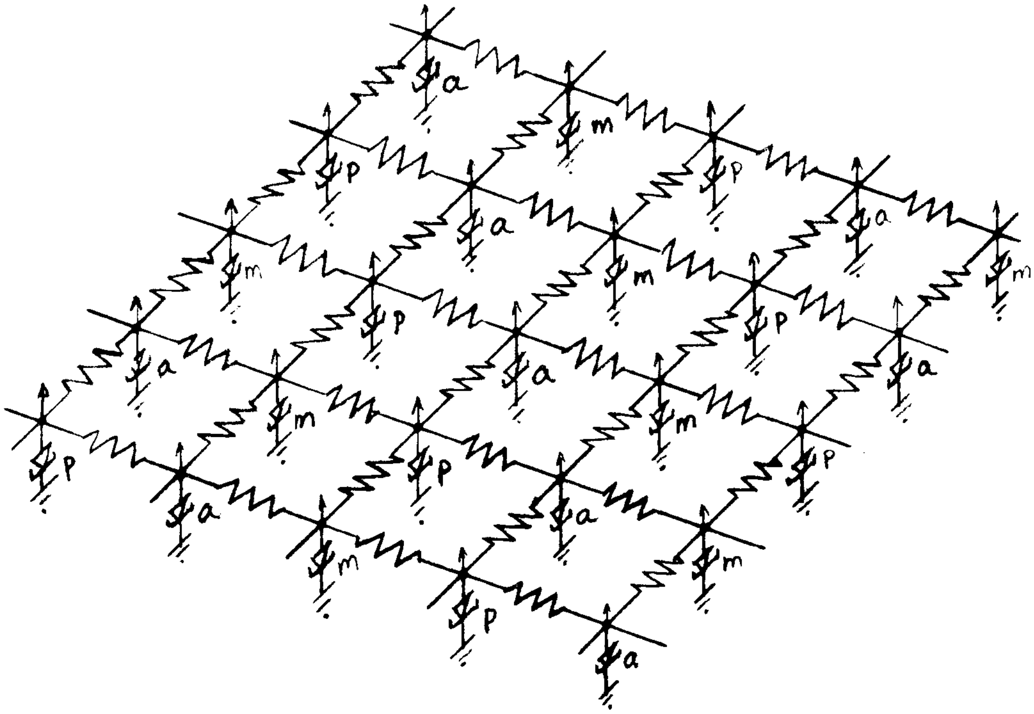


Figure 3.6a. Higher order adaptive logic circuit
(a) adaptive logic node,
(p) programming node, (m) memory node.

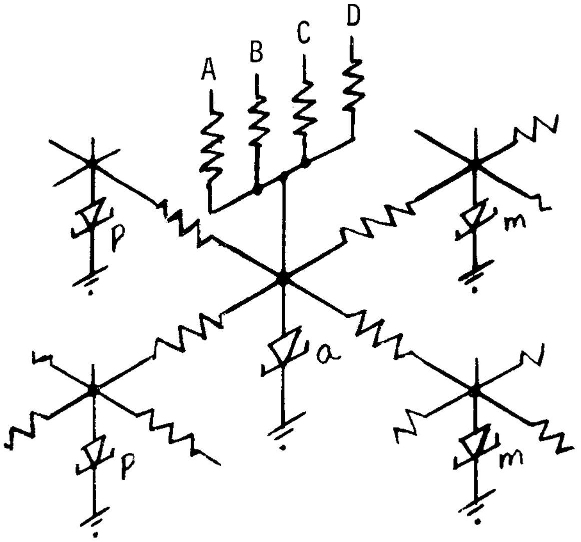


Figure 3.6b. Detail of the higher order adaptive logic with inputs.

IV. DEVELOPMENT OF ADAPTIVE LOGIC CIRCUIT

4.1 Bistable Logic Elements using Transistors

Most threshold semiconductor devices such as Shockley diodes, SCR's, thyristors and unijunction transistors exhibit a high impedance state when they are "OFF" and a low impedance state when they are "ON".

As it was pointed out in the previous section, negative resistance logic elements exhibiting low impedance characteristics in both the "ON" and "OFF" condition, such as the tunnel diode, are compatible with high speed decision making.

It is possible to implement a logic element as described above using transistors only.

In Figure 4.1a, a direct coupled NPN pair is shown.

The V-I characteristics of terminals AB exhibit a negative resistance as shown in Figure 4.1b and "ON" and "OFF" states which closely resemble those of a tunnel diode.

This circuit operates as follows:

Assume that Q_1 was initially "ON", therefore the impedance looking into terminal A is the collector saturation resistance of Q_1. As the current into terminal A is increased so will the base current of Q_2 and therefore Q_2 collector current. This tends to decrease the base current of Q_1. Consequently, Q_1 collector current decreases, which in turn adds to the base current of Q_2 to regenerate the cycle

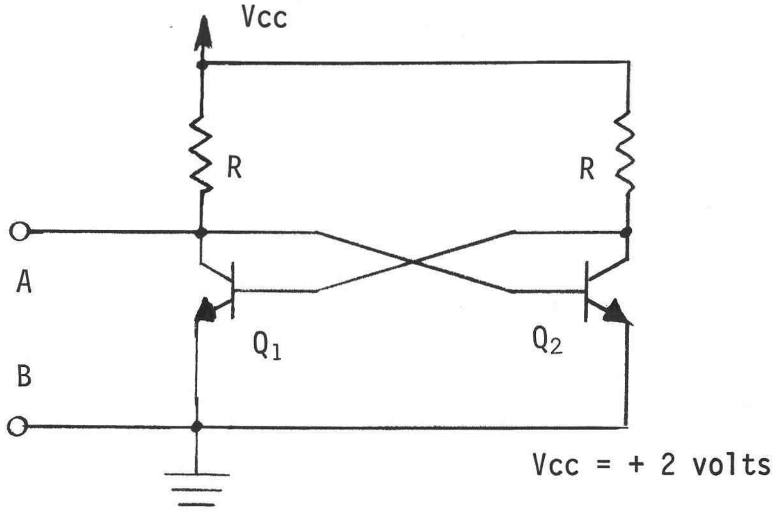


Figure 4.1a. Npn cross-coupled bistable pair.

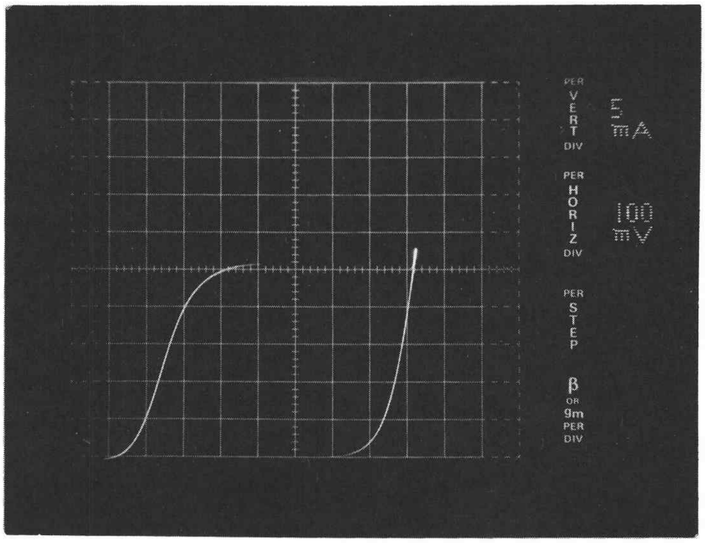


Figure 4.1b. Voltage-current characteristics of the npn cross-coupled bistable pair, seen from the terminal A and B.

until Q_2 is fully "ON" and Q_1 is off. The impedance looking into terminal A is that of the forward biased base emitter junction of Q_2.

