SHORT COURSE LECTURE NOTES ON DIGITAL USE OF CHARGE COUPLED DEVICES

DR. T. A. Zimmerman
TRW Defense and Space Systems Group, Redondo Beach, Ca.

In general CCD's are a good match for certain kinds of signal processing structures. Figure 1 indicates the general characteristics of the favorable structures. We see from the figure that structures which: can be pipelined; have feed forward characteristics; use iterative circuits cells; have a high shift register or memory content; and finally are channelized or can be arranged in parallel processing lines are the best matched to this technology. It is also true that some processing functions can be converted to that form if they are not in that arrangement to start with. This is an important point that is often overlooked in architecturing systems to be used with CCD's.

So CCD's are well matched to certain signal processing structures; why, then, would we use them in a digital mode? Figure 2 summarizes the gist of the argument for the use of digital CCD's. First any digital device is desirable because it has fixed accuracy (n-bits); it is flexible in its application and able to adapt itself to many uses; the characteristics are repeatable from device to device. If we add to these lists of characteristics the two most important CCD characteristics, low power and high functional density, then we have acombination that is very difficult to beat.

The third chart summarized some of the important digital CCD characteristics. We see listed there that the circuits are large and complex due primarily to the high component density capable within the technology. The clock speeds are not blinding fast but are moderate, typically 5 MHz or so. It is important to note that arithmetic functions which require feedback are not at all well suited to the technology; it is much better matched to devices where we have a feed forward, pipeline algorithm or system to be implemented.

One important question is always raised when talking about any new technology and CCD's are no exception. What do the interfaces look like? With the charge coupled device technology we must remember that we have basically an analog device. However, it is not difficult to interface this analog device with the digital

CCD IS A GOOD MATCH TO CERTAIN SIGNAL PROCESSING STRUCTURES

- GENERIC CHARACTERISTICS OF FAVORABLE STRUCTURES
  - PIPELINE (THE EXTRA STORAGE IS "FREE")
  - FEEDFORWARD
  - ITERATIVE CIRCUIT CELLS
  - HIGH SHIFT REGISTER/MEMORY CONTENT
  - CHANNELIZED OR PARALLEL PROCESSING
- SOME SIGNAL PROCESSING FUNCTIONS CAN BE REASONABLY CONVERTED TO A FORM MATCHED TO CCD

FIGURE 1

WHY DIGITAL ....

FIXED ACCURACY
FLEXIBLE IN APPLICATION
REPEATABLE CHARACTERISTICS
REPROGRAMMABLE

.... CCD

LOW POWER
HIGH FUNCTIONAL DENSITY

FIGURE 2

DIGITAL CCD CHARACTERISTICS

- LARGE, COMPLICATED CIRCUITS POSSIBLE BECAUSE OF HIGH COMPONENT DENSITY.
- MODERATE CLOCK SPEEDS - 5 MHZ TYPICAL FOR TODAY'S CIRCUITS
- "STREAMING" FUNCTION BEST SUITED FOR CCD IMPLEMENTATION WITH PIPELINE TECHNIQUES.
- ARITHMETIC OPERATIONS WHICH REQUIRE FEEDBACK ARE NOT AS WELL SUITED TO CCD IMPLEMENTATION.

FIGURE 3

CCD INTERFACES

- CLOCK VOLTAGES ARE TYPICALLY 10-20V WITH 5-10V CLOCK SWINGS
- ELECTRICAL ANALOG INPUT GENERALLY REQUIRES A 5-10V BIAS
- DIGITAL INPUT CAN BE TTL COMPATIBLE
- DIGITAL OUTPUT CAN BE MADE TTL COMPATIBLE WITH PROPER ON-CHIP MOS OUTPUT BUFFER
- ON-CHIP POWER CONSUMPTION IS MINIMAL; TOTAL SYSTEM POWER IS DOMINATED BY CLOCK DRIVER POWER

FIGURE 4

world that will generally surround it. It is very simple for the input to be TTL compatible because the swing on the input side of the device is small, typically a few volts or less. The output can be made TTL compatible provided on chip buffer circuits are implemented. The general trend in all CCD devices these days is to include on chip as much of the extra circuitry as needed for the interface as possible. This includes on chip output buffers, clock drivers, and clock derivation circuits. It should be remembered (as summarized by the last point on the fourth figure) that the on chip power, exclusive of the clock drivers, is really minimal. It consists entirely of the power required to shift the charge packets from one position to another; the majority of the power is dissipated in the clock driver circuits. The ratio between the clock driver power and the other power required by the chip is on the order of 30 to 1.

In order to delve more deeply into the digital characteristics of these devices let us begin with a very brief look at some of the analog characteristics. Figure 5 indicates a plan view for a typical two phase shift register: a input circuit, followed by the shift register itself and finally winding up with an output circuit. Two clock phases are required to drive it, and a total time delay is realized which depends upon the clock frequency and the number of shifts. The following figure shows the amplitude attenuation characteristics realized with this circuit. The amplitude attenuation is a function of the frequency of interest relative to the clock frequency and a function of the $N\varepsilon$ product. Such characteristics are one of the things that are avoided in the use of these devices as digital elements. The following figure indicates one other feature of analog operation that we wish to eliminate and that is the phase characteristics. Phase deviation again is a function of the $N\varepsilon$ product and the signal frequency of interest relative to the carrier frequency as shown in the figure. By going to digital operation we would not have to concern ourselves with either of these kinds of effects. This is one of the advantages of the digital operation.

Figure 8 indicates the range or achievable values for the CCD technology as it is now generally practiced. The limits shown on the figure are somewhat arbitrary and really depend on the ability of each individual semi-conductor house to produce the devices. But as a generalization the figure is still valid. Time delays on the order of a second are easily achievable and in fact numbers several times this have been claimed by various workers in the field. Clock frequencies on the order of a few MHz are not at all unusual. In between these values is another limit which is really determined by the application requirement and device characteristics. This limit is set by the particular design and the implementation being

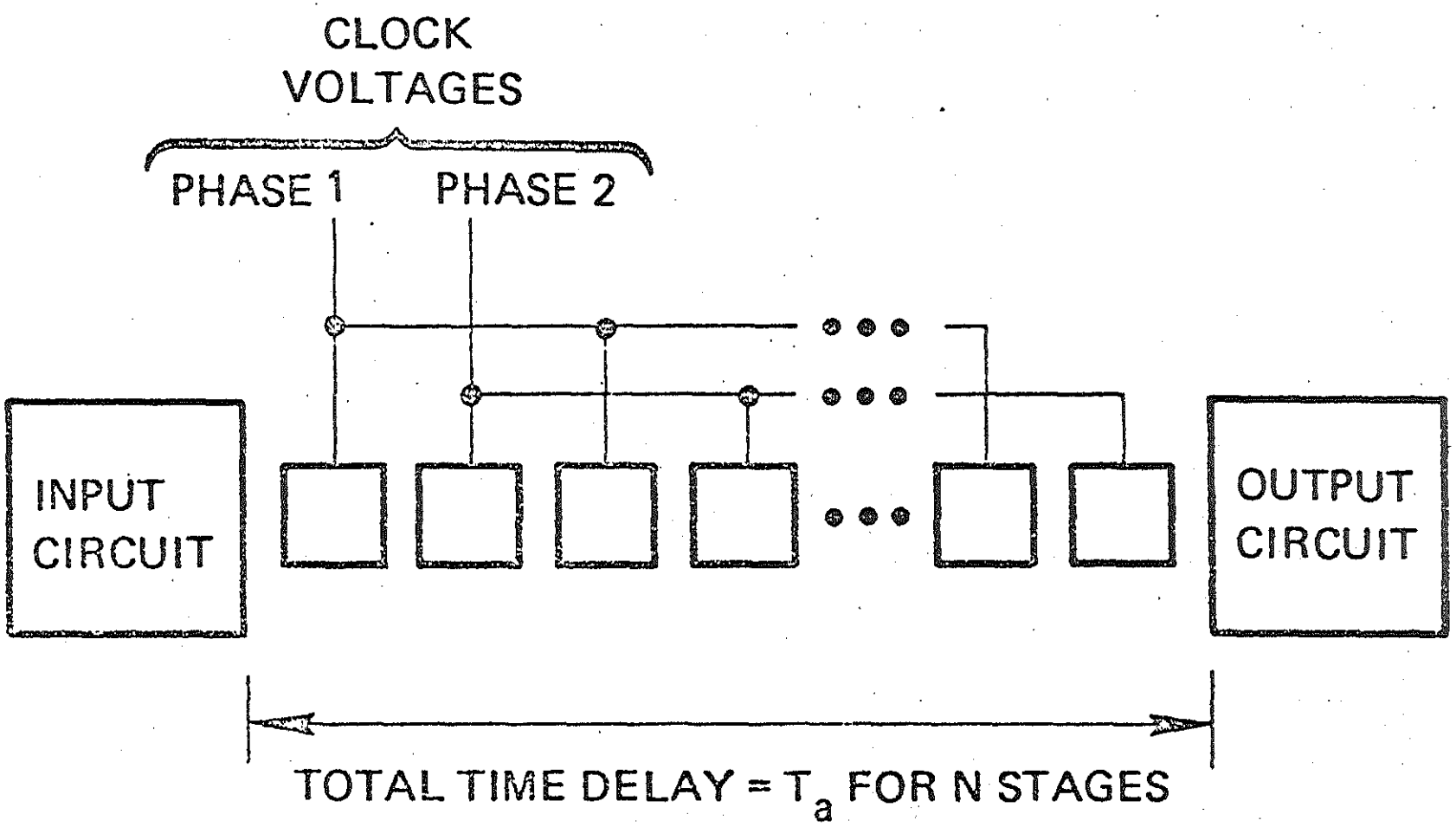


Figure 2. Basic Two-Phase Shift Register Organization (Plan View)

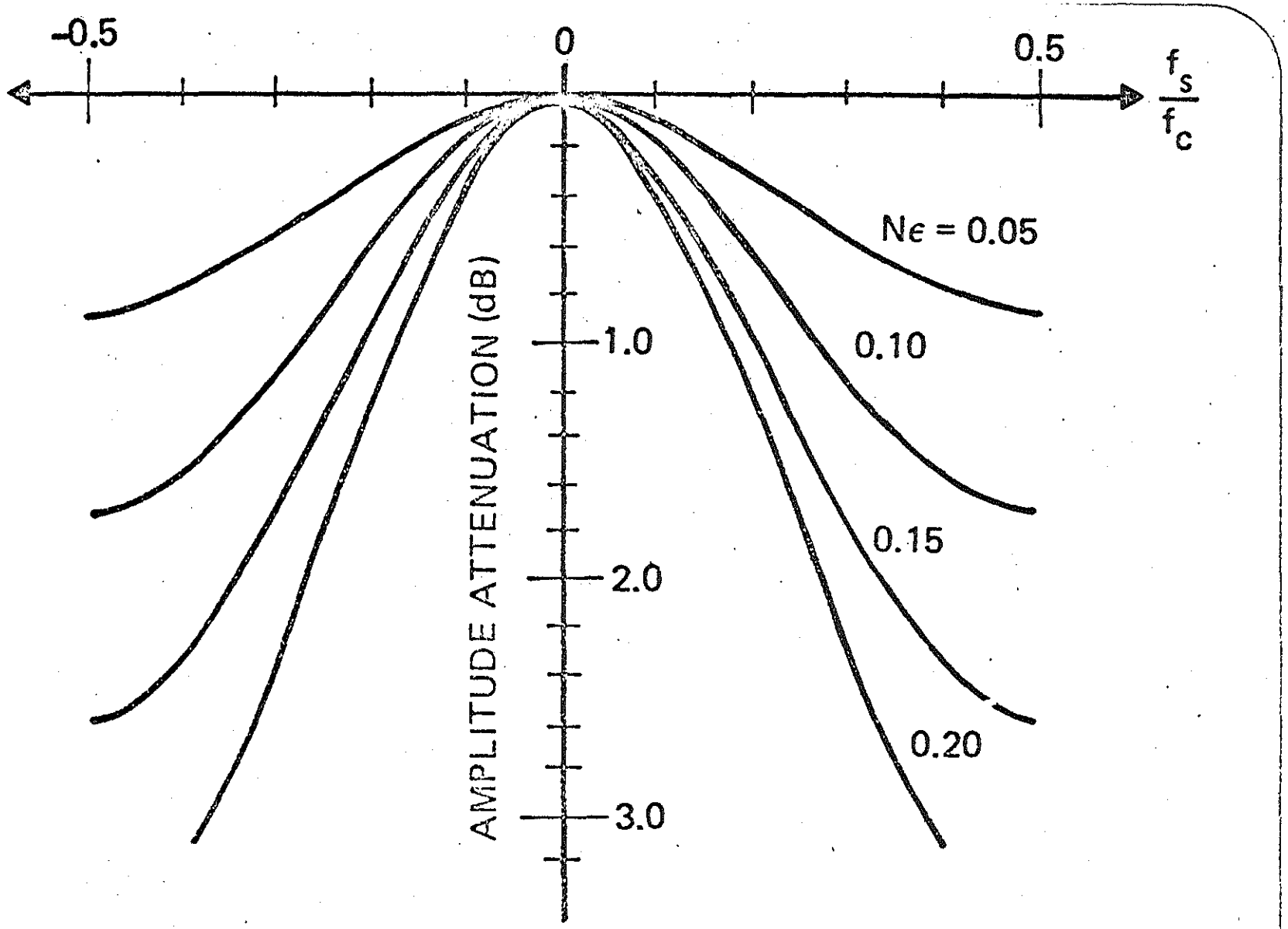


Figure 3. CCD Amplitude Characteristics as a Function of Frequency with the NE Product as a Parameter

FIGURE 6

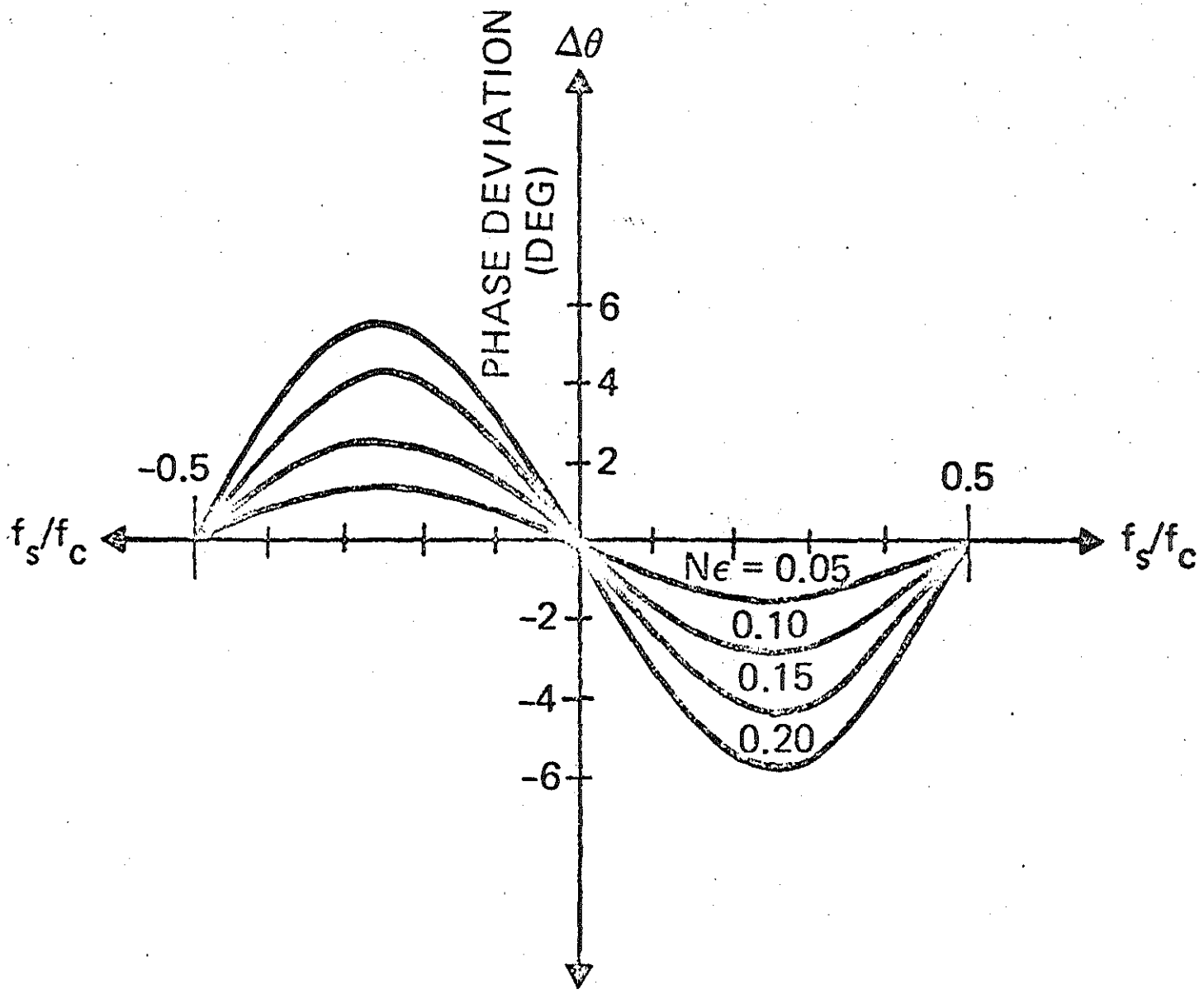


Figure 4. CCD Phase Characteristics as a Function of Frequency with the NE Product as a Parameter

FIGURE 7

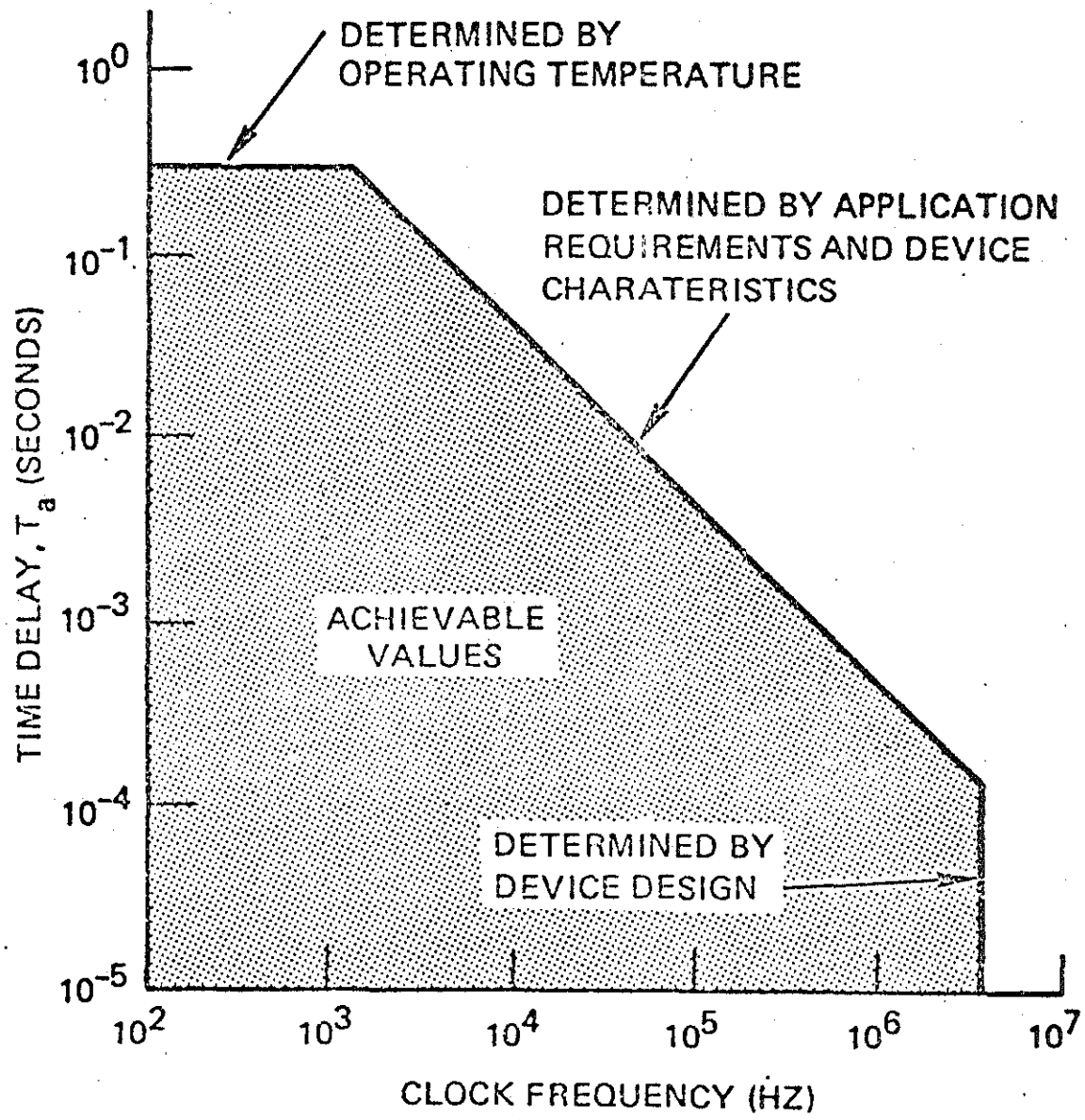


Figure 5. Typical Range of Achievable Time Delay and Clock Frequency Values

FIGURE 8

used. The central region of this figure represents all values of frequency and time delays that can be achieved with the more or less standard technology being practiced today. Certainly there are very important exceptions to this rule and values quite beyond the limits of this chart have been reported.