4.2 NPN-PNP Bistable Pairs

A second scheme for implementing the desired logic element uses an NPN-PNP pair connected as shown in Figure 4.2a. The V-I characteristics are shown in Figure 4.2b . When both Q_1 and Q_2 are off, the impedance looking into terminals AB is essentially R since the base is near zero volts. As the voltage V_{AB} is increased, Q_1 starts to conduct, which causes Q_2 to conduct and regeneration takes place and both transistors conduct heavily. The impedance looking into terminals AB is R/2 in parallel with the impedance of the forward biased base emitter junction of Q_1.

4.3 Adaptive Logic Circuit

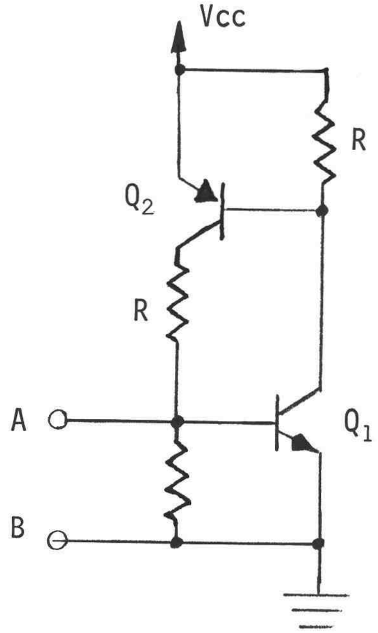
A possible adaptive logic circuit, employing the NPN-PNP pair element described previously, is shown in Figure 4.3.

This arrangement can perform various logical operations such as AND, OR, NAND, NOR, MAJORITY, LEAST, two out of four, ALWAYS, NEVER and MAYBE, by simply changing current i_a.

In this configuration, i_a acts as a negative bias to the input.

In order to obtain an output of "1" at Vo, the input voltage at the current summing node must be positive enough to forward bias the base emitter junction of Q_1 (about + .6V for a silicon transistor).

In order to ease the following analysis, the base-emitter forward bias voltage-drop will be neglected.



Vcc = + 2 volts

Figure 4.2a. Npn-pnp bistable pair.

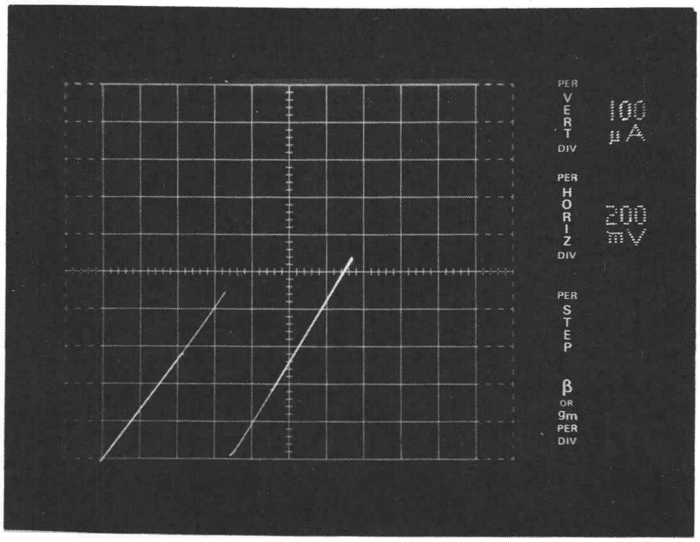


Figure 4.2b. Voltage-current characteristics of the npn-pnp pair seen from the terminal A and B.

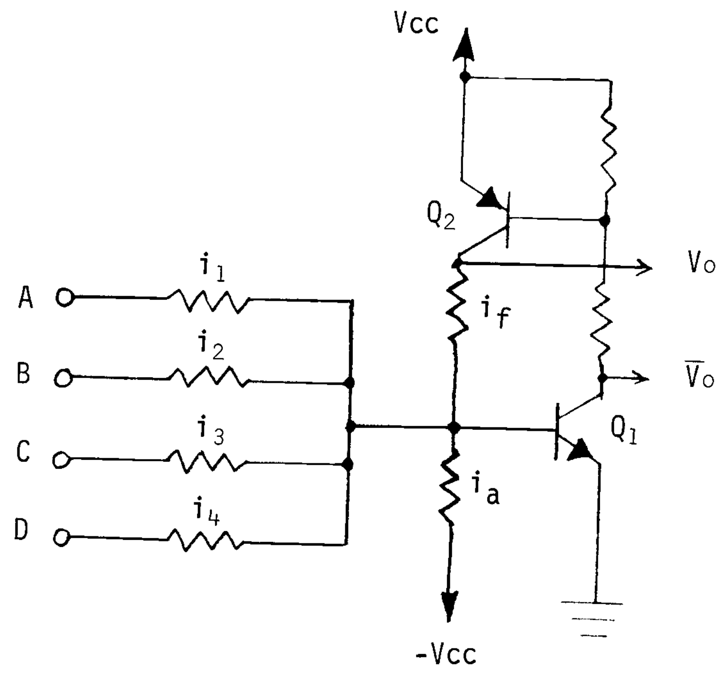


Figure 4.3. Four-input adaptive logic circuit. By changing adaptive current ia, various Boolean logic functions may be performed.

Then the input currents i_n, the adaptive current i_a and the output voltage must obey the following relationships:

$$V_o = 1 , \qquad \sum_{n=1}^{4} i_n + i_a > 0$$

$$V_o = 0 . \qquad \sum_{n=1}^{4} i_n + i_a < 0$$

For example, if a negative bias -3.5 units of current is forced through R_A, it will take all four inputs to overcome the negative bias of -3.5.

An OR logic function may be performed by changing adaptive current, i_A, to -.5 units.

If any one of four inputs receives a signal, it will overcome the negative bias and turn on Q_1, thus producing a logic "1".

Likewise, by forcing different adaptive currents, several Boolean logic functions may be possible without changing circuit interconnections. Most conventional logic circuits require a change of circuit configuration to obtain a different logic function.

Using adaptive logic elements as basic building blocks, various sequential operations such as shown in Table I and Table II can be performed.

The logic function, MAYBE, can be generated by applying statistically randomized current to i_a. In this mode, current through i_a fluctuates randomly between - 4.5 to + 1 unit of current.

TABLE I. CHARACTERISTICS OF THE ADAPTIVE LOGIC CIRCUIT*.

Adapt Current	Logic for Vo	Boolean Expression at Vo
+ 1	ALWAYS	1
- 0.5	OR	A + B + C + D
- 1.5	2 out of 4	A·B + A·C + A·D + B·C + B·D + C·D
- 2.5	MAJORITY	A·B·C + B·C·D + A·C·D + A·B·D+ A·B·C·D
- 3.5	AND	A·B·C·D
- 4.5	NEVER	0
- X_n**	MAYBE	0 or 1

TABLE II. CHARACTERISTICS OF THE ADAPTIVE LOGIC CIRCUIT* OBSERVED AT THE COMPLEMENTARY OUTPUT, Vo.

Adapt Current	Logic for $\overline{V}o$	Boolean Expression for $\overline{V}o$
+ 1	NEVER	0
- 0.5	NOR	$\overline{A + B + C + D}$
- 1.5	2 out of 4	A·B + A·C + A·D + B·C + B·D + C·D
- 2.5	least	$A\cdot\overline{B}\cdot\overline{C}\cdot\overline{D} + \overline{A}\cdot B\cdot\overline{C}\cdot\overline{D} + \overline{A}\cdot\overline{B}\cdot C\cdot\overline{D} + \overline{A}\cdot\overline{B}\cdot\overline{C}\cdot D$
- 3.5	NAND	$\overline{A\cdot B\cdot C\cdot D}$
- 4.5	ALWAYS	1

* Circuit diagram shown in Figure 4.3.