Figure 9 is a canonical representation of a tapped delay line. This kind of structure has been realized many many times in analog CCD. We have a signal that is shifted in from the left and delayed a number of times as it goes down the shift register. At each of these delays a sample of the signal is taken and multiplied by a waiting factor indicated as a lower case "h" on Figure 9; these samples are then summed and that sum represents the output signal. While CCD's in the analog domain represent an excellent opportunity to perform this function, it can just as easily be done in the digital domain and in fact there are some advantages in doing so. The primary advantage stems from the accuracy with which we can represent the various "h's" (multiplying coefficients) in the figure. In the analog domain there seems to be a fundamental limit in the accuracy and repeatability of the multiplying coefficients; in the digital domain we are limited only by the number of bits we chose to use in our multiplier. We will have more to say about such filter implementations as we go through the lecture.

In order to examine some of the implementations we need to look at a few specific device structures. Figure 10 represents one way to test for the presence or absence of charge in a shift register. This cross sectional view represents a single shift register gate with a potential minimum under it and a floating gate interposed between the two. It is well known that the potential of the floating gate is proportional to the difference between the gate voltage and the surface potential as indicated on the figure. It is also possible to determine the amount of charge under a given gate by measuring the current that flows in the gate line, as is also indicated on the figure. We will need both of these concepts for several items we will discuss.

If we look now at Figure 11 we see a fairly popular method for tapping and weighting an analog device. There is a shift register progressing from left to right across the top of the figure and an MOS FET attached to one gate in the register. The potential of the charge sensing device sets the potential on the MOS FET gate and this is then used to regulate the drain current. The product of this drain current and the source resistance labeled R_s gives us an analog weighted tap. Again the difficulties with this approach is the accuracy and repeatability of these

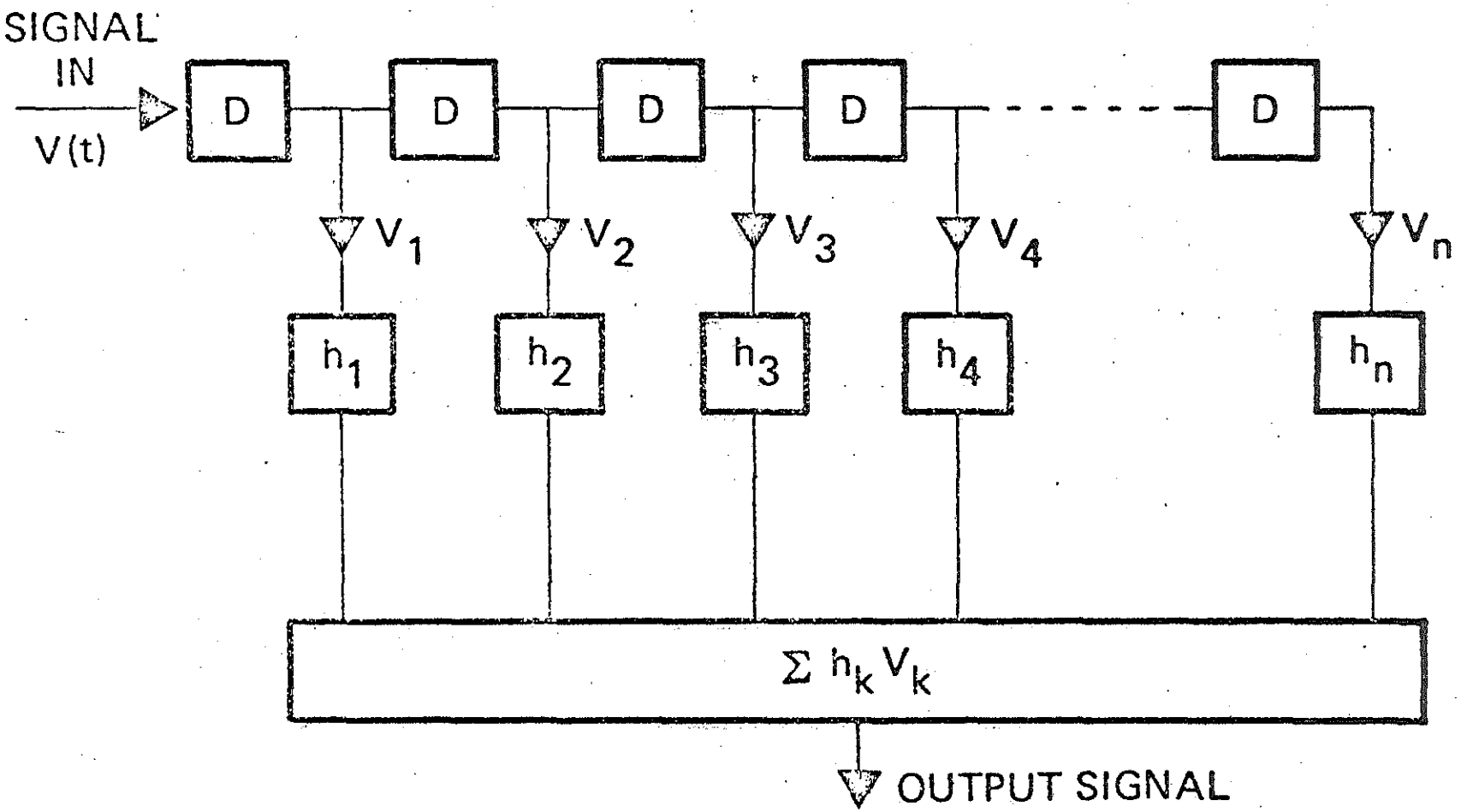


Figure 6. Tapped Delay Line Filter

FIGURE 9

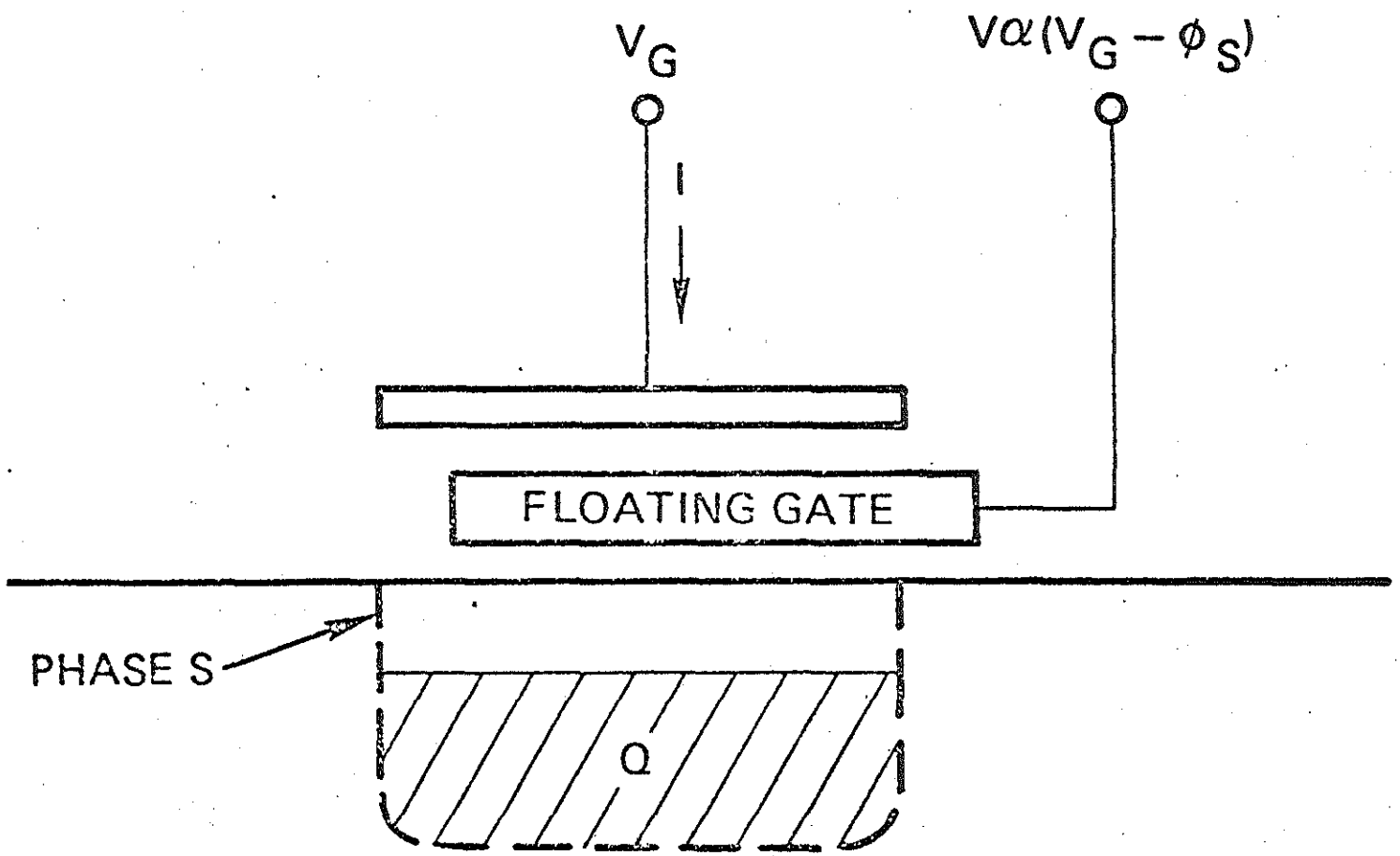


Figure 7. Nondestructive Charge Detection Schemes

FIGURE 10

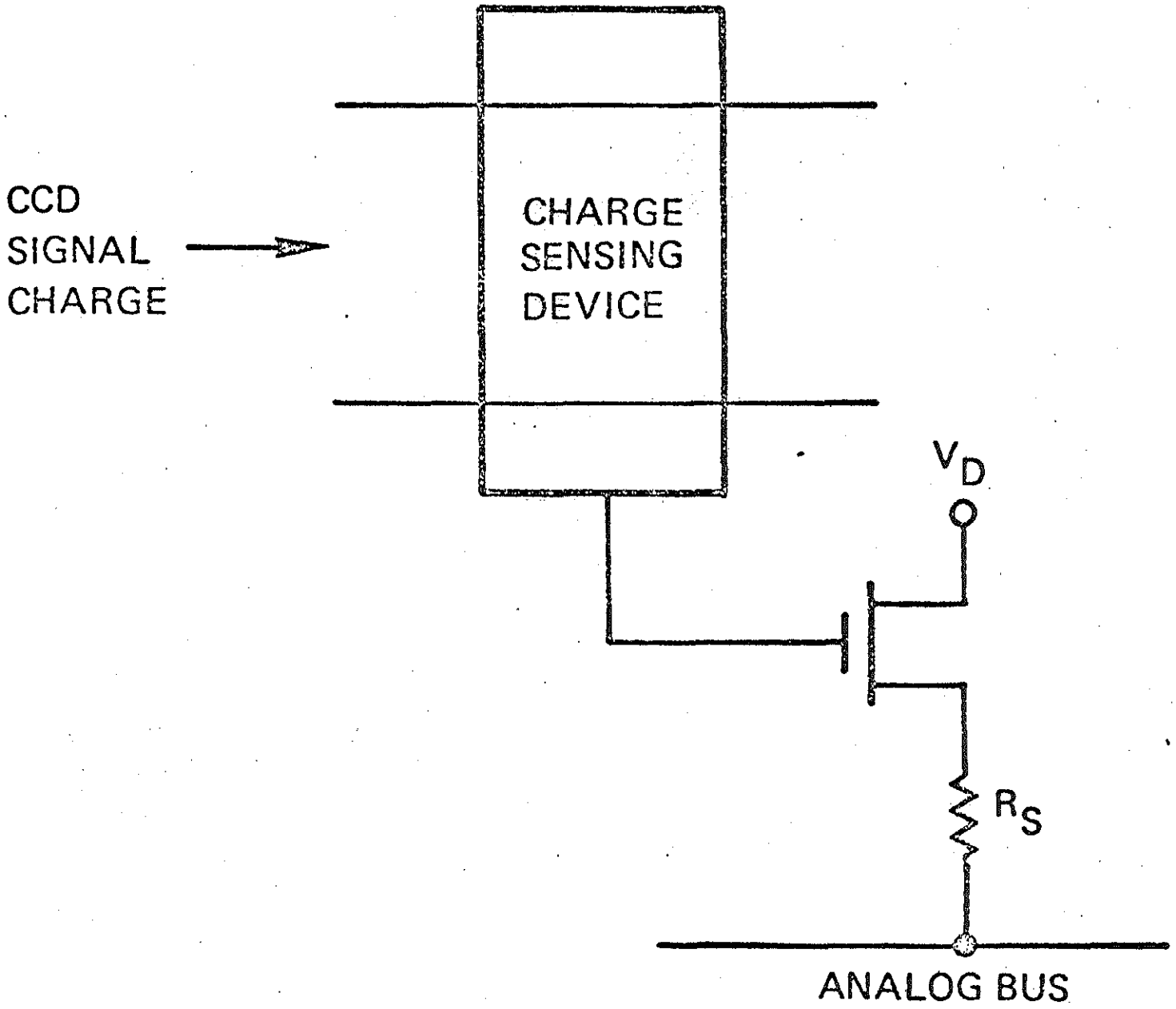


Figure 9. Analog Tapping and Weighting

FIGURE 11

taps. One way to improve the situation is to not go for arbitrary tap values but to assign simple values such as 0's or 1's to each tap.

This can be accomplished by assigning a bus line to connect to all gates which should contain a packet of charge under them (1's bus) and assigning a second bus line to connect to all gates which should not contain a packet of charge under them (0's bus). Figure 12 illustrates this process.

The operation of this shift register which now becomes a matched filter is as follows. The clock lines are all pre-charged to a given value through the circuit indicated just to the left of the differential amplifier. Then the CCD data stream is shifted under the gates. If we examine in detail the charging currents that flow in each of the gates, we will find that the output voltage from the differential amplifier will be a maximum when the input sequence is 110. That is, the two gates on the lefr hand side are filled with charge and the right most gate is empty. Under any other set of conditions for the 3-bit input sequence, the voltage will be less than this maximum. In this way we have constructed a filter that is matched to a particular sequence and that sequence is determined by weighting functions of simple 1's and 0's.

The major disadvantage of this approach is that once the chip is made and the buss lines have been interconnected as indicated, the sequence to which we are matching is fixed. In many systems applications we would by far prefer to be able to alter that sequence at will. This can be achieved by a scheme such was indicated in Figure 13. Here we see that we have in fact three shift registers being indicated. There is a reference shift register on top, the data bit stream in the center, and the inverse of the reference bit stream on the bottom. The figure shows only one section of the three shift registers in plan view. At each bit location we test the reference bit stream and its inverse to determine whether a 1 or a 0 weight is indicated. By using the reference and its inverse and by driving two MOS FET's with these signals we can definitively connect the data bit stream gate to either the 1's buss line or the 0's buss line. This operation, of course, is occuring at every bit location in the shift register and we are only viewing it here at one location.

By this technique we can shift in an arbitrary reference stream (and its inverse), and try to match the incoming signal to this reference. Note that if we construct a shift register properly, we are free to shift the reference stream and the data stream in the same direction, opposite directions, or even hold one

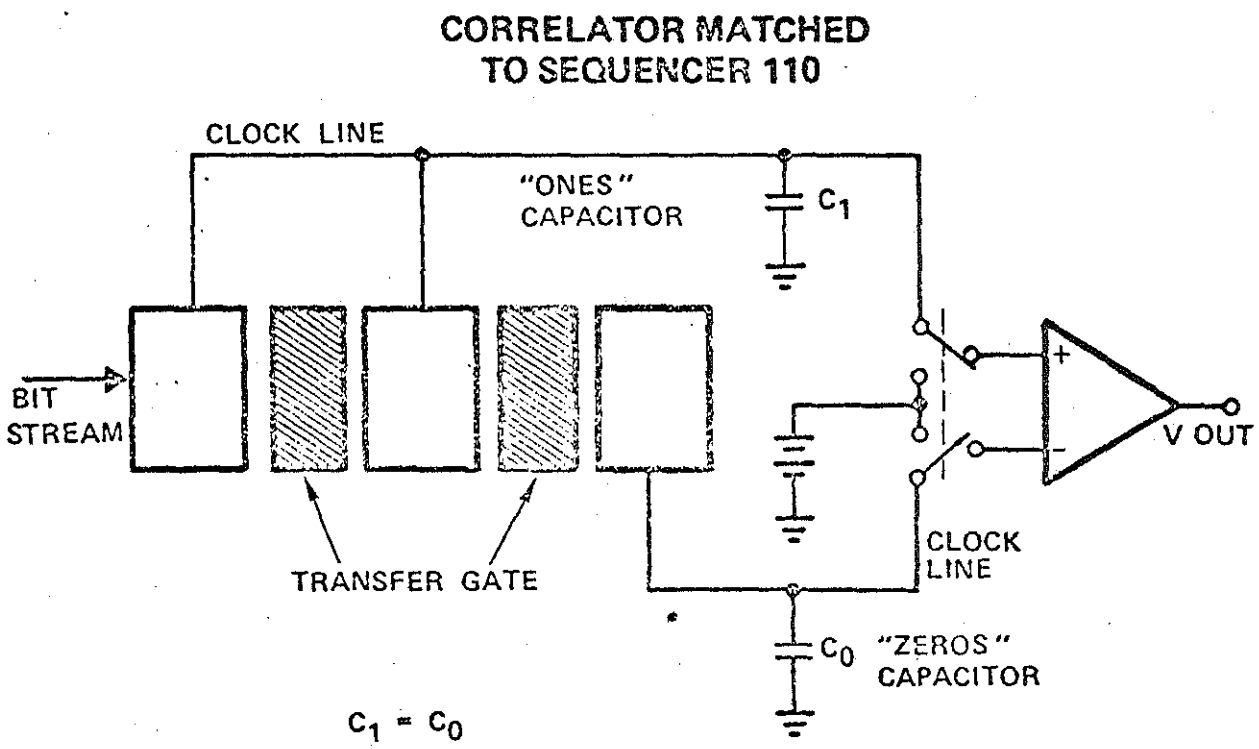

CORRELATOR MATCHED
TO SEQUENCER 110
CLOCK LINE
"ONES"
CAPACITOR
C_1
BIT
STREAM
TRANSFER GATE
+
-
V OUT
CLOCK
LINE
C_0
"ZEROS"
CAPACITOR
$C_1 = C_0$

FIGURE 12

PROGRAMMABLE CORRELATION OF A SINGLE BIT BY THE USE OF TWO AUXILIARY BIT LOCATIONS

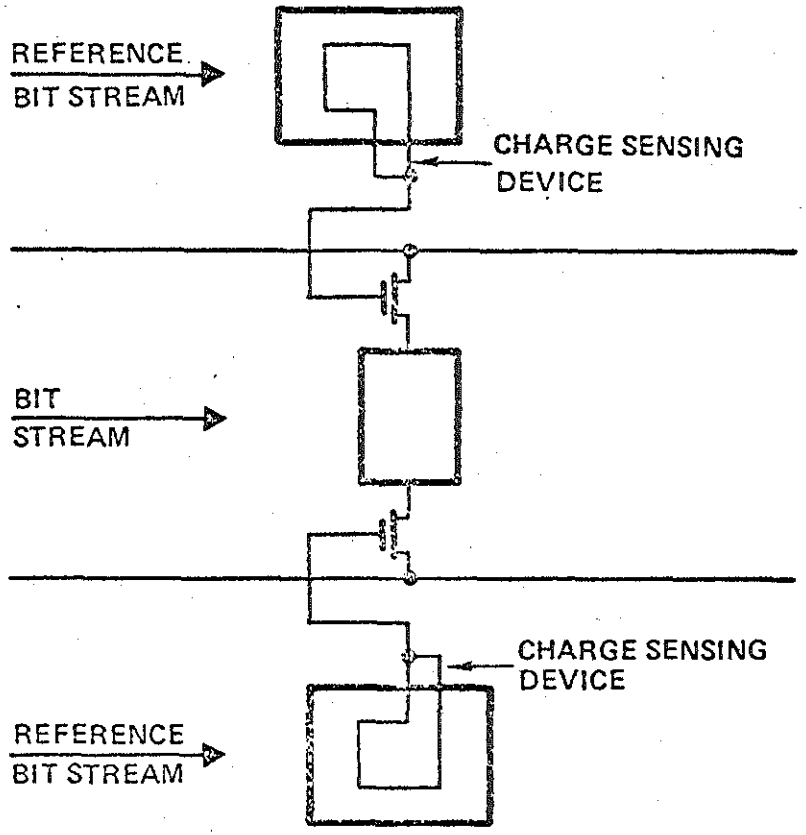


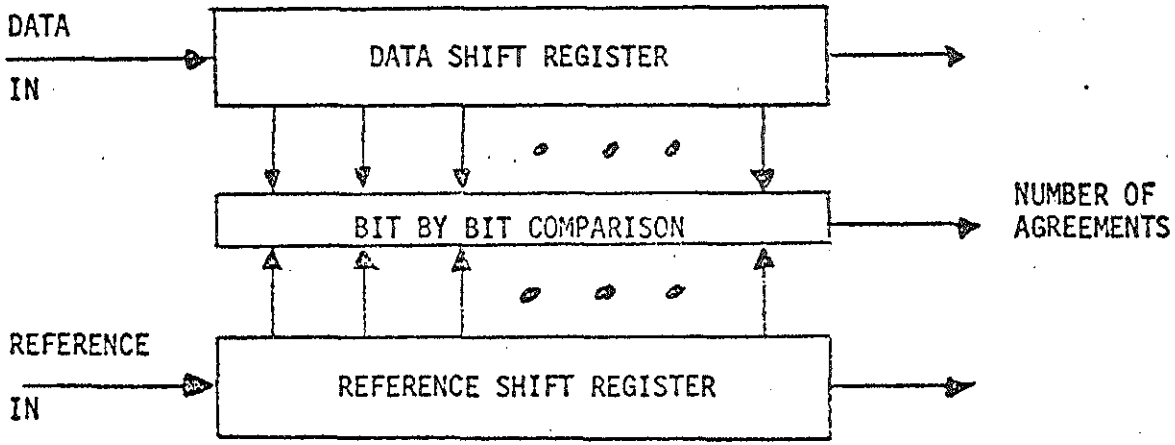
FIGURE 13

stationary while the other one marches past. This gives us a programmable filter and largely removes the objections of the scheme discussed in Figure 12.

In order to take this technique and to derive some more information from it we need to assembly such shift registers in quite long lengths. This is indicated in Figure 14 where we show a reference shift register and a data shift register and schematically indicate a bit by bit comparison. The ideal output from such a correlator is simply the number of agreements between the reference register and the data register minus the number of disagreements. The details of how we achieve this counting of agreements and how we present it to the output represents the real significance of the CCD technology in this kind of application. There is one other item of interest to be noted on the diagram of this figure: the data in the two shift registers is not destroyed or altered by the correlation process shown. This means that we are free to use that data in other techniques or other processing sequences.

The previous technique can be termed bit preserving; Figure 15 shows a bit destructive approach. In this technique, two shift registers have their charge packets physically combined to produce a third shift register. In this way, the original data is destroyed and we produce a new data stream that is a combination of the other two. Both approaches are useful and find applications in digital charge coupled device techniques.

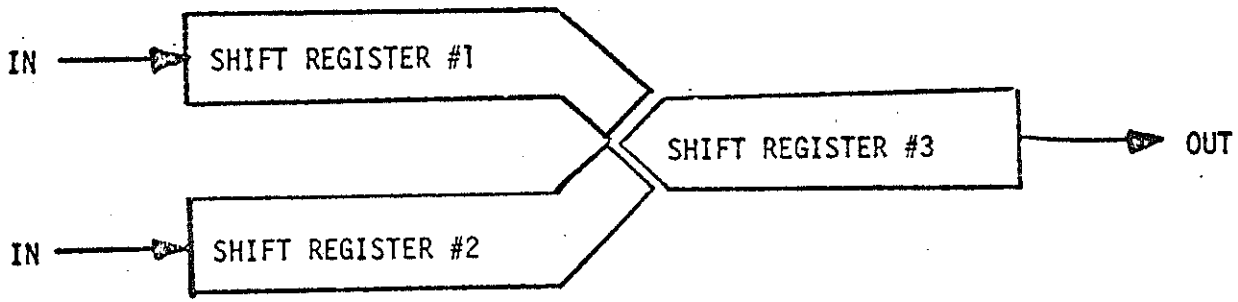
Let us now begin to examine some of the details of how we do logic and arithmetic functions with digital charge coupled devices. Figure 16 shows a very simple plan view schematic of the beginning of a CCD shift register. On the left side of the figure we see a source of carriers for the shift register (simply a biased p-n junction). Gate C is indicated as the first gate in a CCD shift register that extends off to the right of the figure. Between this first gate and the source of carriers we have interposed two transfer gates. Now clearly to get carriers from the source to the first gate we must have both transfer gates in an "on" condition. That means that both control variable A and the control variable B must simultaneously be on. For reasons we will discuss later, the variables A and B are themselves normally derived from other shift register signals and therefore have an inherent inversion in them. This means that the function represented by Figure 16 is really NOT A and NOT B which of course produces the NOR gate as indicated in the figure title.



CORRELATION TECHNIQUE

FIGURE 14

BIT-DESTRUCTIVE LOGIC



- CHARGE PACKETS IN INPUT REGISTERS ARE PHYSICALLY COMBINED TO FORM A CHARGE PACKET FOR OUTPUT REGISTER.

FIGURE 15

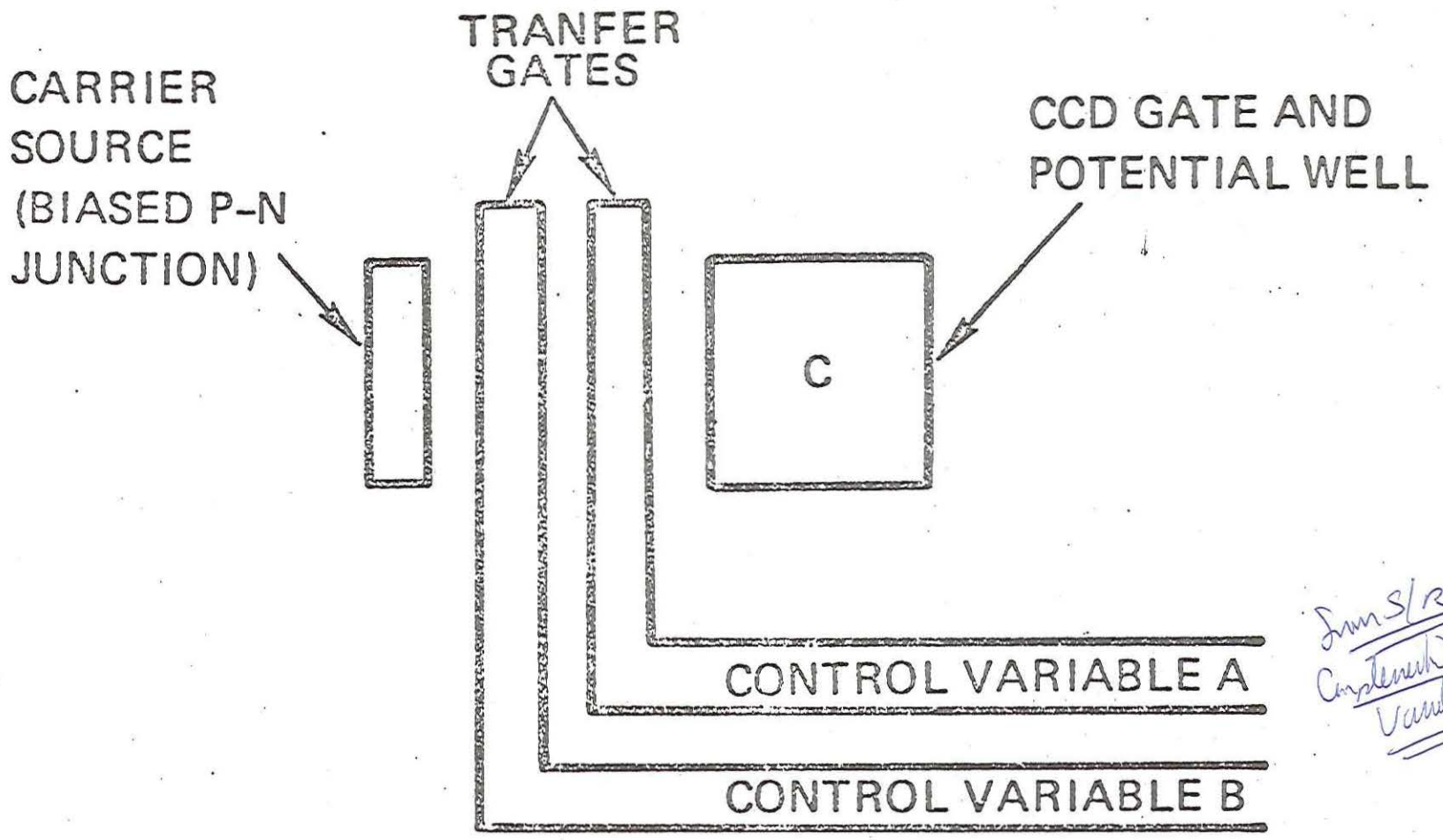


Figure 10. NOR Gate

FIGURE 16

Now by duality, we can change this concept into a NAND gate. Figure 17 shows the plan view for that structure. We have a source of carriers which has two parallel paths to the first gate in the CCD structure. Under these conditions either control variable A or control variable B can allow carriers to transfer from the source to the CCD gate. Once again these variables are generally derived from other shift registers and have the inherent inversion; therefore the function we have here is NOT A OR NOT B which translates into a NAND gate.

If we return for a moment to our bit destructive logic functions, we can examine another kind of structure. The diagram on Figure 18 shows two CCD shift registers, one approaching from the left and one coming from the top of the figure. They meet in a common area and there are two exists, one to the right and one to the bottom. Now let us examine several possibilities for the combinations of inputs A and B. We must remark at the outset that all the storage areas are of equal capacity; that means that they can all store the same amount of charge.

If we assume that there is no charge under A or under B than on the single simultaneous clock pulse that combines and dumps their outputs into the common gate no charge will be transferred there. That means that when we later investigate the contents of gate C we will find that it is empty.

On the other hand if either A or B have charge under them than that charge will transfer to the common gate and subsequently transfer to gate C and we will find it there when we look for it.

Let us look at the most interesting case; both A and B have charge under them. Under this condition, when the charge is transferred to the common gate there is more available than the common gate itself can hold. At that time the barrier comes into play and charge flows over that barrier and fills the area under gate D. This means that gate D will contain charge only under the circumstances that both A and B contain charge. If we then look at C and D as Boolean variables, we can write the two Boolean equations indicated on the figure. C is equal to A OR B while D is equal to A AND D. Thus we have achieved in one simple structure an AND gate and also an OR gate, both using charge destructive techniques.

This concept can be extended to provide an even more useful function, the exclusive OR gate. Such an arrangement is indicated in Figure 19. This figure is really an extention of Figure 18. We have added just a few elements. In