** X_n is a statistically varying random noise.

Normalized Gaussian distribution of random noise (4,48-49) can be expressed as

$$Y = P(X) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{X - \mu}{\sigma}\right)^{2}}$$

where

σ = standard deviation of the distribution (rms value)

μ = mean of the distribution

π = a constant

e = a constant

Y = probability of X, or P(χ)

X = value of random variable

Adaptive current X_n for MAYBE logic has the probability distribution

$$P(X_n) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{X_n - i_e}{1}\right)^{2}}$$

where

i_e = mean value, desired logic current for given logic function between -4.5 to + 1

standard deviation was assumed unity

Output of the circuit will consist of random pulses superimposed on the signal.

The logic MAYBE, though it has been of academic interest in the past, could be made useful in some instances when statistical decisions must be made due to lack of sufficient information.

In the event of uncertainty in the system or lacking sufficient information, it is not possible to make instant decisions, thus it requires repeated operations in order to average out some noise or random effects.

4.4 Adaptive Binary Counter

Most of the conventional binary flip-flop circuits use storage elements such as a capacitor. Switching from one transistor to the other must take place while the charge in the capacitor lasts. If the transistor does not switch during this period, commutating from one transistor to the other will be uncertain. The risetime of the input signal cannot be slower than the storage capacitor time constant; therefore, the input drive waveform must be controlled to be compatible with a specific circuit.

An adaptive binary counter operates on the principle of current division, which makes the circuit insensitive to the variation in input drive and the waveform. A simplified schematic circuit diagram is shown in Figure 4.4.

For ease in analyzing the adaptive binary divider operation, assume electron current rather than conventional current flow. A

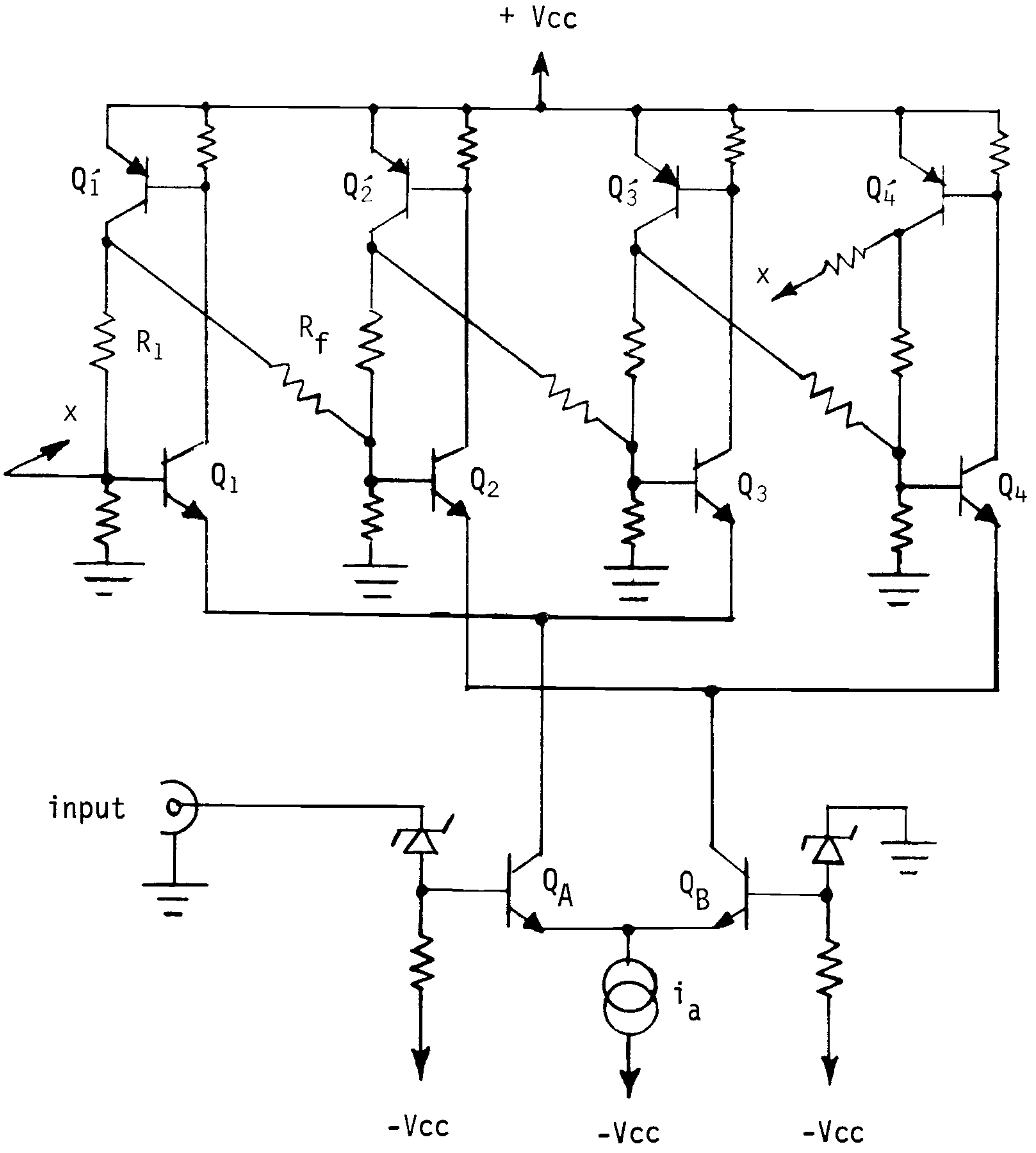


Figure 4.4. Adaptive binary counter.

logic level "high" was applied to the base of Q_A, via the zener diode, and it turns on Q_A. The current i_a will switch into the emitter of Q_A, and come out at the collector of Q_A.

This current which came out of the collector of Q_A can go into emitter of either Q_1 or Q_3, depending on their base voltage. Assume the base of the Q_1 has higher voltage, then the Q_1 turns on and Q_1' will also come on. The collector of Q_1' will be at "high" position through R_1. Q_1' will try to turn on the base of Q_1 more positive. The collector of Q_1' also presets the base of Q_2 positive, but Q_2 will not come on until current is supplied to the emitter of Q_2.

Now a low level voltage is applied to the input, causing i_b to switch from Q_A to Q_B turning on Q_2 since Q_2 base was preset to "high" level. This in turn, turns on Q_2' and the feedback current through R_2 tries to maintain Q_2 "ON". The collector of Q_2' will also preset the base of the next stage to a "high" level. With this preset process, each time the current i_a is switched from Q_A to Q_B transistors Q_1 through Q_4 switch sequentially. Since the switching takes place every half cycle of input, output frequency of any output will be one-half of input frequency, thus divide by two binary counting may be performed.

The advantage of the current-steered binary flip-flop circuit is that it can tolerate variations in input waveform since the circuit does not have any storage element. Another advantage of this approach is that reversible counting is performed easily and reliably by presetting the base of the following stage or the base of the previous stage. An example of such possibility is described next.

4.5 Adaptive n-bit Reversible Counter-Shift-Register

An adaptive n-bit shift register is shown in Figure 4.6.

The operation of this arrangement is very similar to the binary divider shown in Figure 4.4, but "forward" and "reverse" control lines are added. The preset forward-reverse resistors R_r and R_f are composed of source-drain resistance of field-effect transistors.

4.6 Field-Effect Transistor as Variable Resistor

The value of the source-drain resistance can be changed by controlling the bias voltage of the gate. The conductance, G_d, of a field-effect transistor is given (11,384-392)

$$G_d = G_{do} \left[1 - \left(\frac{V_{gs}}{V_p} \right)^{\frac{1}{2}} \right]$$

where G_{do} is a zero biased drain conductance. Vgs is the gate to the source voltage and V_p is a pinch-off voltage of a field-effect transistor.