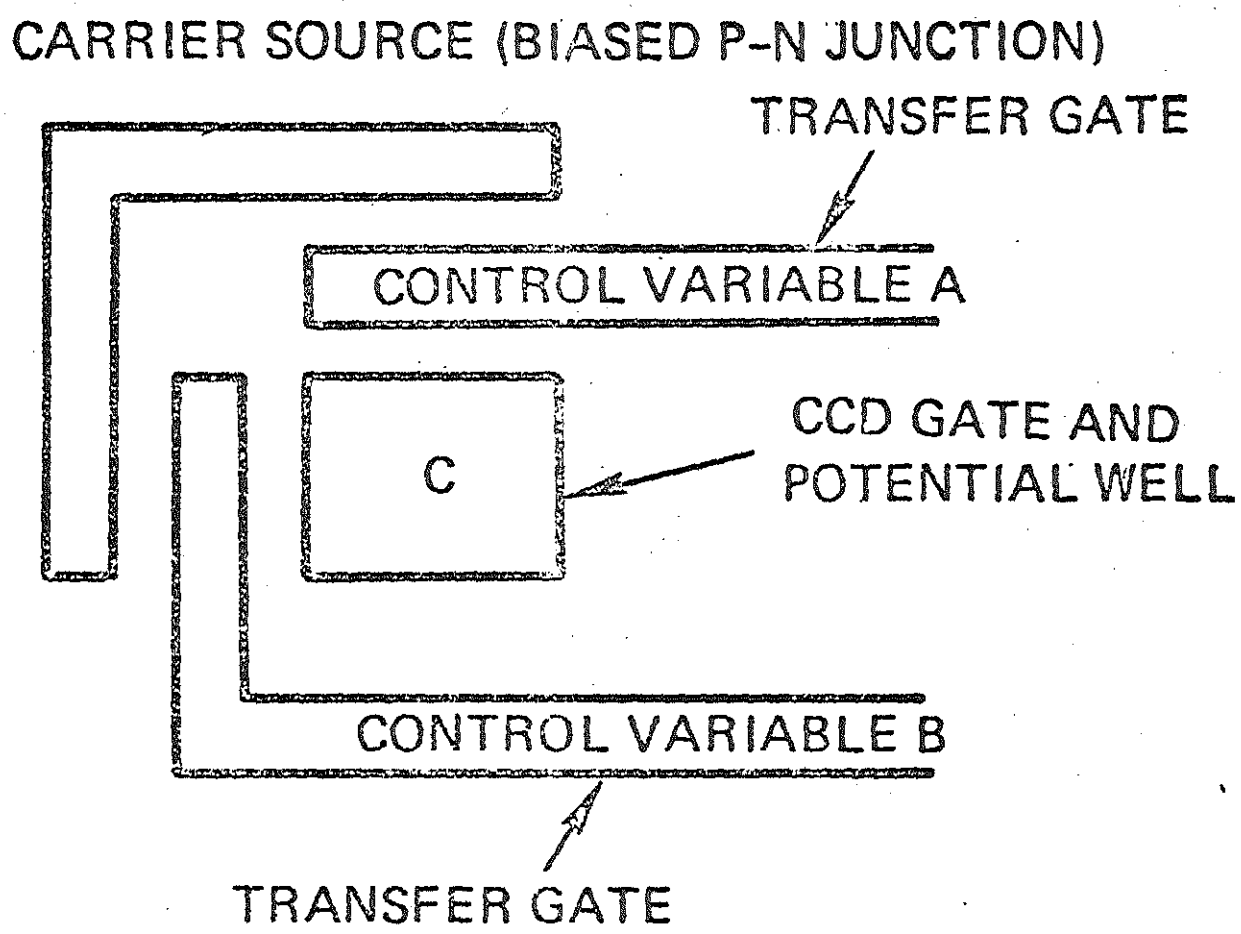


Figure 11. NAND Gate

FIGURE 17

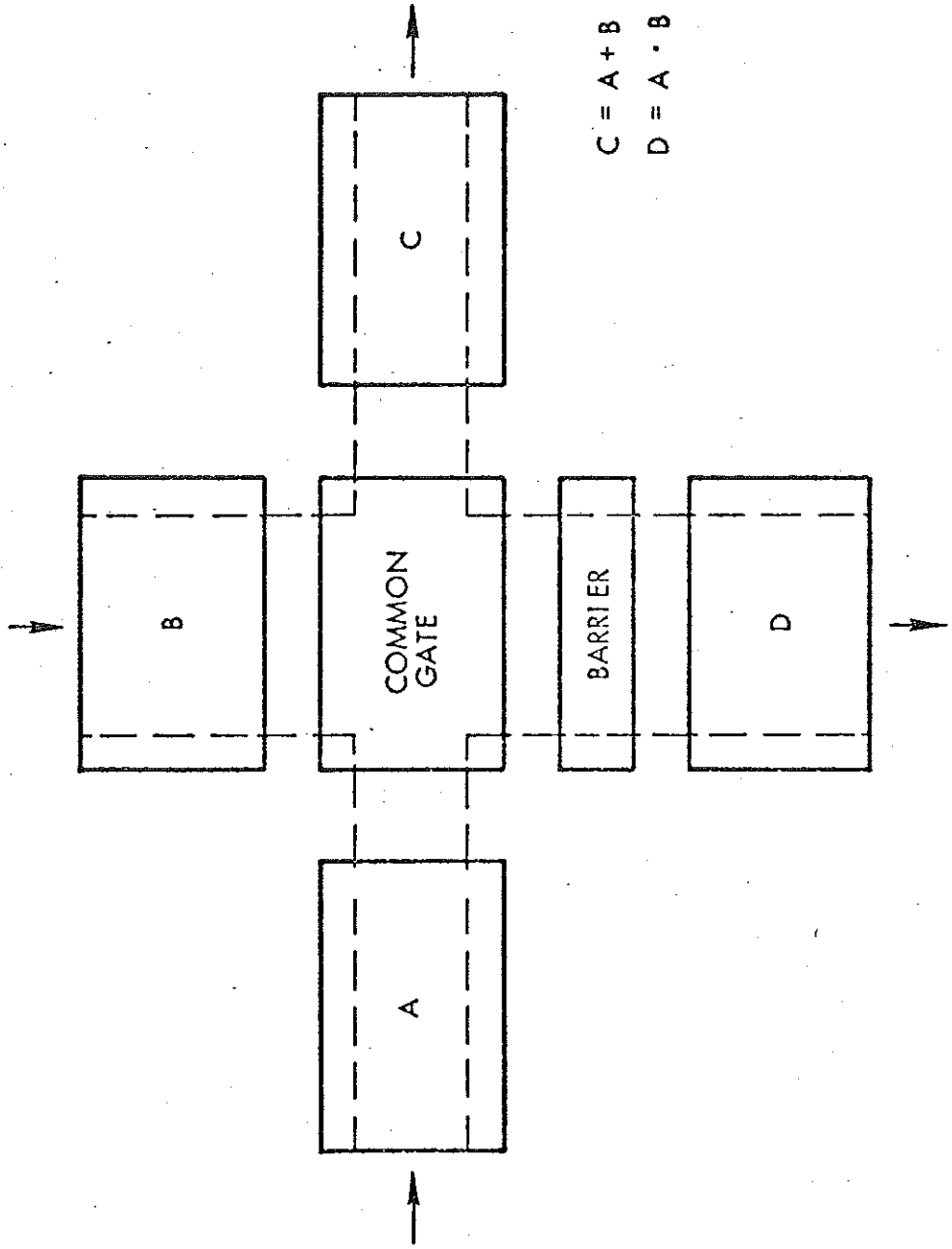


FIGURE 18

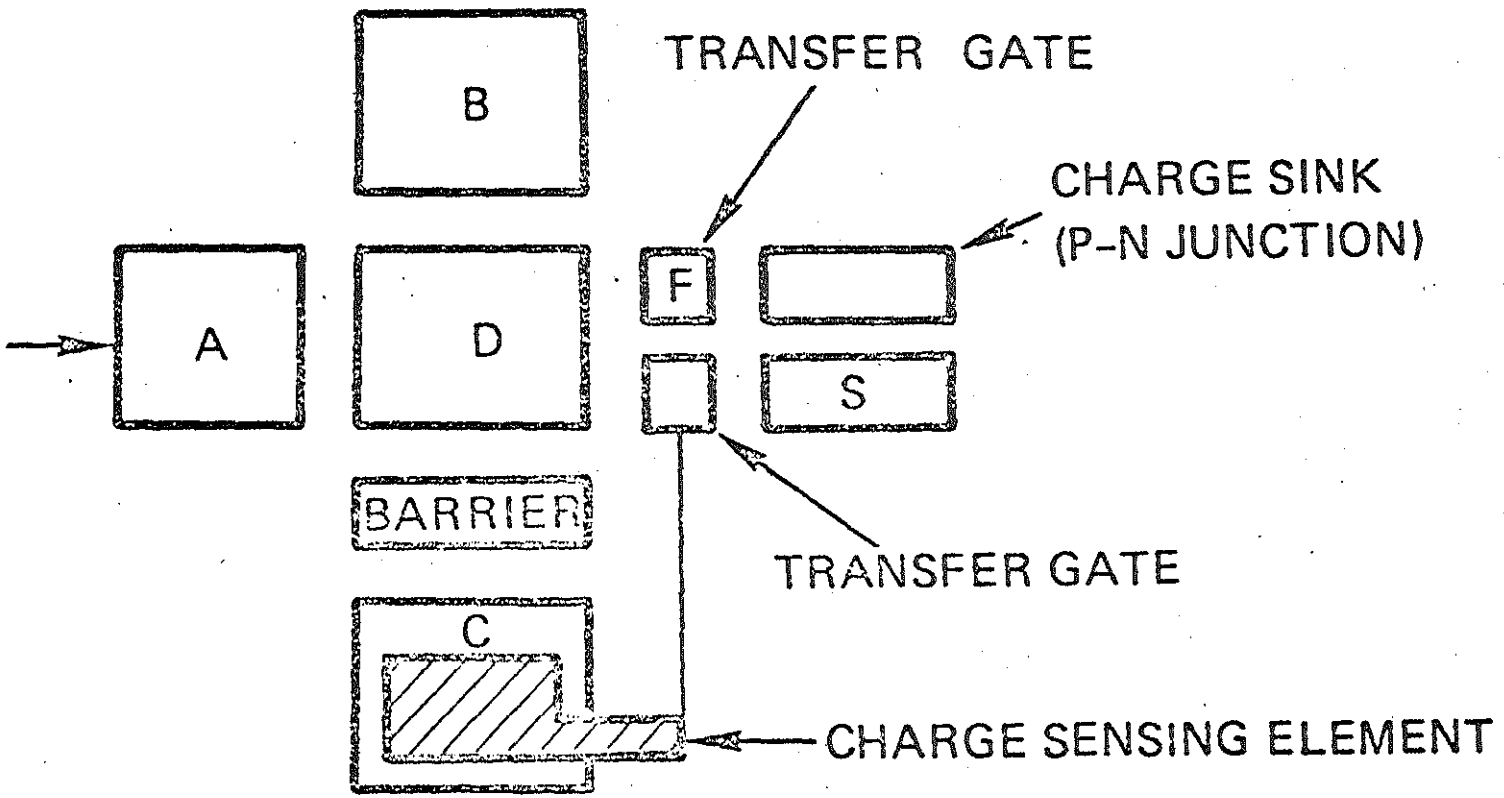


Figure 13. Exclusive OR Gate

FIGURE 19

particular we have taken the gate marked C and added a charge sensing element to it and used that charge sensing element to control the charge flow from what had been the common gate, now labeled gate D. The device operates in the following ways: under the conditions that both A and B contain charge, than C will also receive a packet of charge. This will be detected by the charge sensing element which then turns the transfer gate to the OFF potential thus preventing charge from flowing out from under D to the gate marked S. Under this condition we have created an exclusive OR gate for the gate marked S will have charge when either A or B has charge but not when both contain charge. Of course this function also performs as a half adder circuit, accounting for the labels of S and C meaning SUM and CARRY.

In order to extend this concept to the complete full adder function, we need to make several additions to the circuits shown in Figure 19. One obvious addition is that of a third input. Figure 20 shows the other circuit components that are needed. Basically, we need to have a storage area for the condition when all three inputs (labeled A, B, and G in Figure 20) contain charge. That is the function of the gate labeled I.

When A, B, and G simultaneously attempt to dump their charge under D, we have a situation where the charge first flows to gate C, and then from C to gate I. If we examine all the charge packet location possibilities for this circuit, we find that indeed the sum circuit and carry are appropriate for a complete full adder operation. Note that in Figures 19 and 20 we had an additional sink circuit indicated next to the gate marked D. This is necessary in order to clear out any charge that may remain behind under the gate D so that the next operation can proceed.

The first full adder circuit implemented is shown in Figure 21 and follows closely the layout of Figure 20. The full adder is in fact a very important integral part of all the arithmetic functions that are performed with digital logic circuits. Accordingly it is quite instructive to examine the full adder implementation in a number of technologies and to compare them. We did this in a number of publications that have appeared in the literature. One such publication was done at the 1976 International Electron Device Meeting in Washington. Another such publication will appear in the 1977 Journal of Solid State Circuits Special on Memory and Logic. Figure 22 is taken from that latter presentation. It is a comparison of full adder circuitry and the power required for that full adder realized in a number of technologies. The details of the circuits that are used to produce these estimates are the same as shown in the 1976 IEDM reference. We

FULL ADDER

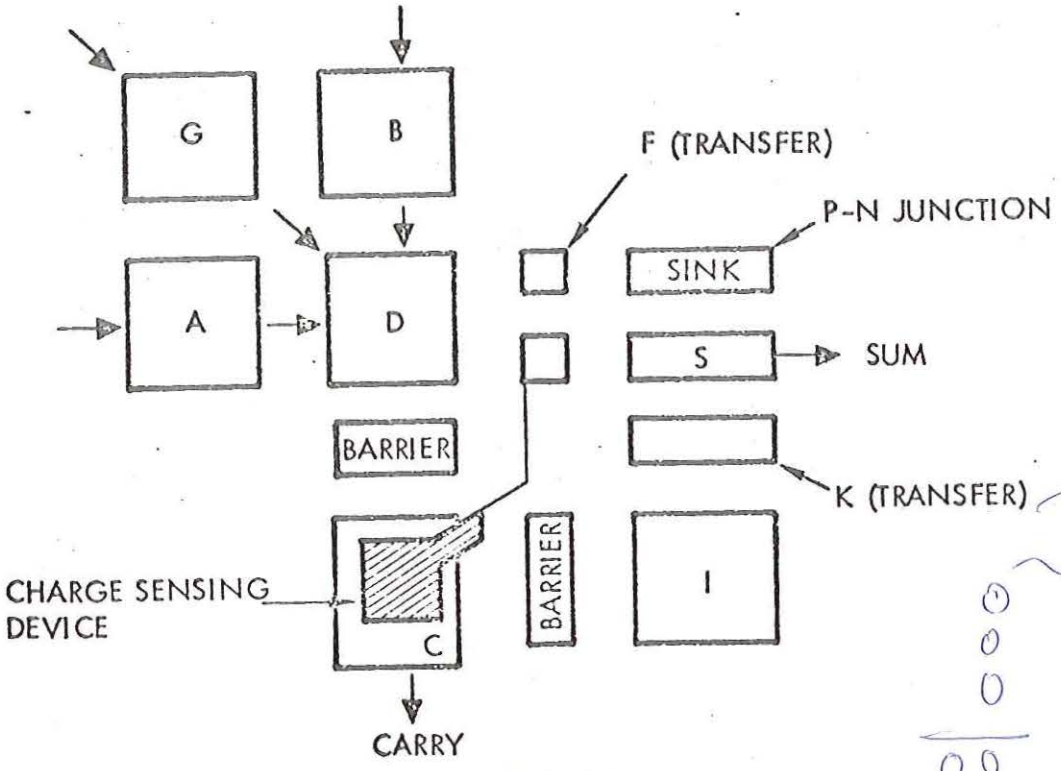
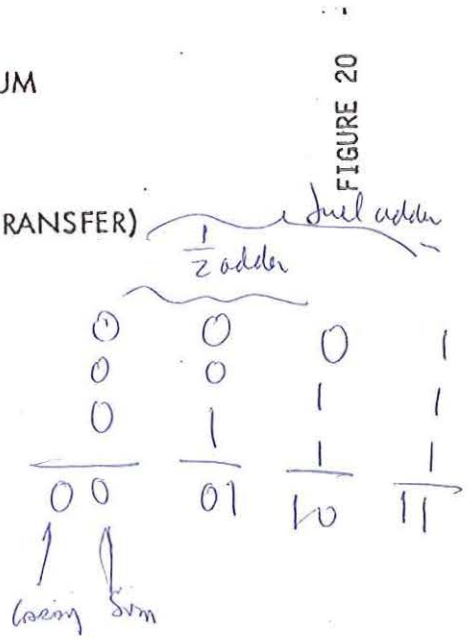


FIGURE 20



• BIT DESTRUCTIVE LOGIC

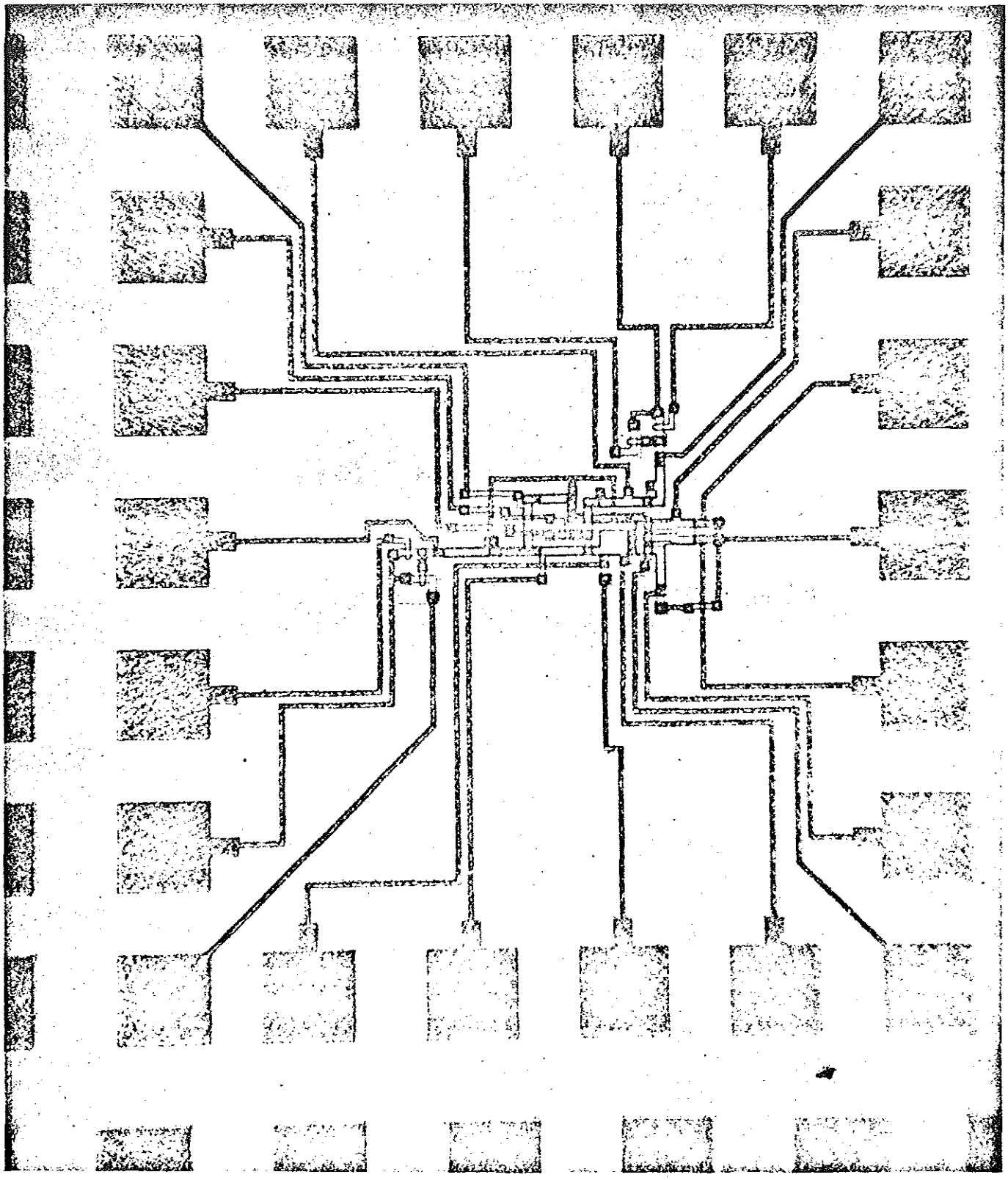


FIGURE 21

FULL ADDER POWER REQUIREMENTS AS A FUNCTION OF FREQUENCY

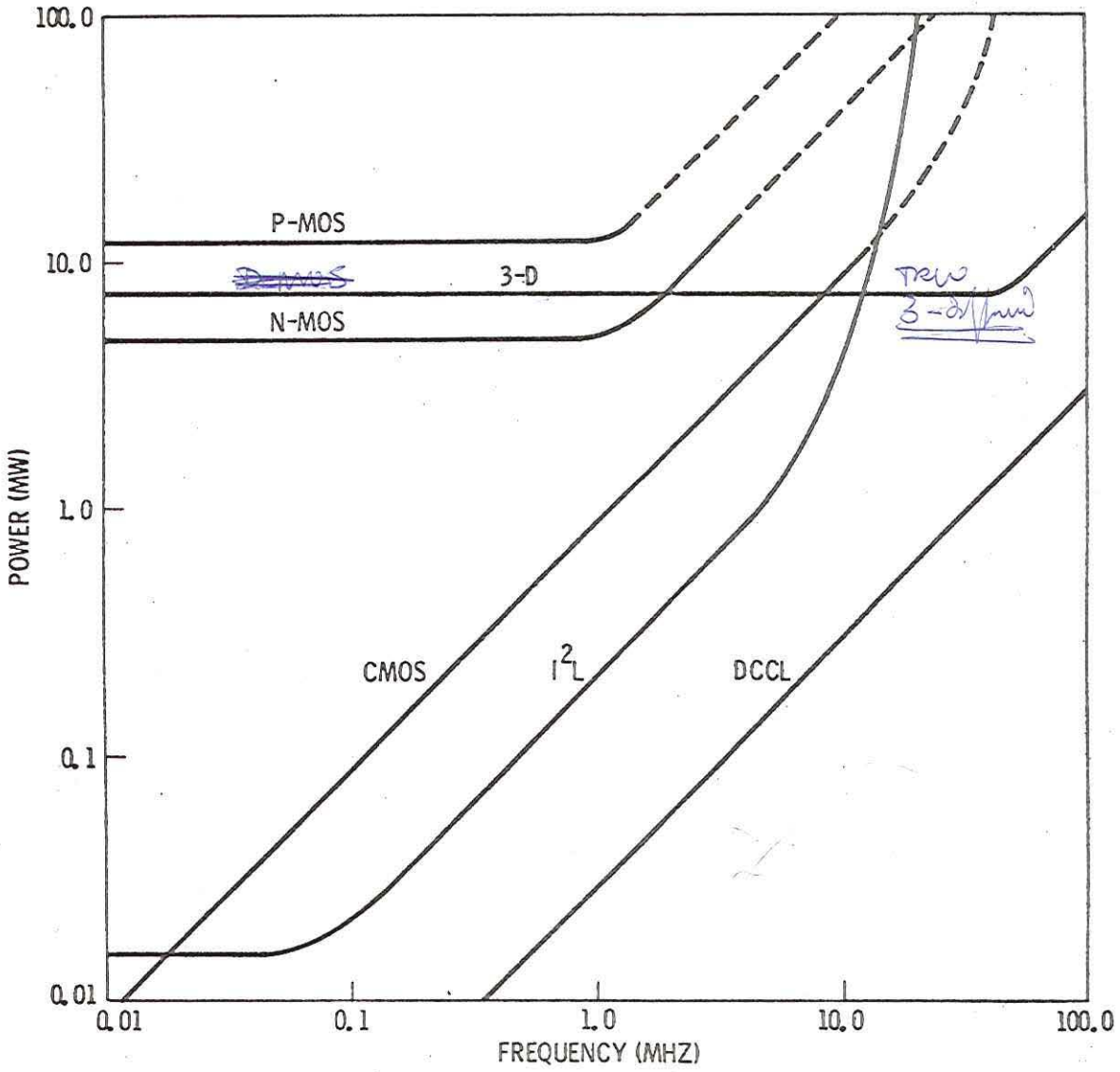


FIGURE 22

see that the digital CCD is about an order of magnitude better than its nearest competitor over any portion of the frequency range. This advantage in power is one of the most attractive features of the digital CCD. To be more specific about the kinds of functions required to realize large arrays, Figure 23 summarizes the types of cells needed for a number of adder and multiplier arrays. We see that half adders, full adders, shift registers, AND gates, and regeneration circuits are all required to realize most arrays.

The following two figures (24 and 25) summarize some of the power requirements at two different frequencies for various arrays and various technologies. It is clear that the digital CCD has a distinct advantage. Finally, Figure 26 shows the area required for the several implementations. Again, the digital CCD has a obvious advantage

Let us look in some more detail at some of these array implementations. Figure 27 schematically indicates the construction of a two-word 4-bit adder in the digital CCD technology. The way we wish to operate this adder is as follows. We would like at one instant in time to present two 4-bit words to the adder and have them clocked in with one clock pulse; then at some time in the future we would get the 5-bit answer from the addition of those 2 words at one clock pulse. In order to achieve this we need to insert delays in various paths along the addition.

The inset in the figure shows the way one would do this addition by hand. We would first add the A1 and B1 bits to get a sum and a potential carry labeled as C2. This carry then would be added to the A2 and B2 bit to produce a new sum labeled S2 and a new carry labeled CS. This process goes on until the entire addition is finished. That means then that we must delay certain bits until they are available at the time the carry will be available.

As seen in the figure, A1 and B1 can go immediately to a half adder. Note that the half adder is all that is needed here since we have only two inputs and not three. The sum and the carry are available immediately after the addition and the sum is put into an output line and delayed while the carry is shifted to the left so that it can now be added to the A2 and B2 lines. Of course A2 and B2 had to be delayed before they reached the full adder which will do their addition. Following this technique, we see that we need a series of delays on the input lines and a corresponding series of delays on the output lines. With these delays in place we can clock in all the bits on the input at one time period and get all the output bits at a single time. These two triangular shaped delays actually account

DIGITAL CCD ARRAYS MAKE MAXIMUM USE OF SHIFT REGISTERS

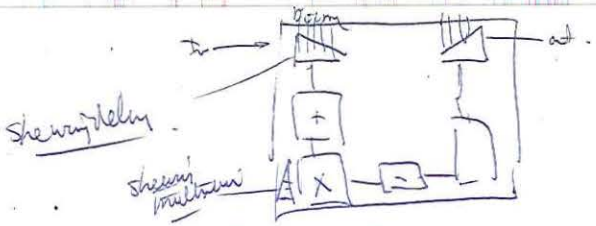
TRW

CELL COUNT FOR VARIOUS DCCL ARRAYS

CELL TYPE	16 + 16	32 + 32	8 X 8	16 X 16
REGENERATION CELLS	75	343	62	89
AND GATES	0	0	64	256
SHIFT REGISTERS	360	1488	190	1328
FULL-ADDERS	15	31	47	214
HALF-ADDERS	1	1	11	16

Based on Clock in Room

FIGURE 23



DIGITAL CCD SAVES FROM 78 TO OVER 99 PERCENT OF THE POWER AT 1 MHZ

TRW
UNIFORM AND SPACE SYSTEMS GROUP

TOTAL POWER DISSIPATION IN WATTS OF VARIOUS SIZE ARRAYS AND TECHNOLOGIES AT A CLOCK FREQUENCY OF 1 MHZ

TECHNOLOGY	16 + 16	32 + 32	8 X 8	16 X 16
DCCL	0.009	0.024	0.008	0.044
CMOS	0.582	2.3	0.820	4.1
P-MOS	2.9	13.3	2.5	15.0
N-MOS	0.531	2.3	0.559	3.1
I^2L	0.040	0.174	0.036	0.215

FIGURE 24

DIGITAL CCD SAVES FROM 85 TO 99 PERCENT OF THE POWER AT 10 MHZ

TRW

TOTAL POWER DISSIPATION IN WATTS OF VARIOUS SIZE ARRAYS AND TECHNOLOGIES AT A CLOCK FREQUENCY OF 10 MHZ

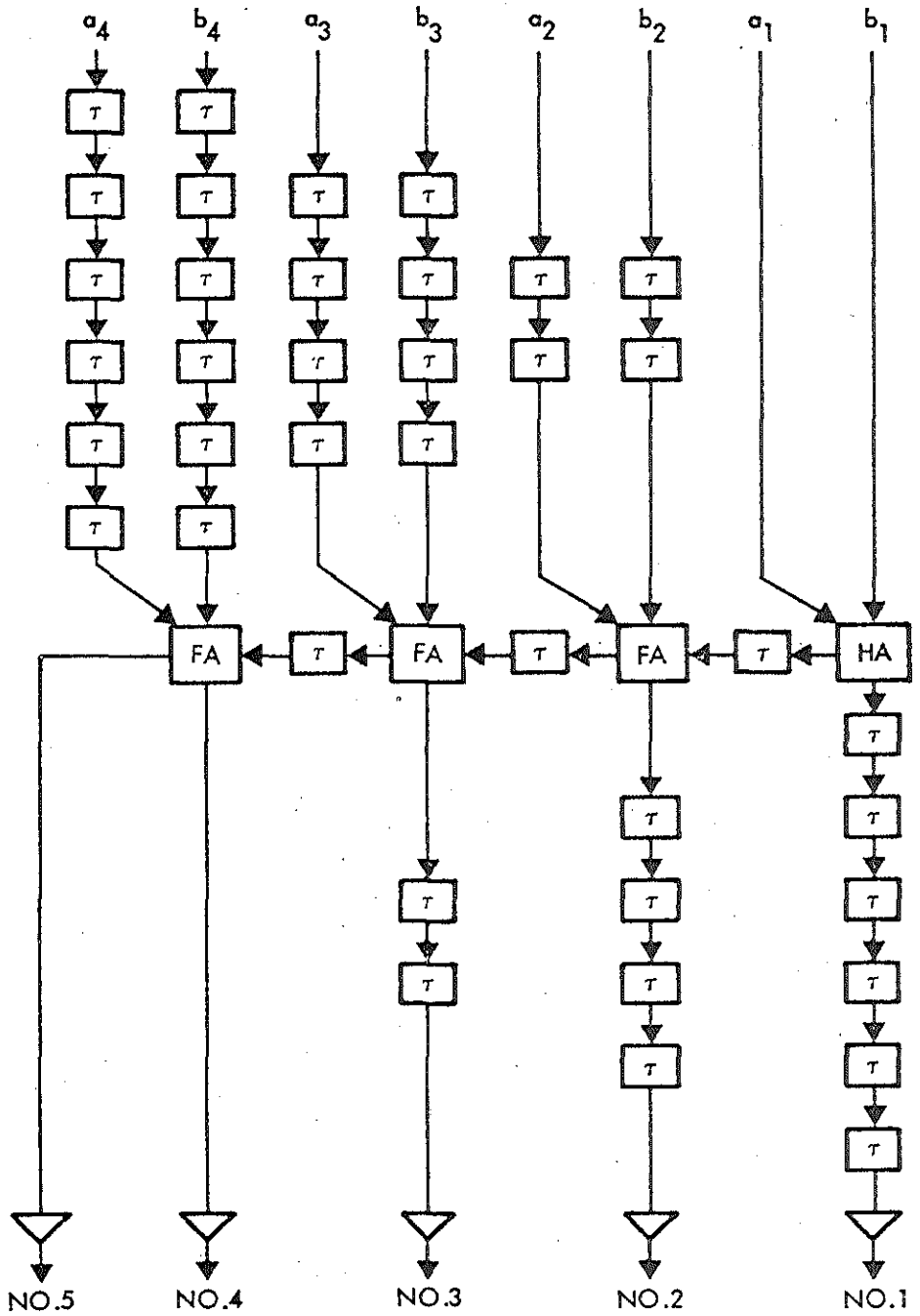
TECHNOLOGY	16 + 16	32 + 32	8 X 8	16 X 16
DCCL	0.089	0.237	0.077	0.444
CMOS	1.8	6.8	2.8	13.8
P-MOS	4.6	19.9	4.9	27.0
N-MOS	1.02	4.3	1.05	4.9
I^2L	0.596	2.7	0.544	3.2

FIGURE 25

ESTIMATES FOR THE ACTIVE AREA IN MM^2 OF VARIOUS ARITHMETIC ARRAYS CONSTRUCTED FROM DIFFERENT SEMICONDUCTOR TECHNOLOGIES

TECHNOLOGY	16 + 16	32 + 32	8 X 8	16 X 16
DCCL	2.92	8.94	3.1	28.0
P-MOS	11.3	49.2	12.2	67.7
N-MOS	7.78	34.7	7.65	44.2
CMOS	16.5	70.2	19.5	104
I^2L	14.9	64.9	26.2	137

FIGURE 26



TWO WORD, FOUR-BIT ADDER

CARRY BITS		c4	c3	c2	
A-WORD		a4	a3	a2	a1
B-WORD		b4	b3	b2	b1
SUM	c_5	S4	S3	S2	S1

FIGURE 27

for a large portion of the area needed to implement adders in this technique. We will have more to say about these delays a little bit later on.

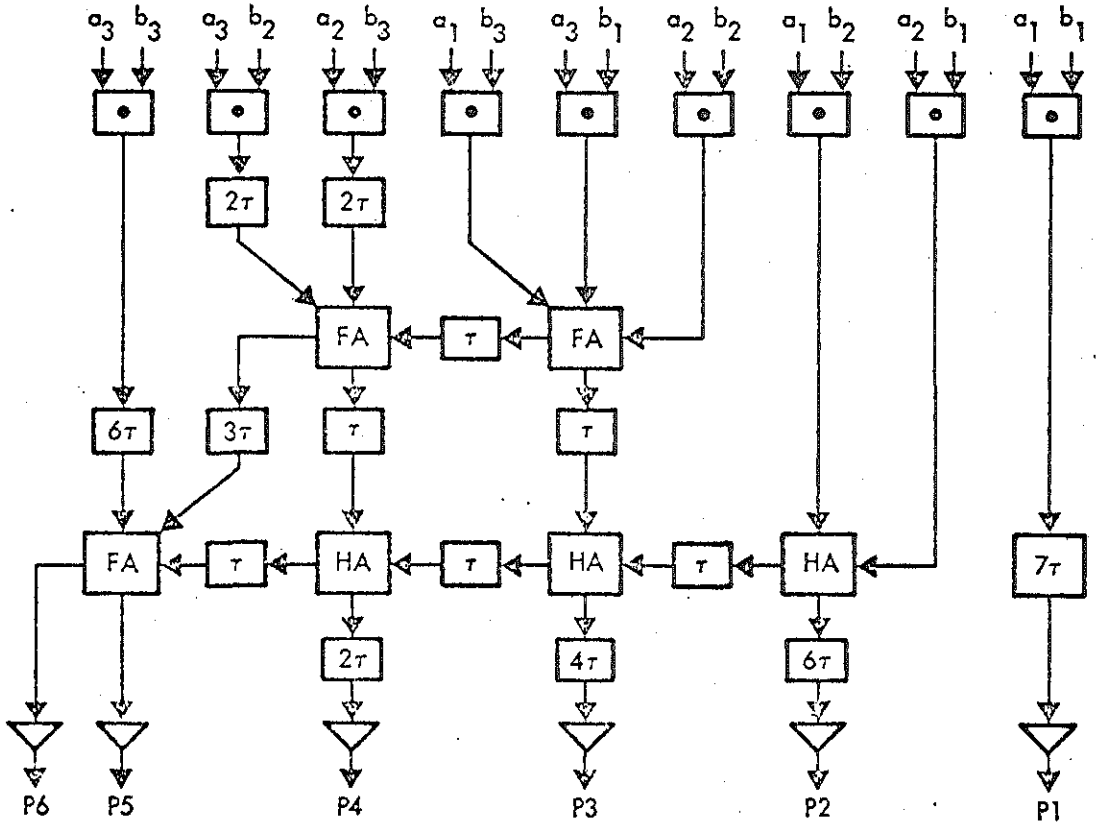
Using similar arguments, we can show that the construction of the pipeline multiplier has an analogous structure. Figure 28 represents a 3 x 3 multiplier realized in this fashion. Note the addition of a new function on this figure, the AND circuit needed to perform the first operation in the multiplication. We see also from this figure that the delays are not as regularly spaced throughout the structure as they are with the adder function. The inset on Figure 28 shows how this function would be done by hand and where the carries are applied throughout the function. Note that it is so structured that on one clock pulse we take in the two 3-bit words; these are propagated throughout the structure; and the 6-bit answer arrives at some time later, but again all bits arrive on one clock pulse.

The following two figures show some actual implementations of multipliers and adders realized in this technology. Figure 29 shows a 16 + 16 bit adder that produces a 17-bit sum. It is clear that the majority of the area here is taken up by the two triangular shaped delay lines on the input and the output. The actual arithmetic operations are being performed only along the diagonal of this device. Figure 30 shows and 8 x 8 multiplier with the inputs being on the left and 16-bit output on the right. Once again we see that the delays are not as evenly spaced in this realization as they are with the adder.

Now an important item to note here is that the majority of these delays in both the adder and multiplier can in fact be eliminated from many implementations. This is true whenever we are able to work with a single chip that combines a number of arithmetic operations. These delays are really needed only to synchronize the inputs to the chip and to synchronize the outputs. On operations between input and output, the data need not by synchronized and the chips can work with what is commonly called skewed arithmetic. That means that in the common LSI implementation of these functions the areas required is significantly less than that shown in Figures 29 and 30.

As an indication of the types of output, waveforms seen with these devices (Figure 31) records some data from a 4-bit 2-word adder array. The vertical scale is 5 volts per centimeter and the outputs are seen to be quite distinguishable.

3X3 PIPELINE MULTIPLIER



		A-WORD	a3	a2	a1
		B-WORD	b3	b2	b1
			a_3b_1	a_2b_1	a_1b_1
		a_3b_2	a_2b_2	a_1b_2	
	a_3b_3	a_2b_3	a_1b_3		
	C_5	C_4	C_3		
	C'_5	C'_4			
PRODUCT p^6	p^5	p^4	p^3	p^2	p^1

FIGURE 28

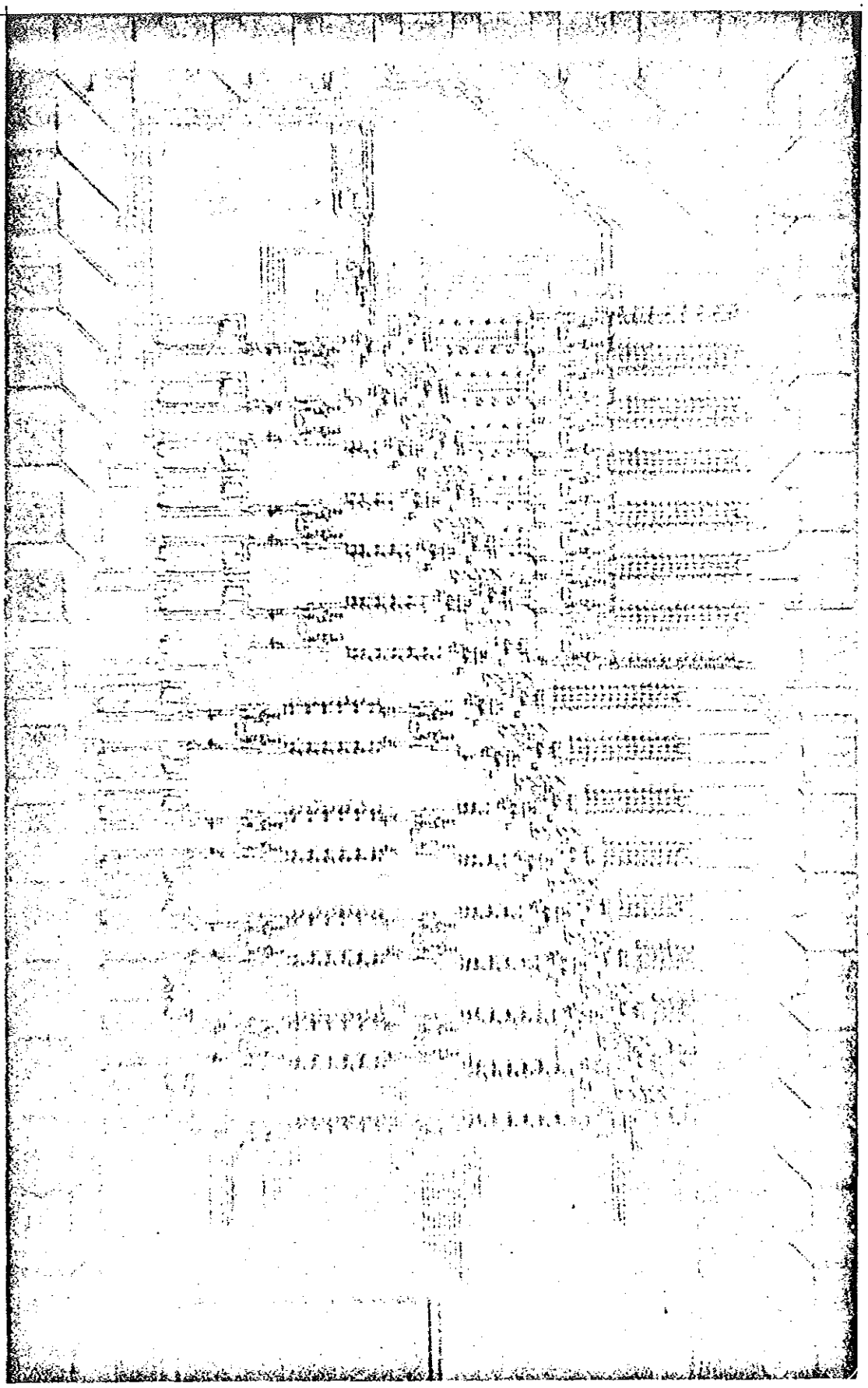


FIGURE 29

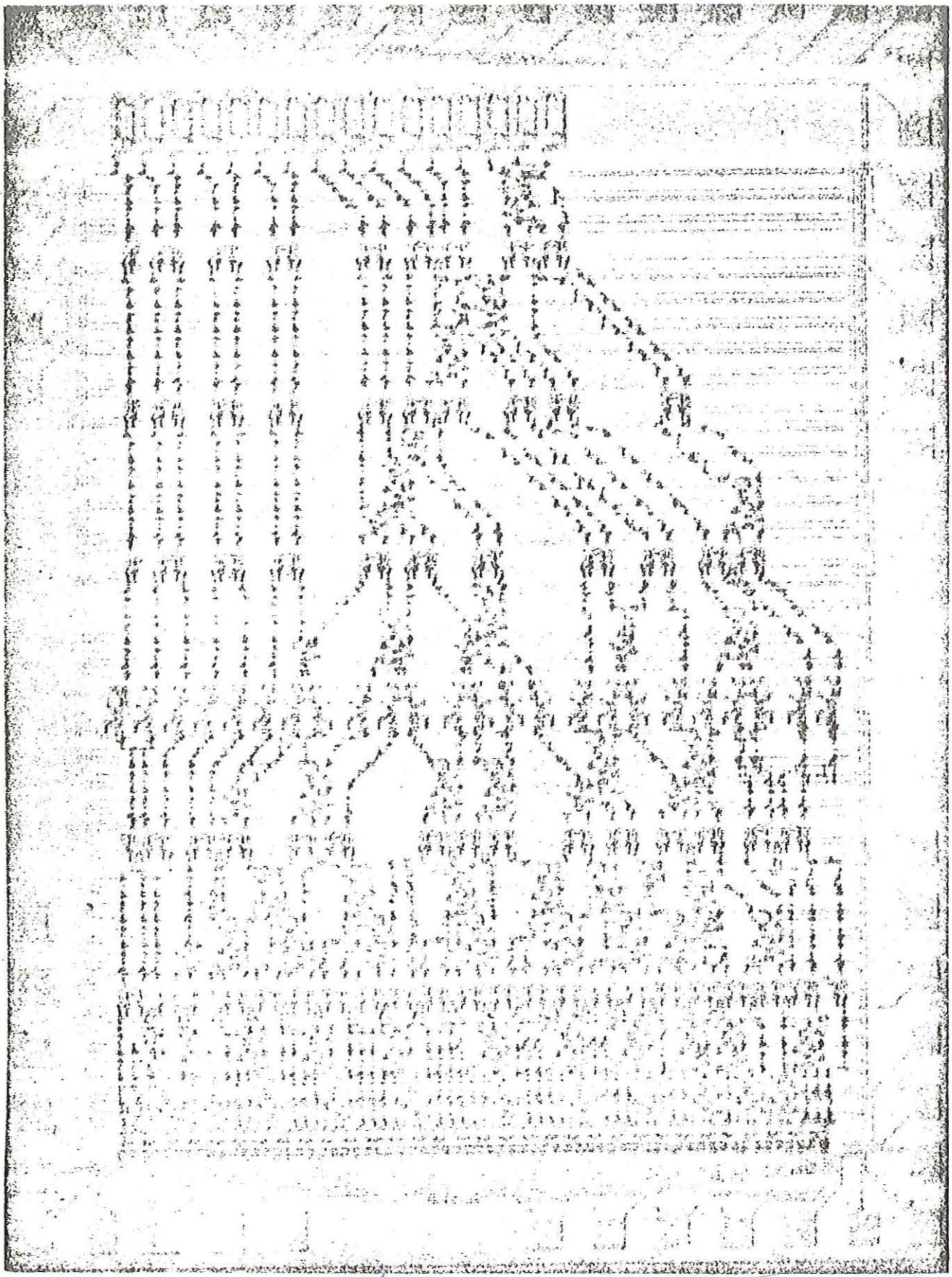


FIGURE 30

OUTPUT SIGNALS FROM TWO-WORD, FOUR-BIT ADDER ARRAY
WHEN THE A-WORD = 1110 AND B-WORD = 0100

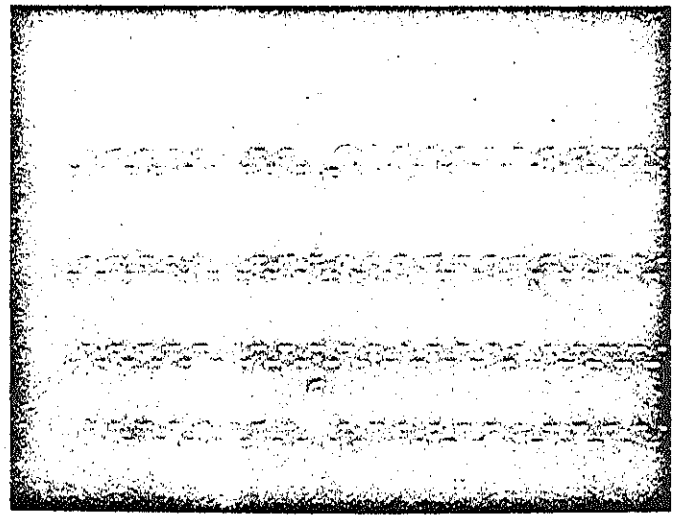
NOT SHOWN, $S_1 = 0$

OUTPUT BIT $S_2 = 1$

OUTPUT BIT $S_3 = 0$

OUTPUT BIT $S_4 = 0$

OUTPUT BIT $S_5 = 1$



OUTPUT PULSES FROM TWO-WORD, FOUR-BIT ADDER ARRAY
WHEN THE A-WORD = 0101 AND THE B-WORD = 0101

NOT SHOWN, $S_1 = 0$

OUTPUT BIT $S_2 = 1$

OUTPUT BIT $S_3 = 0$

OUTPUT BIT $S_4 = 1$

OUTPUT BIT $S_5 = 0$

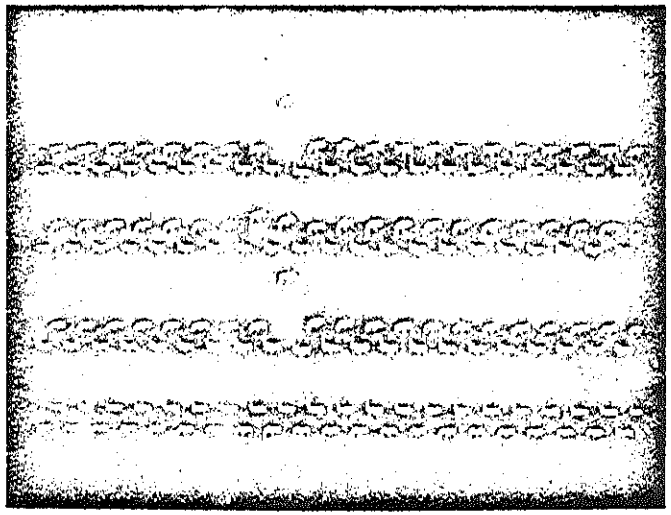


FIGURE 31

Now that we have looked at some of the basic implementations of logic and arithmetic functions, let us examine some applications for these devices. Figure 32 summarizes some research that we did into the application of digital CCD's. On the left hand side we list a number of systems that we examined and then in the center the hardware application that was appropriate for that system. Finally, on the right hand side we listed our conclusion; in this case a general purpose single signal processing chip could be made to implement a number of functions since their operations were so similar. This led us to design conceptually a arithmetic unit chip that could be reconfigured electronically to perform a number of functions. Figure 33 lists the basic features of this chip.