The variation of Rd with Vgs can be closely approximated by the empirical expression

$$Rd = \frac{Ro}{1 - KVgs}$$

where Ro = drain resistance at zero gate bias

K = a constant

Vgs = gate-to-source voltage

The variable resistance characteristics of a field-effect transistor are shown in Figure 4.5b.

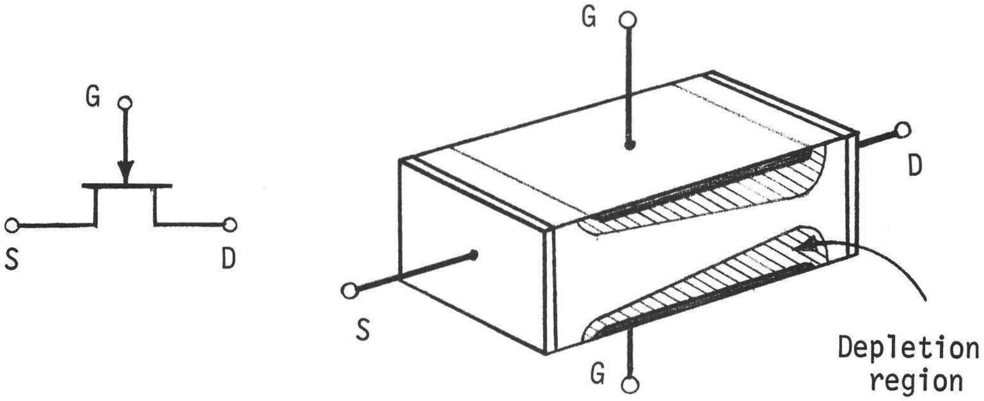


Figure 4.5a. Field-effect transistor and its basic structure of an n-channel FET. If a negative bias is applied to the gate, the depletion region moves in, thus changing the resistivity of the channel.

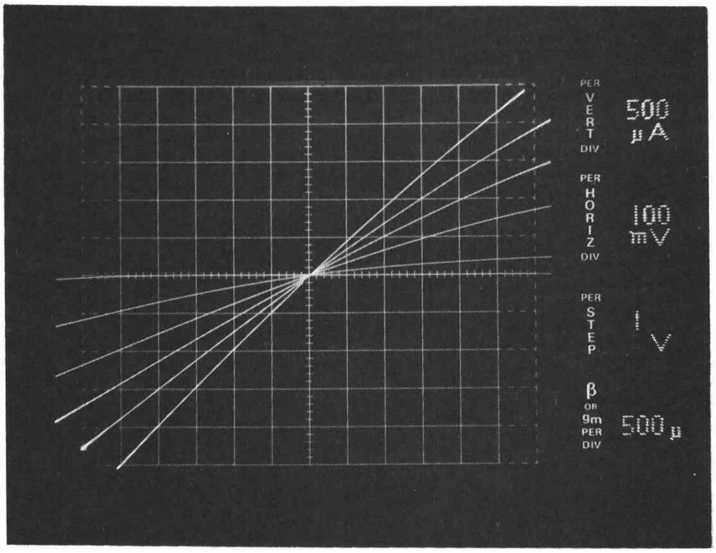
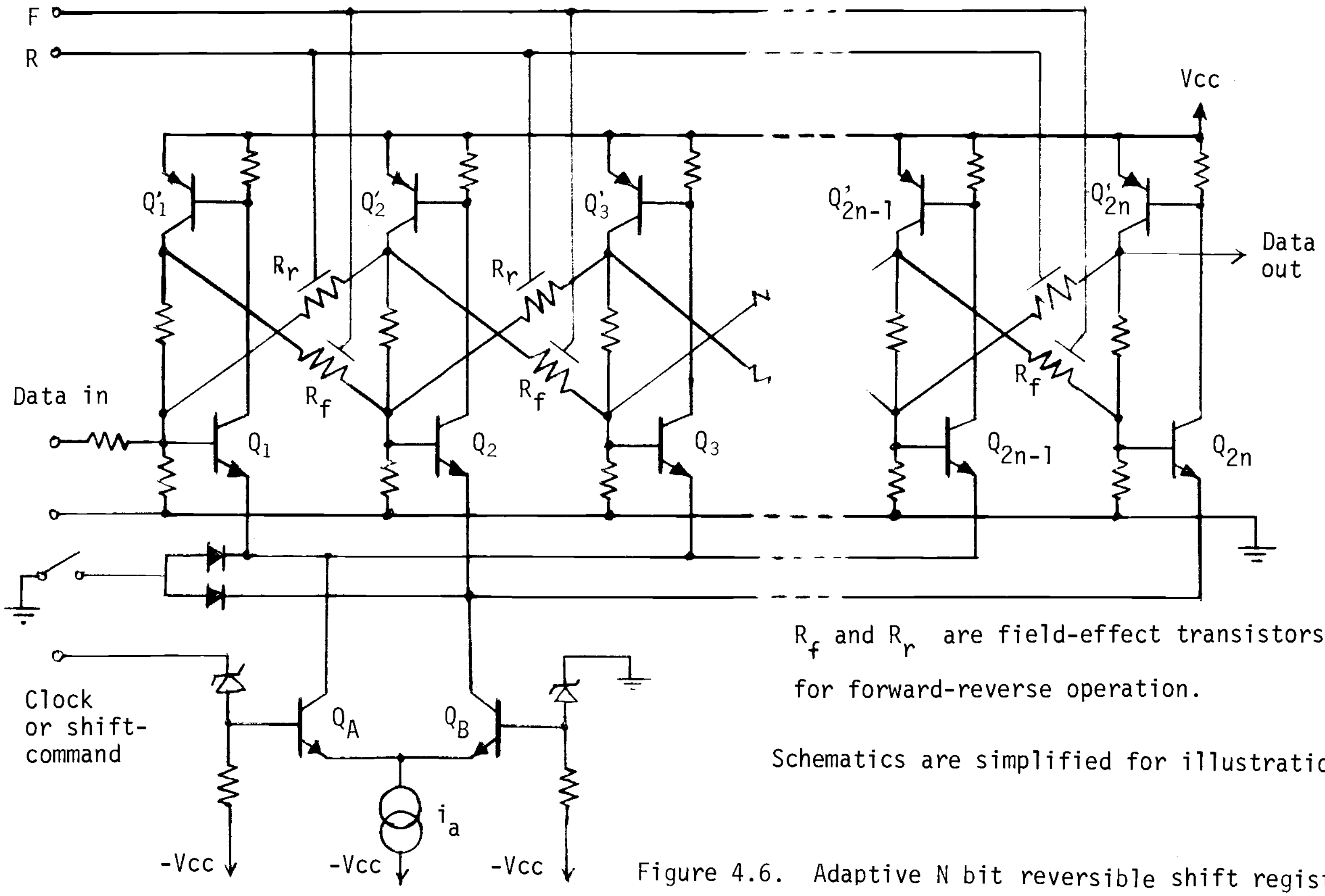


Figure 4.5b. Field-effect transistor source-to-drain characteristics for various bias, showing the various resistance characteristics.



R_f and R_r are field-effect transistors for forward-reverse operation.

Schematics are simplified for illustration.

Figure 4.6. Adaptive N bit reversible shift register

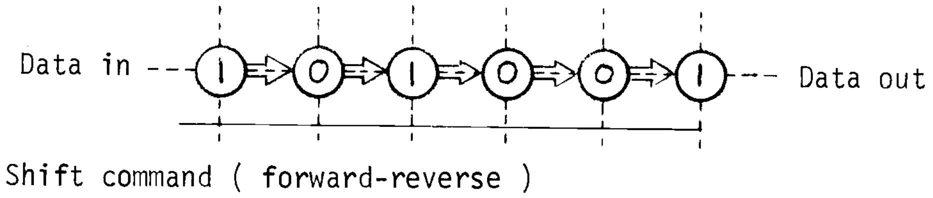


Figure 4.7a. Horizontal shift register with bidirectional-shift capabilities.