We see that the chip was primarily configured to do the FFT kernal operation. This function appeared in many of the systems designs that we studied and seemed to be a fairly universal operation. We decided that the accuracy required for most systems meant that we needed to do 16-bit arithmetic. The full kernal operation requires four 16-bit multipliers and three 16-bit adders on a single chip. The fundamental kernal actually requires four adders, but due to the pipeline operation of the chip we can time multiplex one of the adders so that only three are really required.

The signals are time multiplexed onto and off of the chip in order to reduce the number of input/output pins that would otherwise be necessary. In the way that it was conceived, the chip also performed the decoding of control lines and self structuring of input data to maintain several options in its operation. We also elected to have feedback from the output to the input right on the chip, again in an effort to reduce the number of pins required. In order to explain the basic operation of the chip, let us look at Figure 34.

Figure 34 shows the flow diagram for an 8 point DFT. Note that a "butterfly" structure is apparent in this diagram and is in fact the basis for the name. We also see that the input is scrambled in going through the FFT. Note for example the F4 sample appears as the second line on input in this diagram, but the second line on output on the right hand side of the diagram is an F1 sample. This scrambling (generally termed bit reversing) must be accounted for in the FFT operation. Our concept was to produce a single chip that would do the butterfly operation and to make a companion chip which would do the memory operation required. Therefore we conceived of the structure shown in Figure 35.

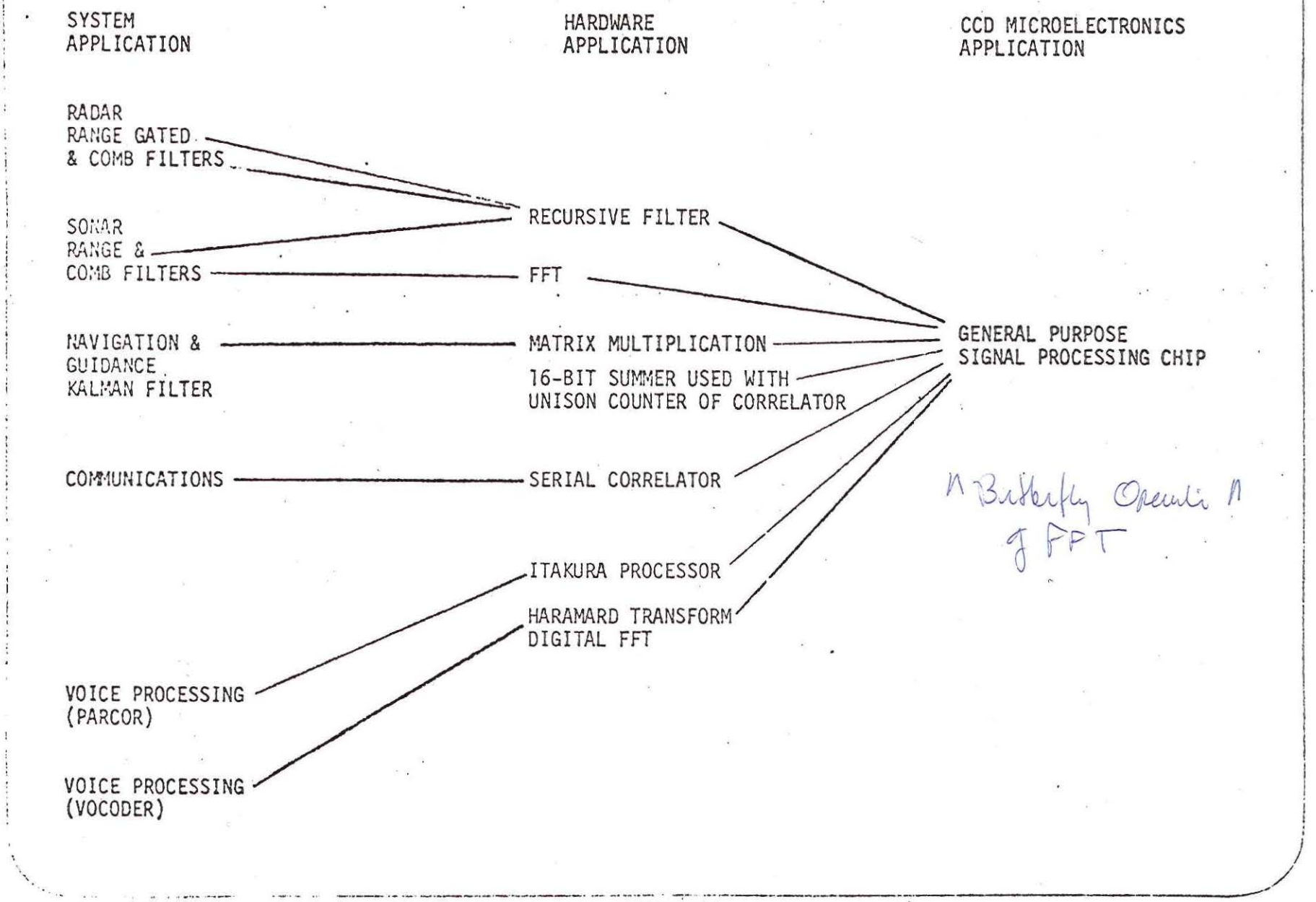

SYSTEM APPLICATION
HARDWARE APPLICATION
CCD MICROELECTRONICS APPLICATION
RADAR RANGE GATED & COMB FILTERS
SONAR RANGE & COMB FILTERS
NAVIGATION & GUIDANCE KALMAN FILTER
COMMUNICATIONS
VOICE PROCESSING (PARCOR)
VOICE PROCESSING (VOCODER)
RECURSIVE FILTER
FFT
MATRIX MULTIPLICATION
16-BIT SUMMER USED WITH UNISON COUNTER OF CORRELATOR
SERIAL CORRELATOR
ITAKURA PROCESSOR
HARAMARD TRANSFORM DIGITAL FFT
GENERAL PURPOSE SIGNAL PROCESSING CHIP
N Butterfly Operation of FFT

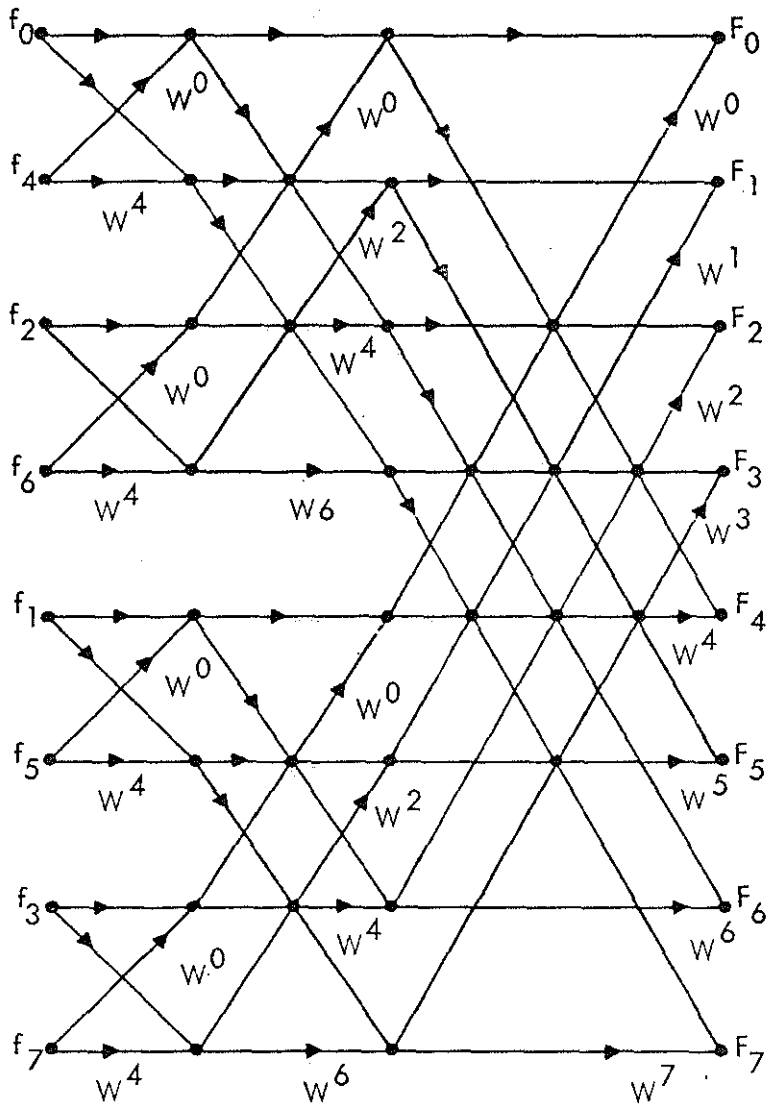
FIGURE 32

CCD ARITHMETIC CHIP CONFIGURATION

- CONFIGURED FOR FFT KERNAL OPERATION
- 16-BIT PARALLEL ARITHMETIC
- FOUR 16-BIT MULTIPLIERS AND THREE 16-BIT ADDERS ON CHIP
- SIGNALS TIME MULTIPLEXED AT INPUT AND OUTPUT
- ON CHIP DECODER FOR CONTROL INPUTS
- CHIP CAN PERFORM VARIETY OF RECURSIVE DIGITAL FILTERING FUNCTIONS
- OUTPUT TO INPUT FEEDBACK INCLUDED ON CHIP

FIGURE 33

EIGHT POINT DFT



EIGHT POINT DFT COMPLETELY REDUCED TO COMPLEX MULTIPLICATIONS AND ADDITIONS BY REPEATED DECIMATION IN TIME. FOR THIS DIAGRAM INPUTS ARE REQUIRED TO BE IN BIT-REVERSED ORDER.

FIGURE 34

FFT KERNAL OPERATION

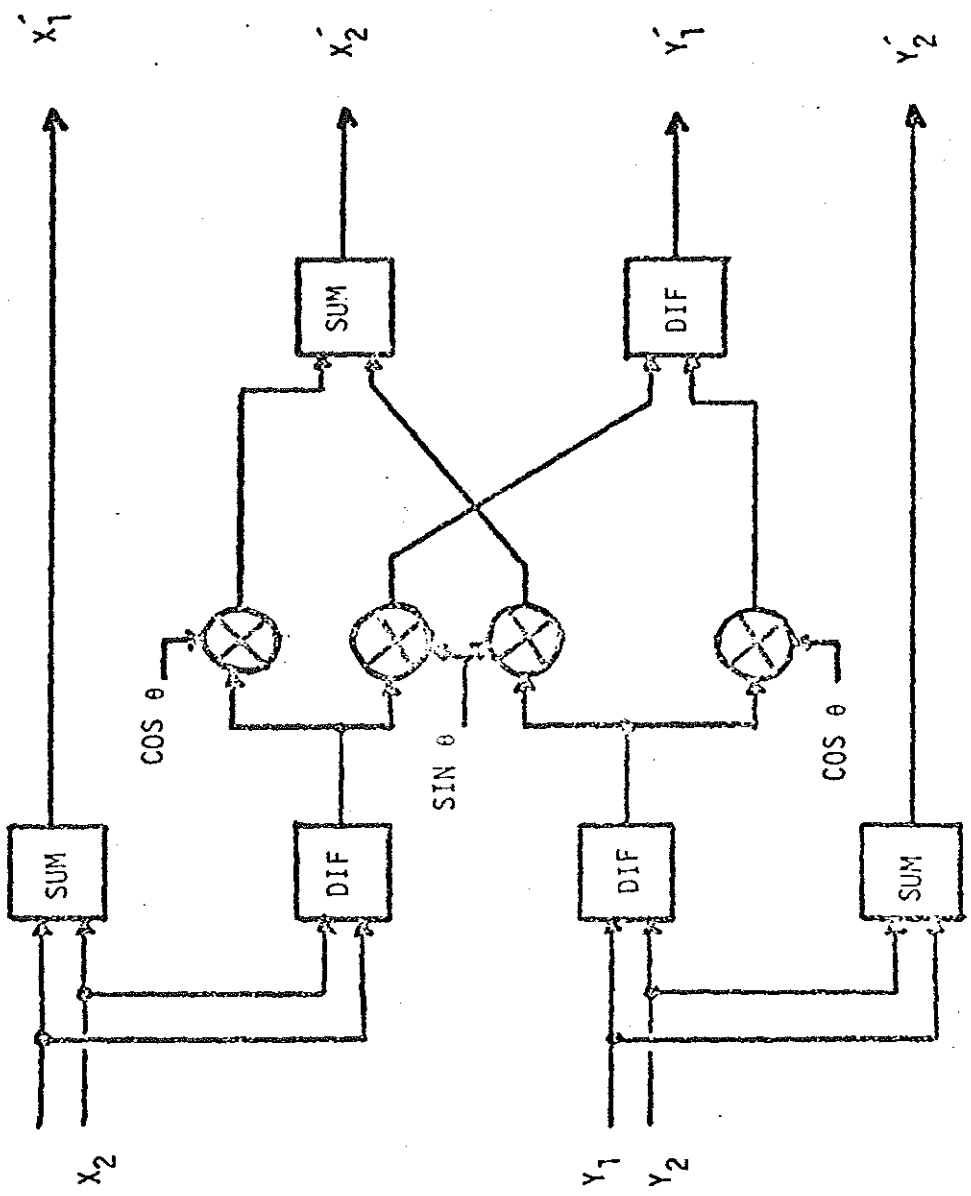


FIGURE 35

This shows the basic operations that must be performed to do the FFT kernal. If one can perform these operations however, it is also useful to be able to change the order in which you do the sum, differencing and multiplication so that other electronic functions can be achieved. That led us to design the chip shown in Figure 36. Note that this chip has a number of muliplexing and delay gates scattered throughout the multipliers and adders previously discussed. This allows us to make the chip much more flexible and to conceptually use the chip to perform a number of functions in addition to the FFT kernal.

Figure 37 indicates our original concept for the layout and area required for the various functions of this chip. Note that the multipliers occupy the majority of the chip area as one would expect. The next few figures indicate various potential uses for such a chip.

Since we have multiplication, addition and multiplex gates available to us, we can restructure the order of operation on the chip and achieve a number of distinct functions. Figure 38 shows a single pole recursive filter with scaled output that can be achieved by this chip. In the A portion of the figure on the left hand side we see the conical flow form structure for a single pole recursive filter; the B portion of the figure shows how this would be achieved with the previously discussed arithmetic chip by implementing a zero for certain multiplication factors which in effect, then, disable certain lines. The same concept can be applied to a number of other functions. Figure 39 shows the same idea used to produce two single pole recursive filters. Figure 40 shows how a lattace filter can be produced in this way while Figure 41 shows how a second type of lattace filter can be realized. In Figure 42 we use the CCD arithmetic chip to produce a 2-pole recursive filter, again through the judicious use of the multipliers and MUX circuits on the chip. Finally, Figure 43 shows one other conic form realized with the same chip.

Now as we mentioned earlier, a companion memory chip must be configured to properly perform the FFT. The one that we will discuss here was sized as a 512 word, 16-bit per word chip. It was configured to perform a 256 point FFT where we have two parallel shift registers of 128 complex words each. In addition to performing this function, we conceived the chip with other timing and control circuits included such that we can do FFT of dimensions smaller than 256 bits. The chip operation is such that is automatically performs the bit reversing sequence needed. Figure 45 indicates how this is achieved.