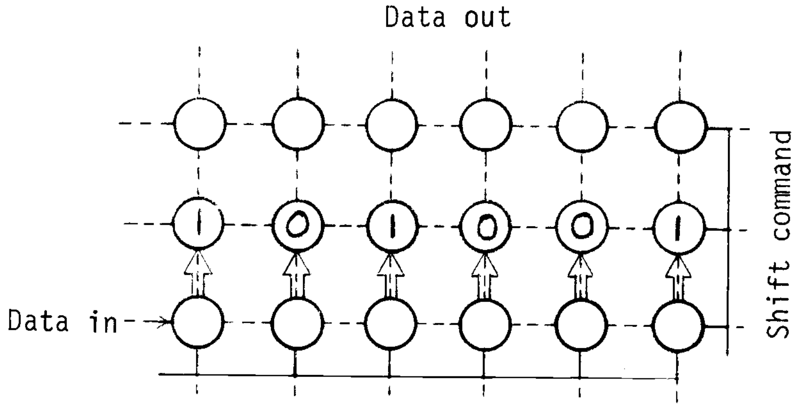


Figure 4.7 b. Vertical parallel-shift added.

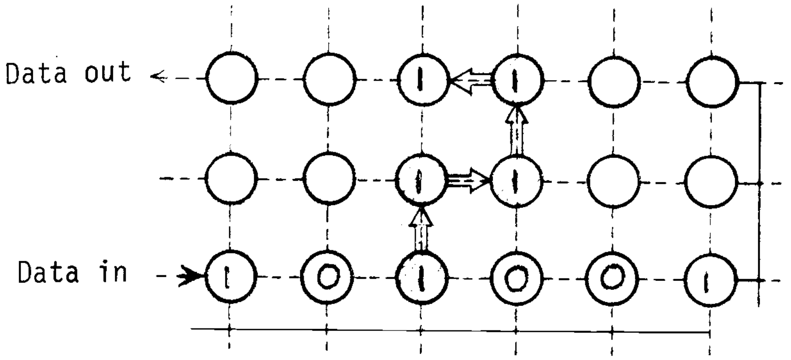


Figure 4.7c. Two-dimensional shift register allows pattern-shift both vertical and horizontal dimension allowing more flexibility.

4.7 Two-dimensional Serial-Parallel Shift-Register

By utilizing forward-reverse capabilities described in section 4.5 and illustrated in Figure 4.6, it is possible to add vertical-shift, up or down, in addition to the horizontal-shift operations.

This will further enhance the speed of certain arithmetic operations or memory operations by shifting parallel. To perform a shift-up operation with a serial shift-register, n clock pulses are required to read out one shift-register to the other unit.

Two-dimensional shift-register operations are illustrated in Figure 4.7a through Figure 4.7c.

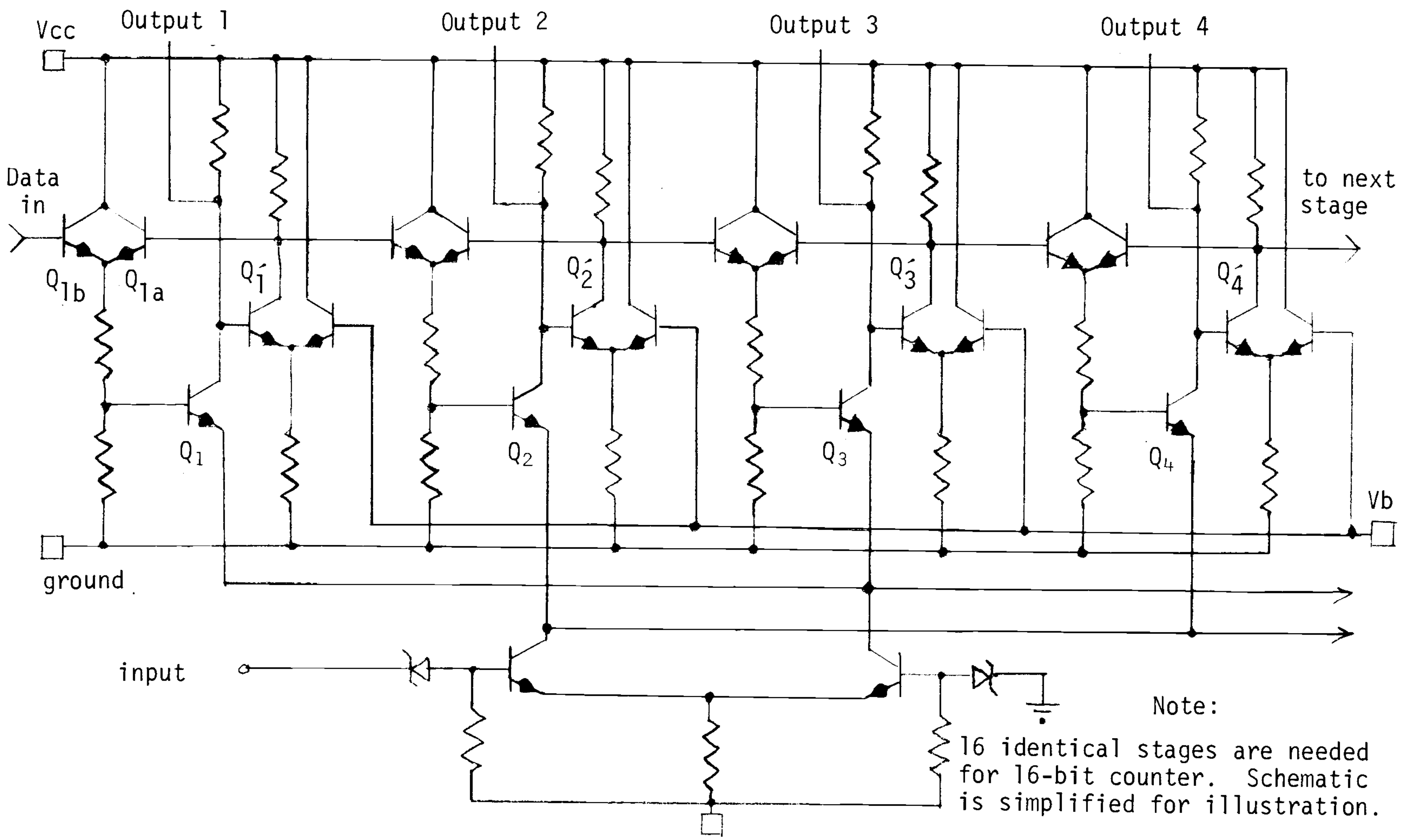
4.8 High Speed 16-bit Counter

A simplified schematic diagram of a high-speed 16-bit counter utilizing all npn transistors is shown in Figure 4.8. An all npn transistor version is used for integrated circuit realization and to achieve high-speed operation. The circuit operates on a current-steering principle similar to the circuit in Figure 4.4.

This type of logic (current steering) is particularly adaptive for very high speed since it has no commutating capacitors for coupling between stages. It can tolerate greater variation in input drive change and transistor parameter variations.

Reversible, and two-dimensional shift-register operations as pointed out in the previous section also can be implemented.

High-speed performance of the circuit is shown in Figure 4.9a. The top oscilloscope trace is an input clock of 500 MHz. The middle



Note:
16 identical stages are needed for 16-bit counter. Schematic is simplified for illustration.

Figure 4.8. High-speed 16-bit counter, utilizing all npn transistors for integrated circuit realization. Clock rate of 500 MHz with 1 ns bit-rate was achieved.