TRW
SYSTEMS GROUP

ARITHMETIC FUNCTIONS

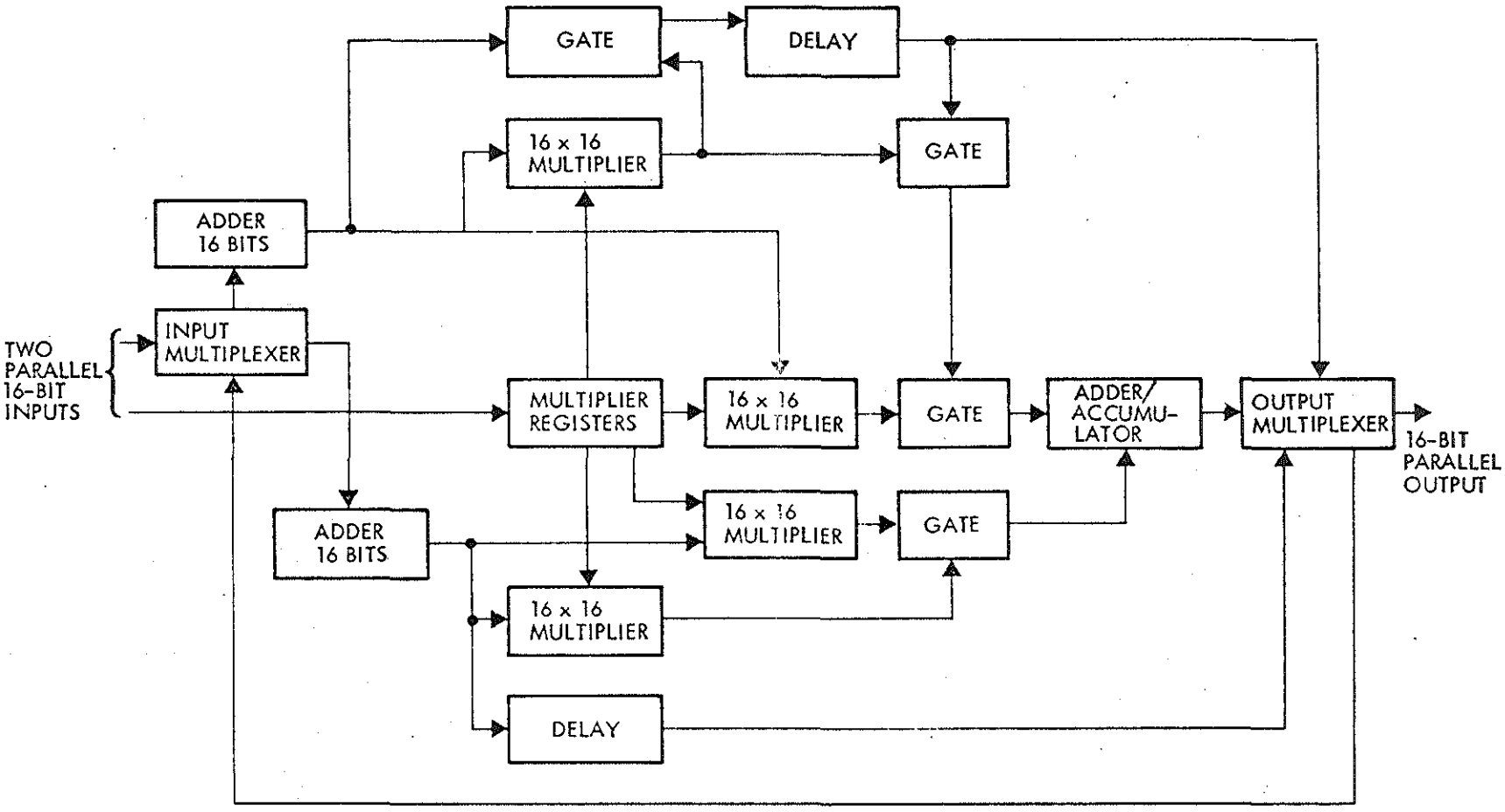


FIGURE 36

TRW SYSTEMS GROUP

ARITHMETIC UNIT LAYOUT

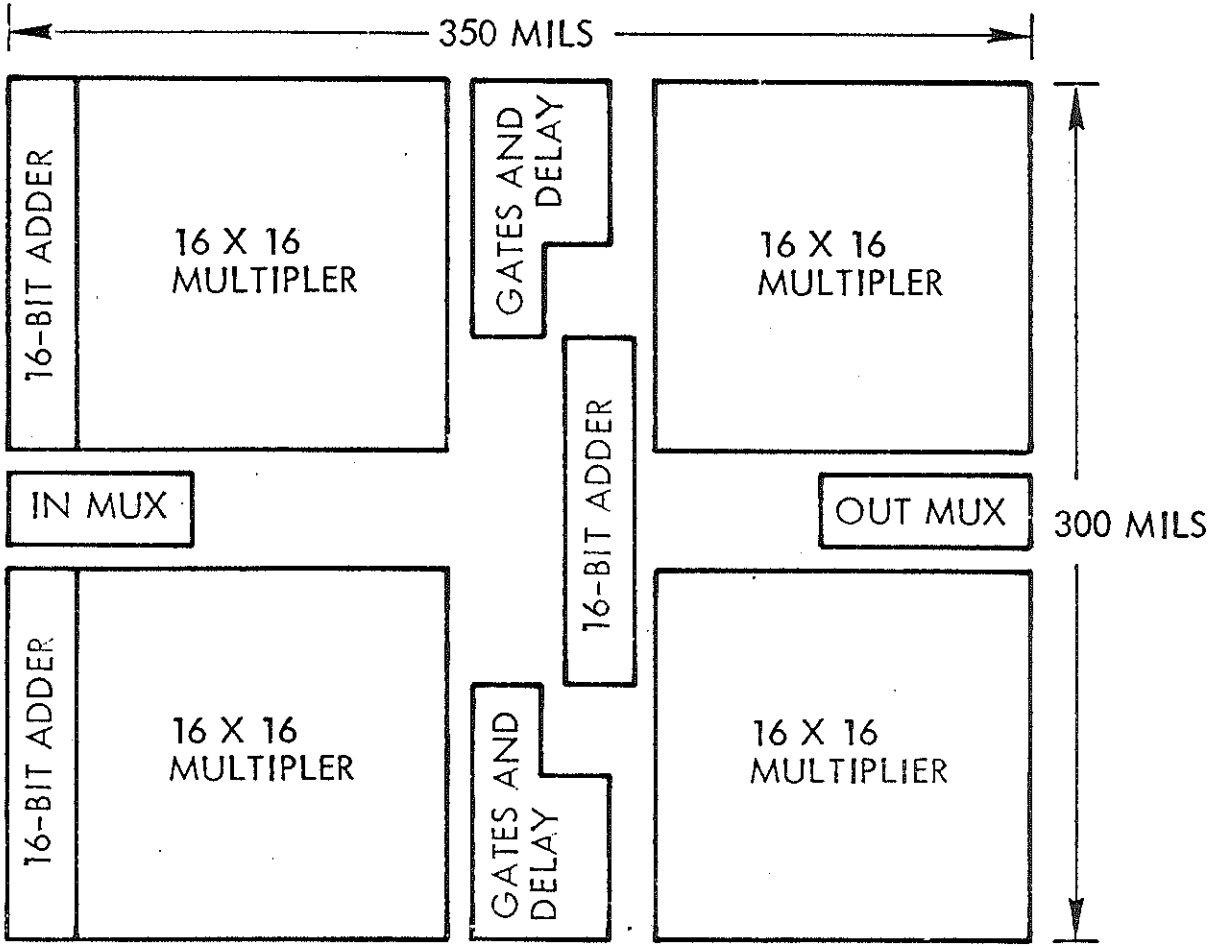
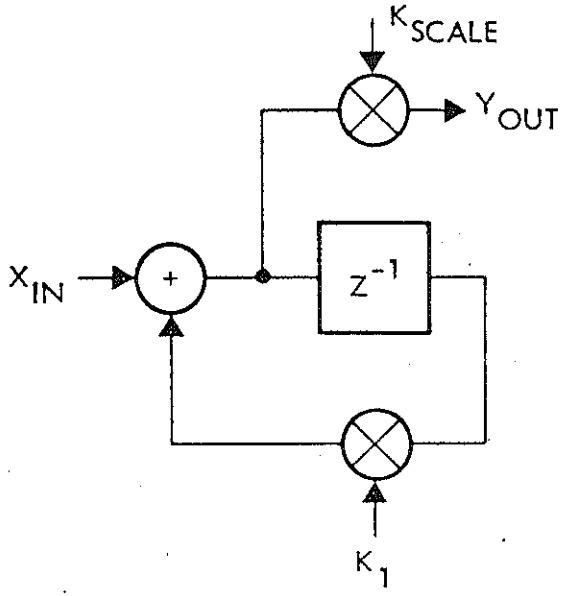


FIGURE 37

TRW
SYSTEMS GROUP

SINGLE-POLE RECURSIVE FILTER WITH SCALED OUTPUT

(A)



$$Y_{OUT} = \frac{K_{SCALE} X_{IN}}{1 + K_1 A^{-1}}$$

(B)

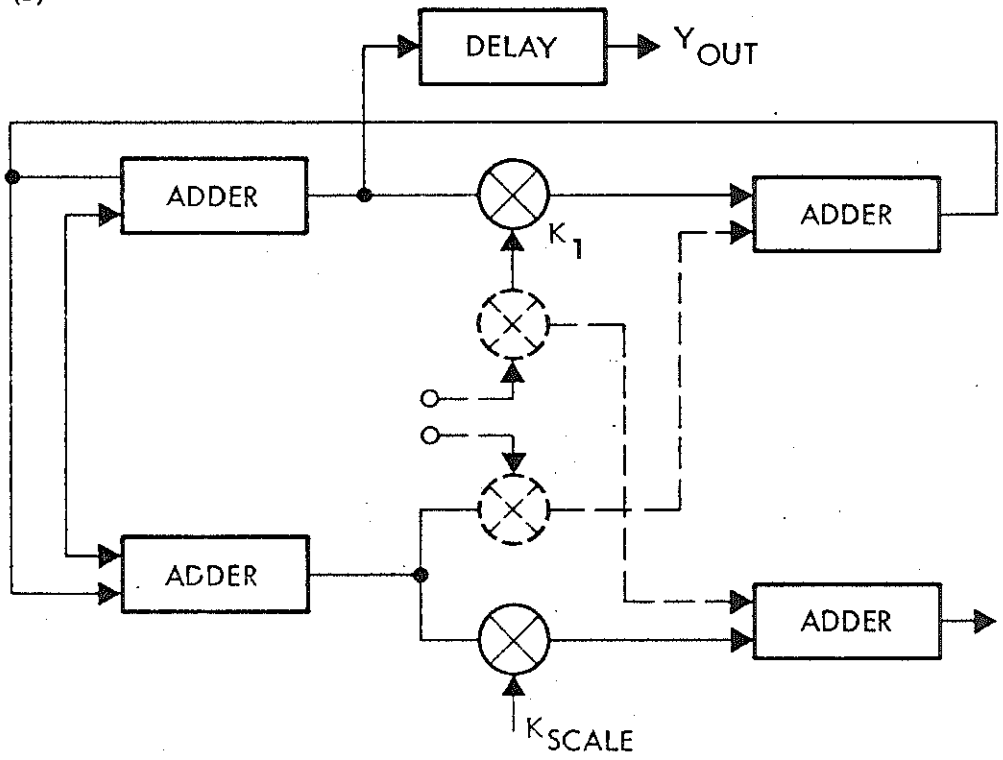


FIGURE 38

TRW SYSTEMS GROUP

USE OF CCD ARITHMETIC UNIT AS PARALLEL SINGLE-POLE RECURSIVE FILTERS

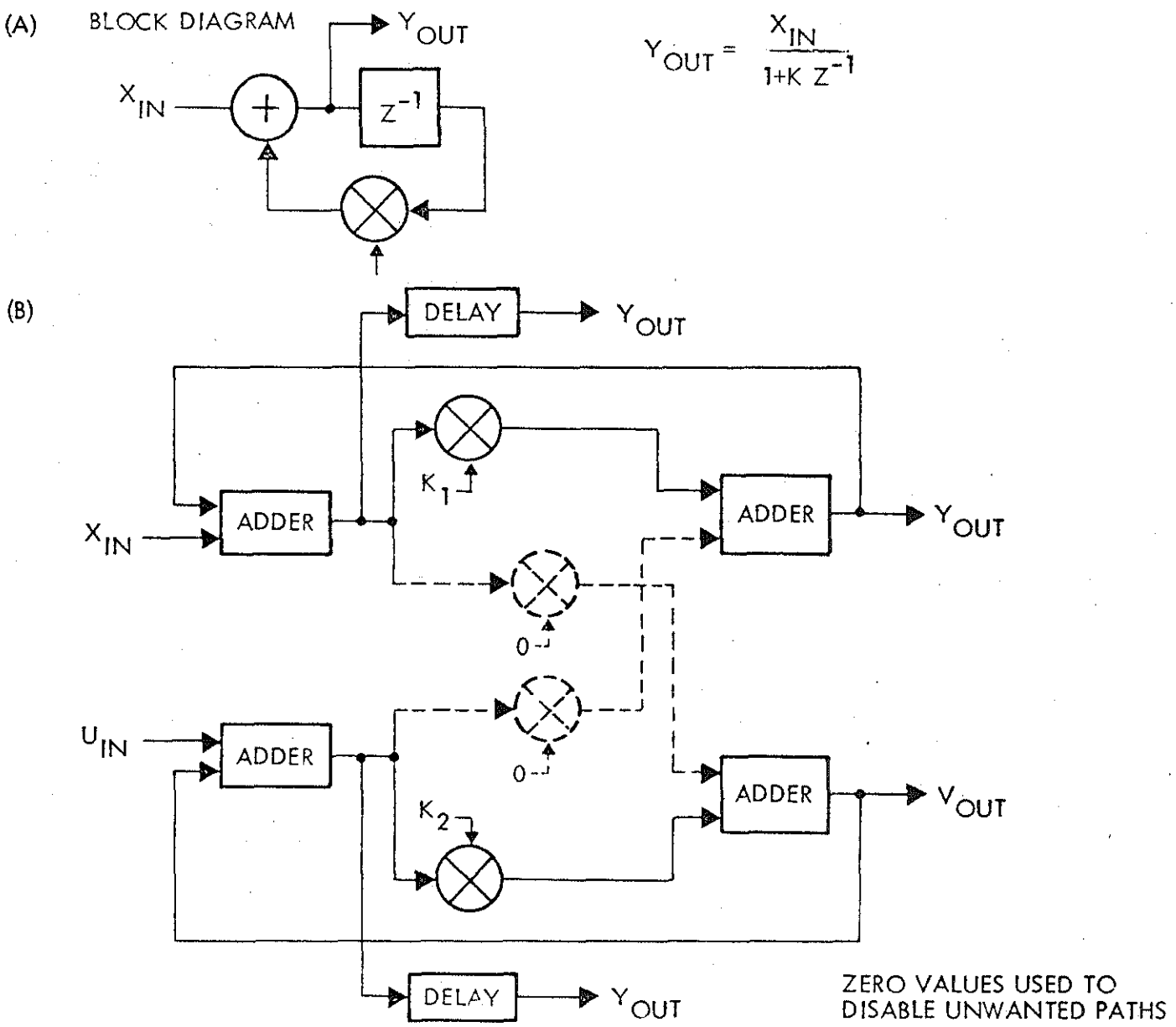


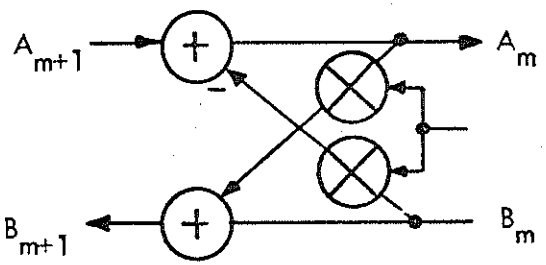
FIGURE 39

TRW SYSTEMS GROUP

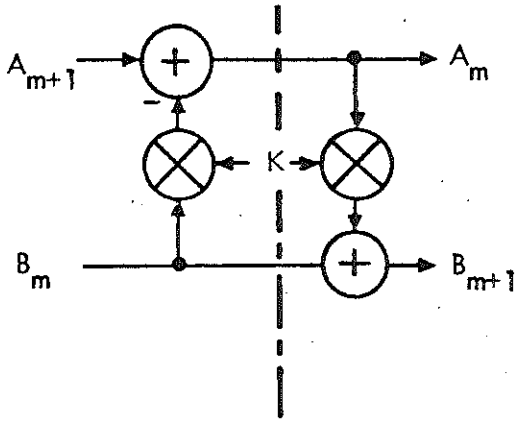
USE OF CCD ARITHMETIC UNIT AS LATTICE FILTER (SYNTHESIZER)

(A)

BLOCK DIAGRAM



EQUIVALENT BLOCK DIAGRAM



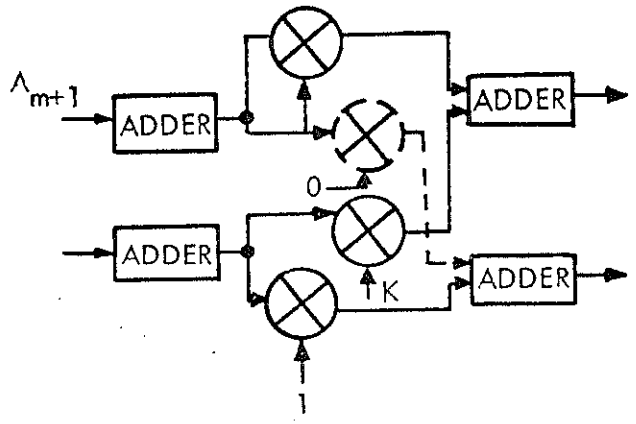
$$A_m = A_{m+1} - K\,B_m$$

$$B_{m+1} = K\,A_m + B_m$$

(B)

TWO PASSES THROUGH ARITHMETIC CHIP REQUIRED

FIRST PASS



SECOND PASS

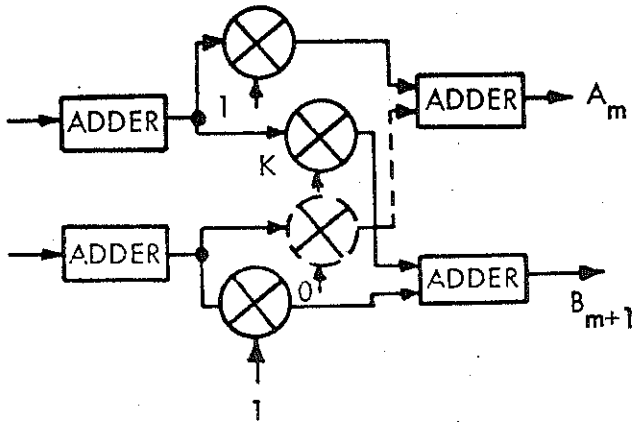
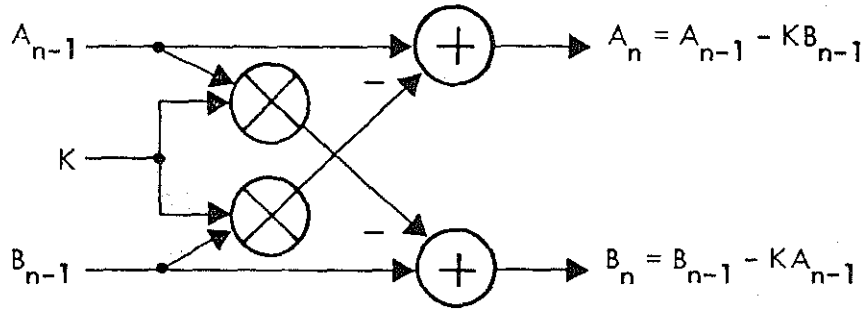


FIGURE 40

TRW
SYSTEMS GROUP

USE OF CCD ARITHMETIC UNIT AS LATTICE FILTER (ANALYZER)

(A)



(B)