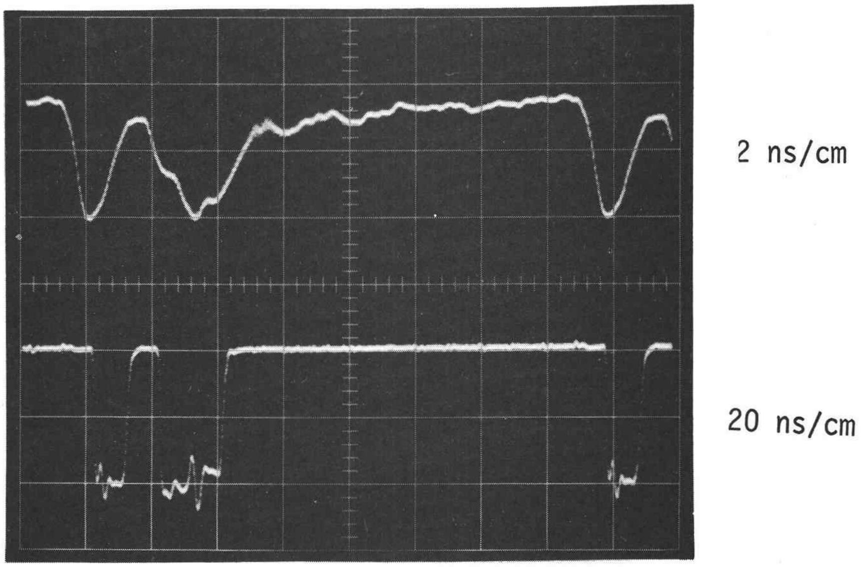


Figure 4.9a. 16-bit word generator. Photograph is a double exposure to illustrate waveform change with clock rate of 500 MHz and 50 MHz.

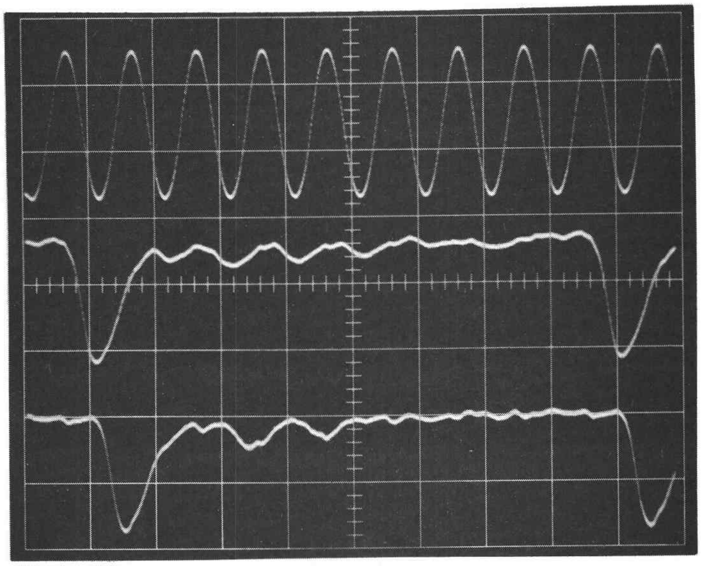


Figure 4.9b. 16-bit counter output. Top trace, 500 MHz input. Middle trace, output 1. Bottom trace, output 2. Vertical sensitivity: 500 mv/cm. Time base: 2 ns/cm.

trace is the waveform observed at output 1. The bottom trace is the waveform observed at output 2. The sweep speed of the time base is 2 ns/cm. Note that the time difference between bit-1 and bit-2 is 1 ns. This is explained since this type of counter shifts a bit every half cycle.

In Figure 4.9b, a serial-word 1011000000000000 is shown at two operating frequencies, 500 MHz and 50 MHz.

4.9 Read-only Strip-line Memory

For a high-speed operation, inductive memories have limitations at high frequencies (impedance of the inductors increases with increase of frequency, thus current decreases when driven by a constant voltage).

A strip-line or flat conductor which is placed over a grounding plane exhibits distributed inductance and capacitance.

Because of combined inductance-capacitance, the impedance of the line appears as a constant resistance over a very wide range of frequency. The impedance of a line is:

$$Z = \sqrt{\frac{L}{C}}$$

where

L = the total inductance
C = the total capacitance

and the delay time of a line is:

$$t_d = \sqrt{L\ C}$$

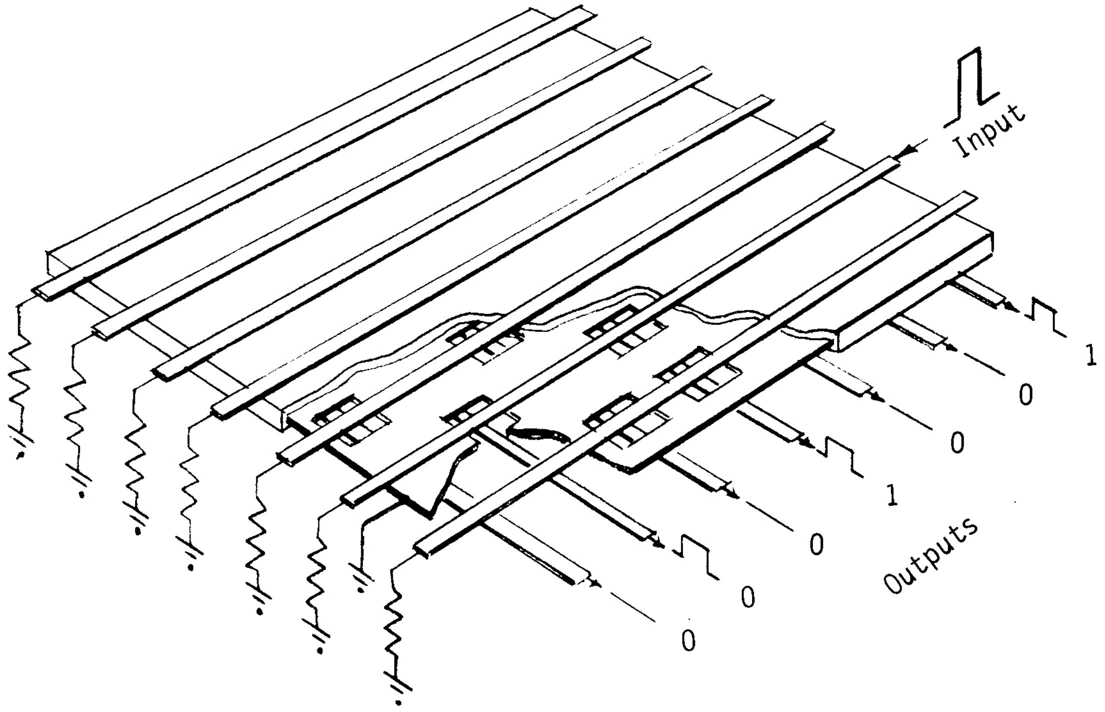


Figure 4.10. Strip-line read-only memory.
If a signal is applied to the top strip line, the signal will propagate to the other end, and wherever an opening on the middle ground plane, the pulse will appear on the bottom line.

Referring to the arrangement shown in Figure 4.10, if a signal is applied to one of the input delay lines, the signal will propagate along the line and the signal energy will be terminated by the termination resistor at the end of the line. If there is a ground plane between the top and bottom lines, no signal appears on the bottom output line. If a hole is cut in the ground plane as shown, part of an input signal will appear at the bottom output line.

By selectively opening the middle metalization, a fixed word or coded pulses can be generated. Such a read-only memory can be realized by etched circuit boards or possibly by integrated circuit, multi-layer metalization techniques.

4.10 Adaptive Two-channel Delay Line

In high-speed digital operations, delay lines find a variety of applications, such as time delay matching, timing standard, data transmission lines, and recirculating data storage. In some of the critical high-speed operations, delay time matching is a problem. For example, if a high density information-serial word is stored in recirculating delay lines, the delay time of the clock must be carefully matched to that of the signal at the output.

A conventional two-channel delay line is shown in Figure 4.11a. The delay time matching at the outputs will be subjected to the delay time variations in each delay line.

A possible adaptive two-channel delay line using only one delay line is shown in Figure 4.11b. Since it utilizes only one delay

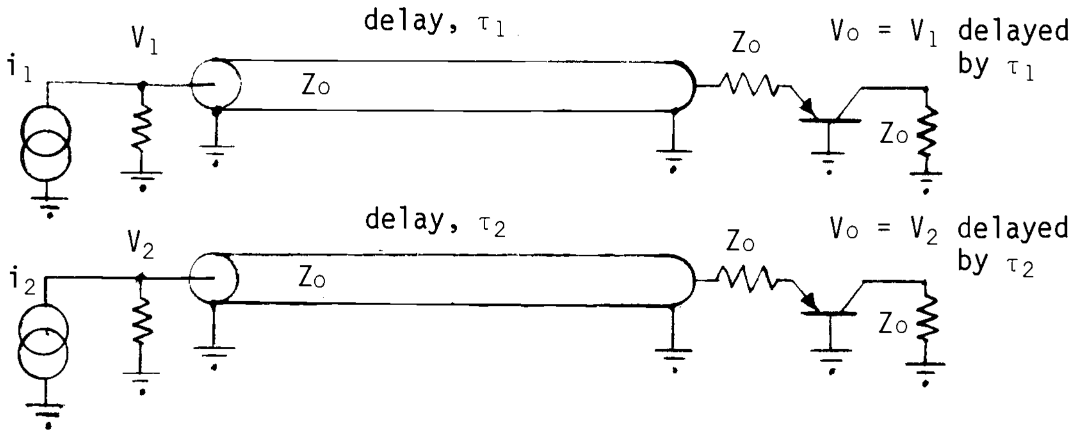


Figure 4.11a. Conventional dual delay line. The delay time matching subject to the variation of τ_1 and τ_2.

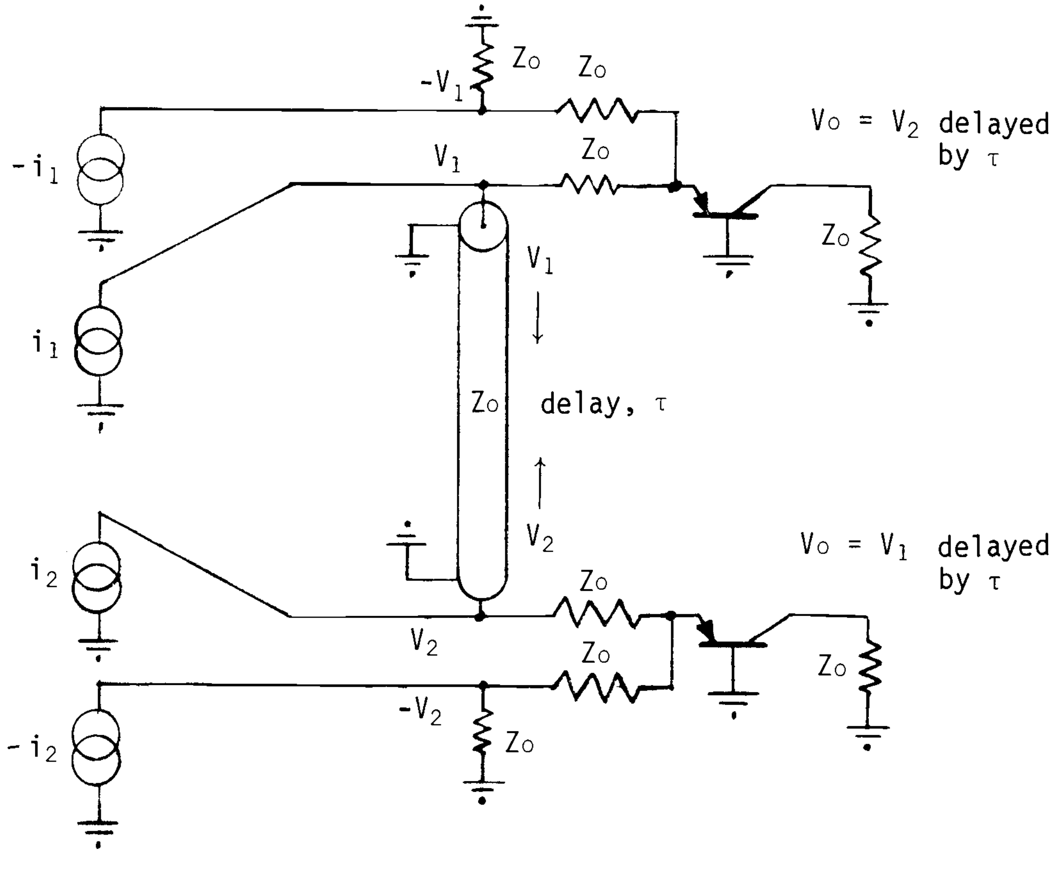


Figure 4.11b. Adaptive delay-line can delay two independent signals and maintain precise delay matching.

line for both channels, identical delays will be ensured in spite of variation in the delay characteristics. This is particularly important, when data through the long delay line is to be gated with the clock at the output. In this arrangement, the data current i_1 appears output of Q_1, after delay τ, and i_2 appears output of Q_2, after delay τ, sharing the same delay line. To analyze the operation, assume i_1 and - i_1 currents were turned "ON". At the junction of delay line and the termination resistor, the current i_1 will split in half since both impedances are equal. The voltage at the sending-end, V_1, will be

$$V_1 = \frac{i_1 Z_o}{2}$$

The wave-front, V_1, will travel through the delay line and reappear at the other end and will be terminated by the load Z_o. The current which flows in to the Z_o will flow in to the emitter of grounded base transistor and reappear at the collector (for simplicity, assume the emitter resistance is negligible and the emitter to the collector current gain, alpha, of the grounded-base state is unity). This will result in an output current at the collector Q_2 equal to one-half of i_2, thus generating voltage of V_1.

Meanwhile, at the emitter of Q_1, one-half of i_1 will be received due to i_1 and one-half of - i_1 due to - i_1. Since two currents are equal and opposite, the net current equals zero, thus there will be no i_1 output result at the collector of Q_1.

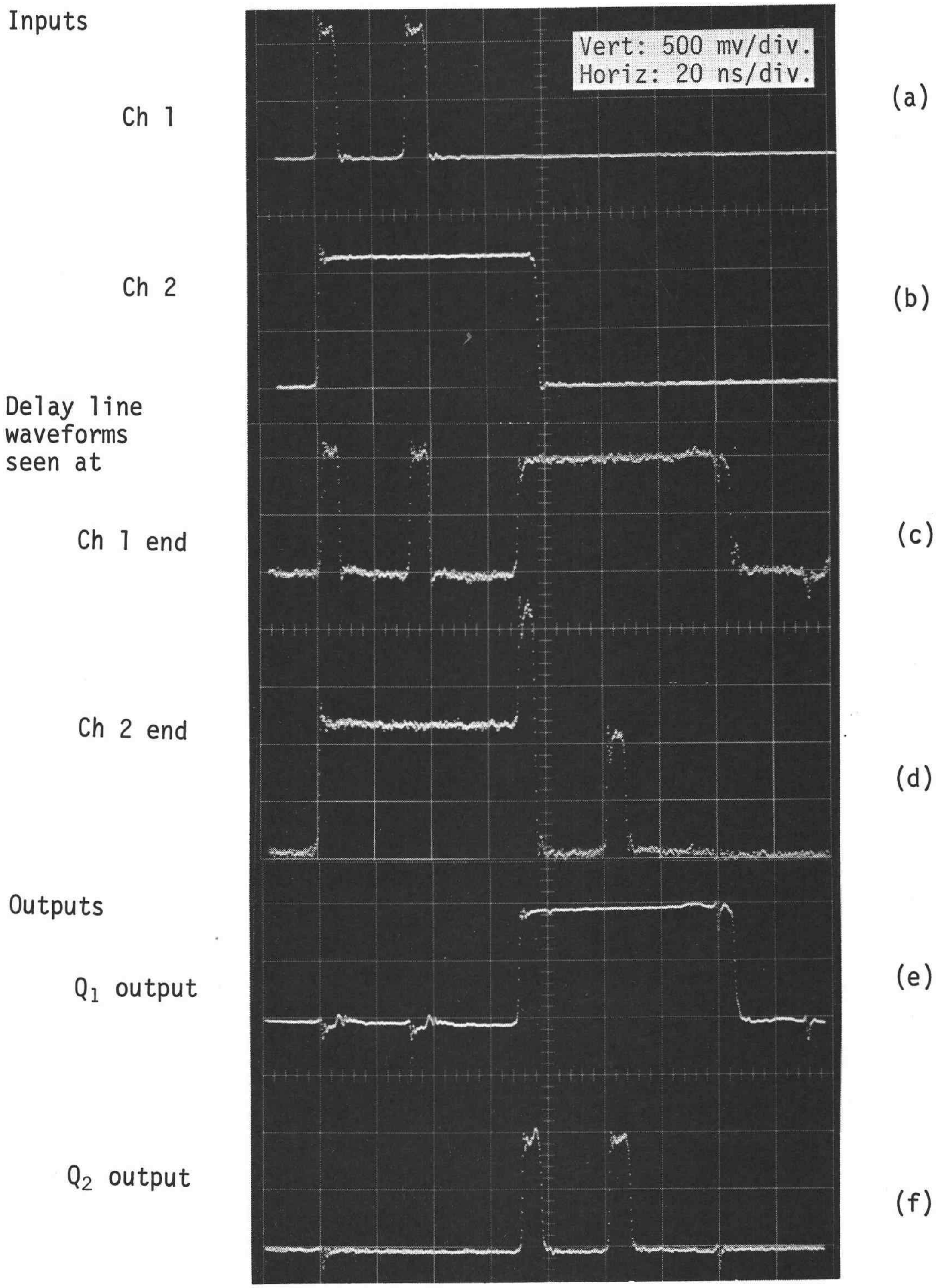


Figure 4.12. Adaptive two-channel delay line. Photographs showing inputs, composite waveform at delay line and the outputs.

Likewise, signal i_2 will reappear only at the opposite transistor Q_2 and no output at the collector of Q_1.

In Figure 4.12, input and output waveforms of an adaptive two-channel delay line is shown. A double-pulse was applied to channel 1 and a single 80 ns pulse was applied to channel 2. A composite waveform seen at the channel 1 end of the delay line is shown in Figure 4.12 (waveform, c). The composite waveform consists of the undelayed channel-1 signal and the delayed channel-2 signal. When the composite signal goes through transistor Q_1 (operating as a grounded base current summing amplifier with gain of close to unity), the unwanted channel-1 signal will be subtracted, thus resulting in the delayed channel 2 signal at the output of Q_2 as shown in Figure 4.12 (waveform, e). Likewise, the other signal, channel-1, will appear at the output collector of Q_1 after delay time of τ (Figure 4.12; waveform, f).

As demonstrated, it is possible to make an adaptive delay arrangement which can provide exactly matched delay-time using a single delay line.

V. DISCUSSION OF RESULTS AND CONCLUSIONS

Adaptive logic circuits described in section 4.3 can perform a variety of logical operations such as AND, OR, NAND, NOR, MAJORITY, LEAST, etc., by changing the adapt current i_a.

Using such a flexible logic element it is possible to reduce the complexity of conventional data processing operations.

An adaptive reversible 16-bit shift register using field-effect transistor as variable resistor was described in section 4.5.

Possibilities of high speed logic operation using current-switching mode operation are shown in section 4.6. As an example, a 16-bit counter with a 1 ns output bit rate was shown.

The read-only strip-line memory, described in section 4.9, may allow operation in the region of subnanoseconds. This can further enhance the speed of some mathematical conversion time.

The two-directional delay line which can transmit two independent signals allows several possibilities such as doubling storage capacities if it was used as a recirculating memory, another use is for interconnecting a signal from one location to another without doubling the number of wires.

In time-shared operation with a single line, digital data cannot be sent and received simultaneously, thus it requires certain dead time after the end of one message to receive another message.

5.1 Conclusions and Recommendation

Most of the prior work on adaptive logic was relatively slow.

Adaptive logic proposed here is more applicable to the high speed logic operations. A variety of logic functions may be performed without changing the circuit interconnections.

With integrated circuit techniques, much new adaptive logic became realizable. Although what has been proposed here is a very simple adaptation logic, future adaptive logic investigation should be directed in conjunction with I. C. capabilities.

BIBLIOGRAPHY

1. Brian, A. et al. A large, self-contained learning machine. WESCON Convention Record 6.1. 1963. 9 p.
2. Crafts, H. S. A magnetic variable-gain component for adaptive networks. Stanford, California, 1962. 64 p. (Stanford University. Stanford Electronics Laboratories. Technical Report no. TR1851-2)
3. Crane, H. D. Neuristor studies. Stanford, California, 1960. 191 p. (Stanford University. Stanford Electronics Laboratories. Technical Report no. TR1506-2)
4. Dixon, Wilfrid J. and Frank J. Massey, Jr. Introduction to statistical analysis. 2d ed. New York, McGraw-Hill, 1957. 488 p.
5. Hoff, M. E., Jr. Learning phenomena in networks of adaptive switching circuits. Stanford, California, 1962. 85 p. (Stanford University. Stanford Electronics Laboratories. Technical Report no. TR1554-1)
6. Jones, L. E. On the choice of subgoals for learning control systems. Institute of Electrical and Electronic Engineers, Transactions on Automatic Control 13:613-621. 1968.
7. Ledley, Robert Steven. Digital computer and control engineering. New York, McGraw-Hill, 1960. 835 p.
8. Low, P. R. et al. Components for trainable systems. International Solid-State Circuits Conference, Digest of Technical Papers 5. 1962.

9. Low, P. R. Influence of component imperfections on performance of trainable systems. WESCON Convention Record 6.3. 1963. 9 p.

10. Mattson, R. L. The analysis and synthesis of adaptive systems which use networks of threshold elements. Stanford, California, 1962. 95 p. (Stanford University. Stanford Electronics Laboratories. Technical Report no. TR1553-1)

11. Millman, Jacob and Christos C. Halkias. Electronic devices and circuits. New York, McGraw-Hill, 1967. 752 p.

12. Mishkin, Eli and Ludwig Braun, Jr. (eds.). Adaptive control systems. New York, McGraw-Hill, 1961. 533 p.

13. Phister, Montgomery, Jr. Logical design of digital computers. New York, Wiley, 1958. 408 p.

14. Pierce, W. H. Improving reliability of digital systems by redundancy and adaption. Stanford, California, 1961. 121 p. (Stanford University. Stanford Electronics Laboratories. Technical Report no. TR1552-3)

15. Russo, F. A. and R. J. Valek. Process performance computer for adaptive control systems. Institute of Electrical and Electronic Engineers, Transactions on Computers 17:1027-1036. 1968.

16. Tsypkin, Ya. Z. Self-learning. Institute of Electrical and Electronic Engineers, Transactions on Automatic Control 13:608-612. 1968.

17. Vendelin, G. D. A silver sulfide adaptive component. Stanford, California, 1963. 71 p. (Stanford University. Stanford Electronics Laboratories. Technical Report no. TR1853-1)

18. Widrow, Bernard and James B. Angell. Reliable, trainable networks for computing and control. Stanford, California, 1962. 123 p. (Stanford University. Stanford Electronics Laboratories. Technical Report no. TR1554-2)