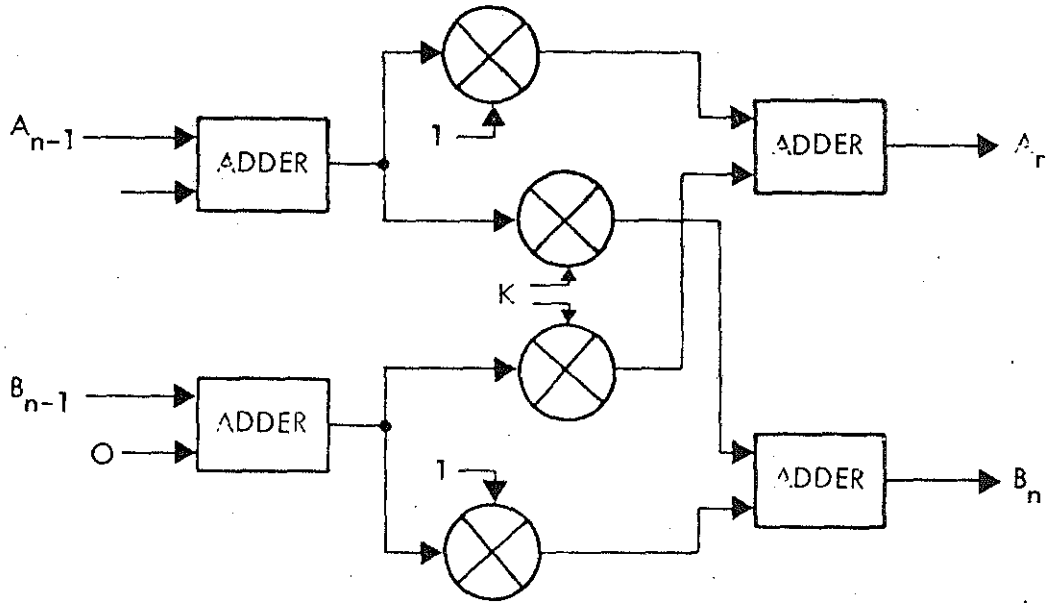
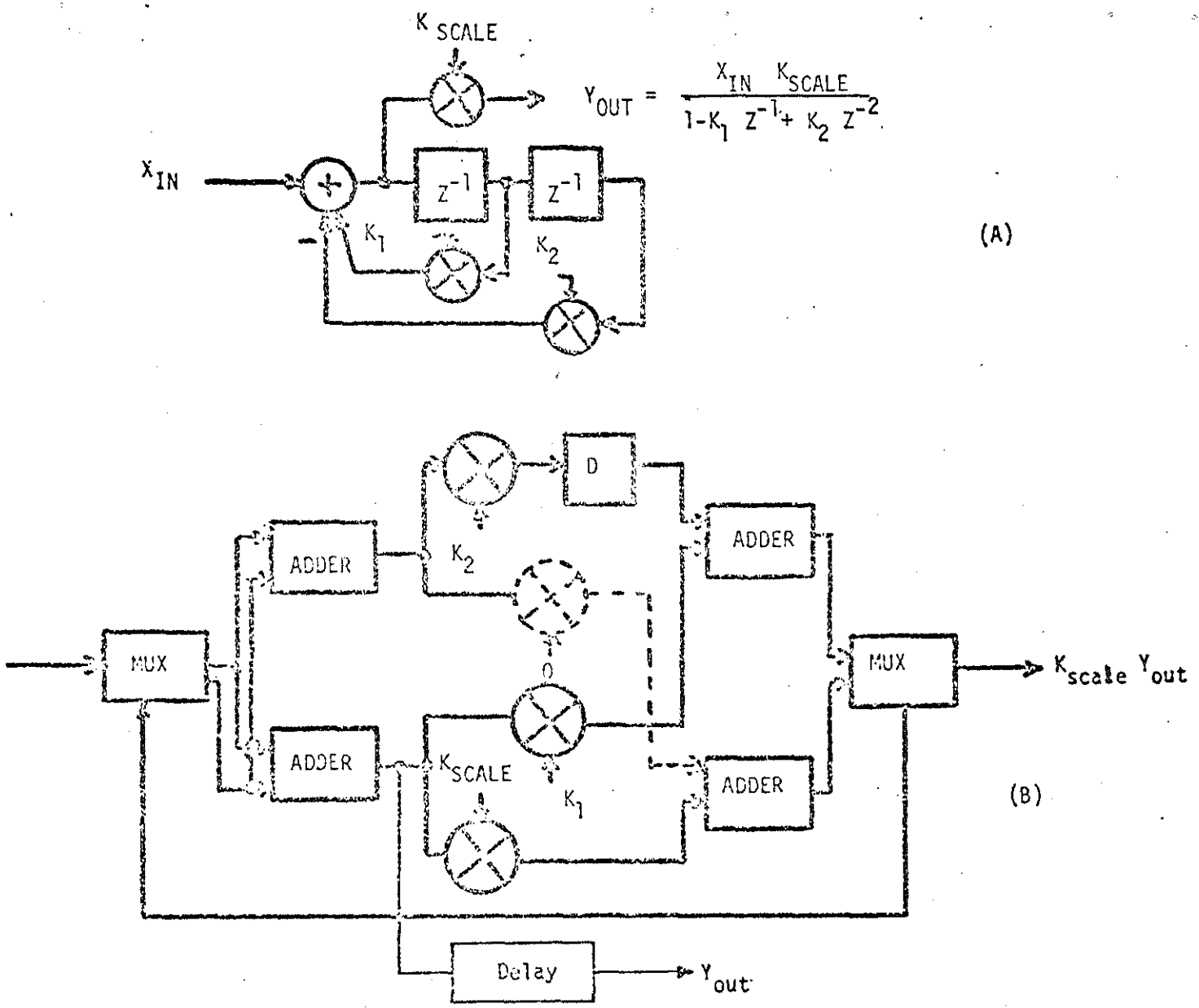
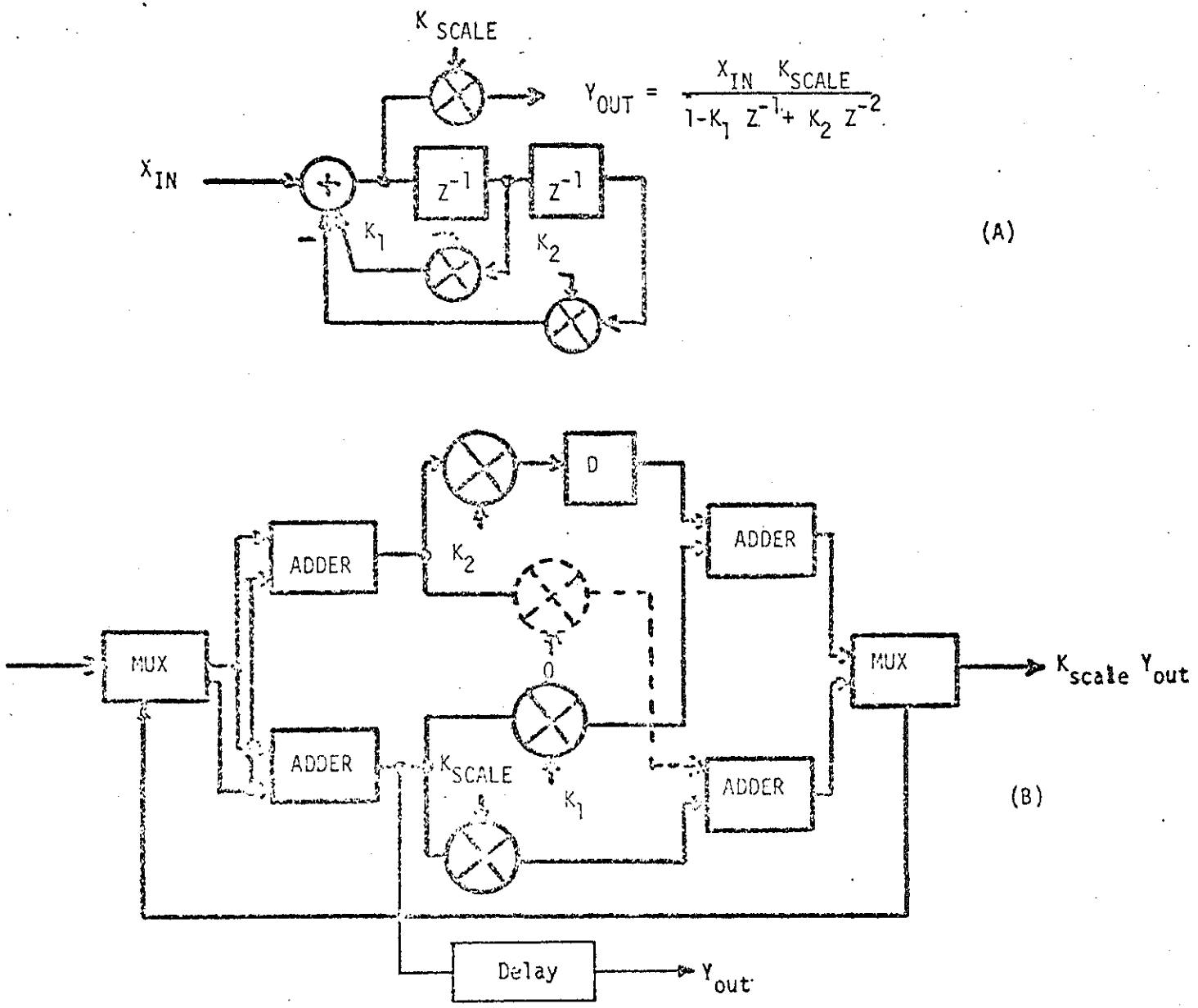


FIGURE 41



Use of CCD Arithmetic Chip as Two-Pole Recursive Filter

FIGURE 42



Use of CCD Arithmetic Chip as Two-Pole Recursive Filter

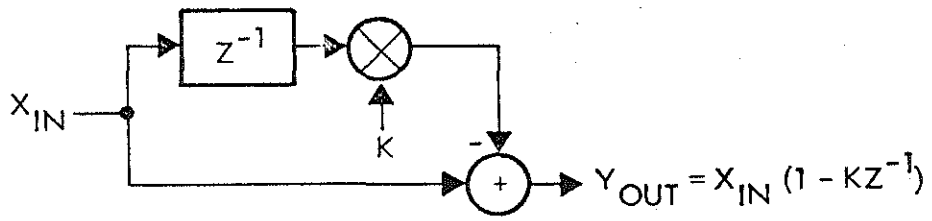
FIGURE 42

TRW
SYSTEMS GROUP

DATA INPUTS TO THE CCD ARITHMETIC UNIT

(A)

BLOCK DIAGRAM



(B)

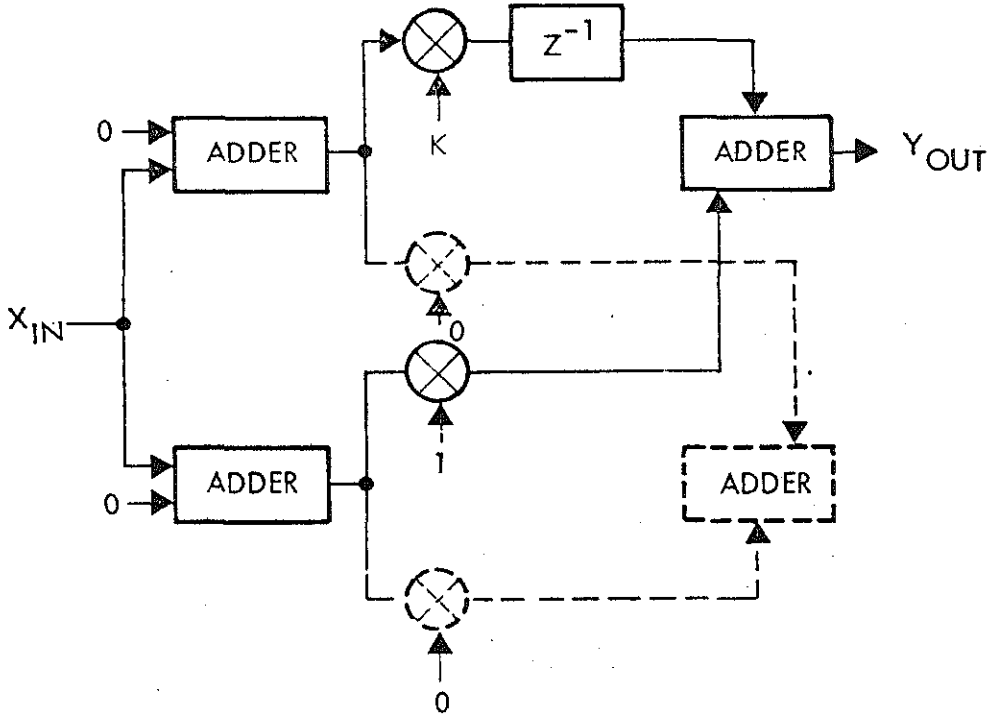


FIGURE 43

MEMORY CHIP CONFIGURATION

- 512 WORDS, 16 BITS/WORD = 8192 BITS
- CONFIGURED FOR 256 POINT FFT
  TWO PARALLEL SHIFT REGISTERS OF 128 COMPLEX WORDS.
- GATES INSERTED EVERY 2^M STAGES (M = 1, 2,..., 6)
  TO PROVIDE FFT DATA SEQUENCING
- ON CHIP TIMING AND CONTROL FOR FFT OF DIMENSION (4, 8, 16, 32, 64, 128, 256, MORE THAN 256).
- EITHER SEQUENTIAL OR "BIT REVERSED" OPERATION
- CHIP CAN ALSO OPERATE AS VARIABLE DELAY 1 TO 256 COMPLEX WORDS

FIGURE 44

FFT MEMORY CONFIGURATION

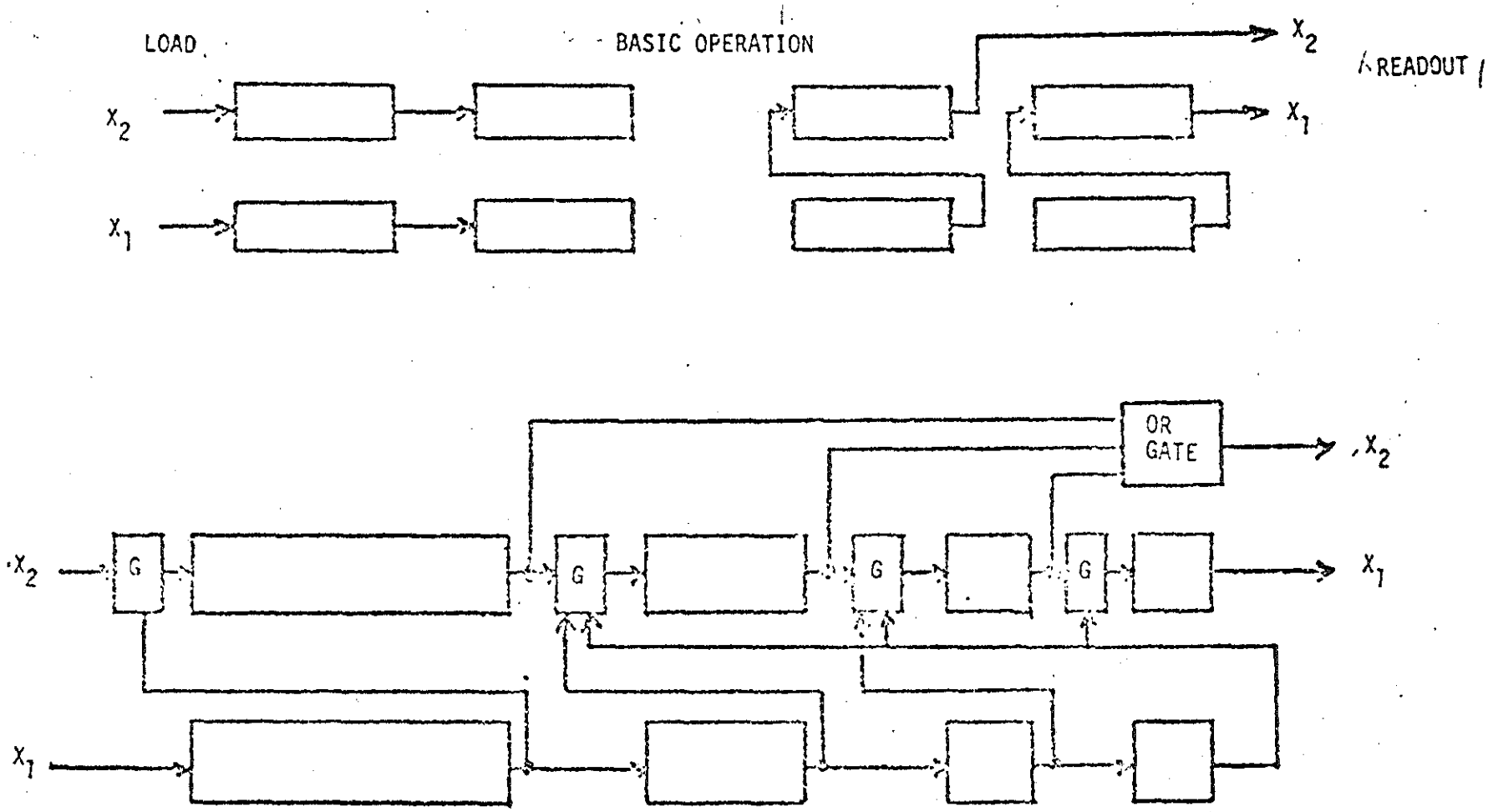


FIGURE 45

The top of Figure 45 shows first the loading operation and then the shifting operation needed to do the bit reversing. The bottom part of the figure indicates how this is continued throughout a shift register,portions of which are related to each other by powers of two. This power of 2 relationship allows us to use the memory chip with any size FFT up to 256. Figure 46 is an estimate of the size of such a chip.

In our study of systems, we performed one other investigation which came down to recommending a single correlator chip to perform a number of functions. Figure 47 indicates again the systems application on the left hand side and the hardware applications in the middle; all these are satisfied by the single correlator chip indicated at the right hand portion of the figure.

In Figure 48, we show our concept for doing a CCD digital correlator that employs digital multipliers to achieve multi-level correlation. That is to say, the figure showns a concept where a 4-bit quantized input signal is applied to the center shift register and the I and the Q reference channel (each quantized to 4-bits) are applied to the two outer channels. Than at each shift register location along the way a 4 x 4 multiplier is used to multiply the input signal by both the I and Q reference. In the chip configuration shown here, this occurs over 32 stages. The output of the multipliers can then go into a summing tree which is indicated on the figure as a simultaneous sum. This pipeline summation produces a 13-bit output for both the I and Q channels. These two 13-bit outputs represent the degree of agreement of the input signal with the I and Q reference. The chip parameters are summarized in Figure 49. Note that both forward and backward shifting is possible provided that the shift register design is properly performed. The previous FFT chip can be used to combine some of the outputs from a number of correlator chips to produce a correlation accuracy greater than 13-bits. Figure 40 shows the general layout such a chip would have.

In Figure 51 we indicate how we can cascade these correlator chips to produce correlations whose lengths are an integral multiple of 32-bits. The FFT chip is performing the arithmetic functions necessary to combine the I and Q channels.

Note that we could do other things with the correlator chip as shown by Figure 52, where a non-recursive digital filter is realized. Here we are spliting up the assumed 8-bit quantized input into the four most significant and the four less significant bits and then passing these through a correlator chip. Eventually the FFT chip combins the I and Q outputs.

ESTIMATED MEMORY CHIP LAYOUT

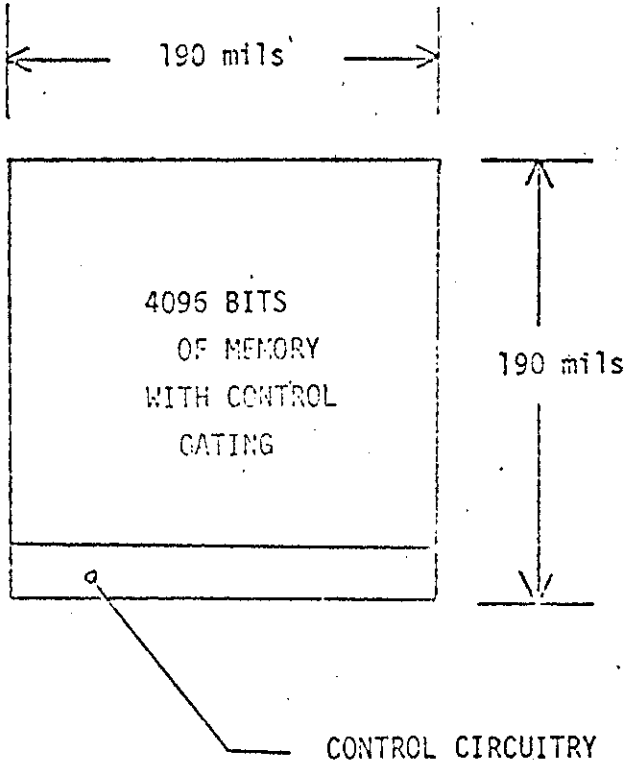


FIGURE 46

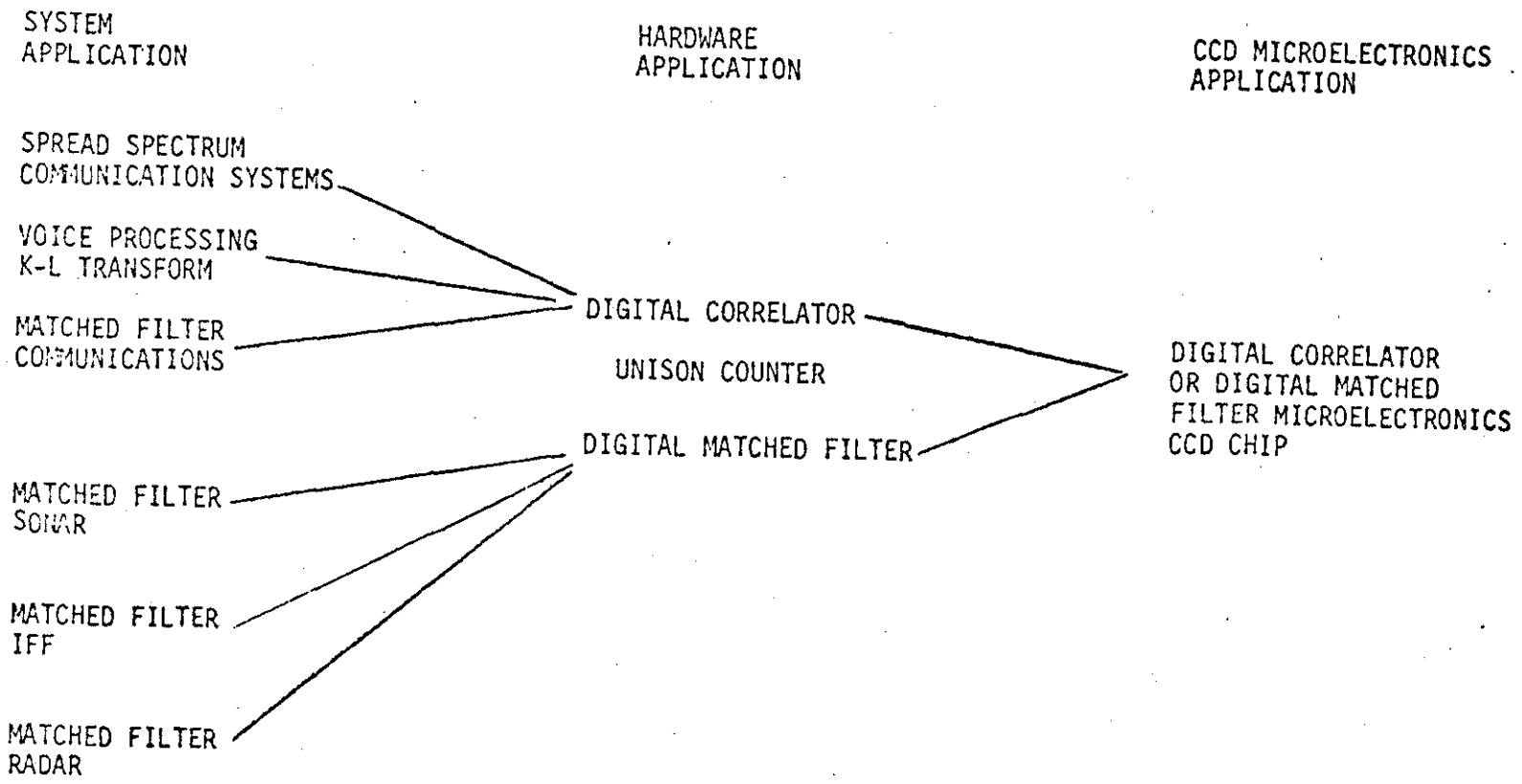
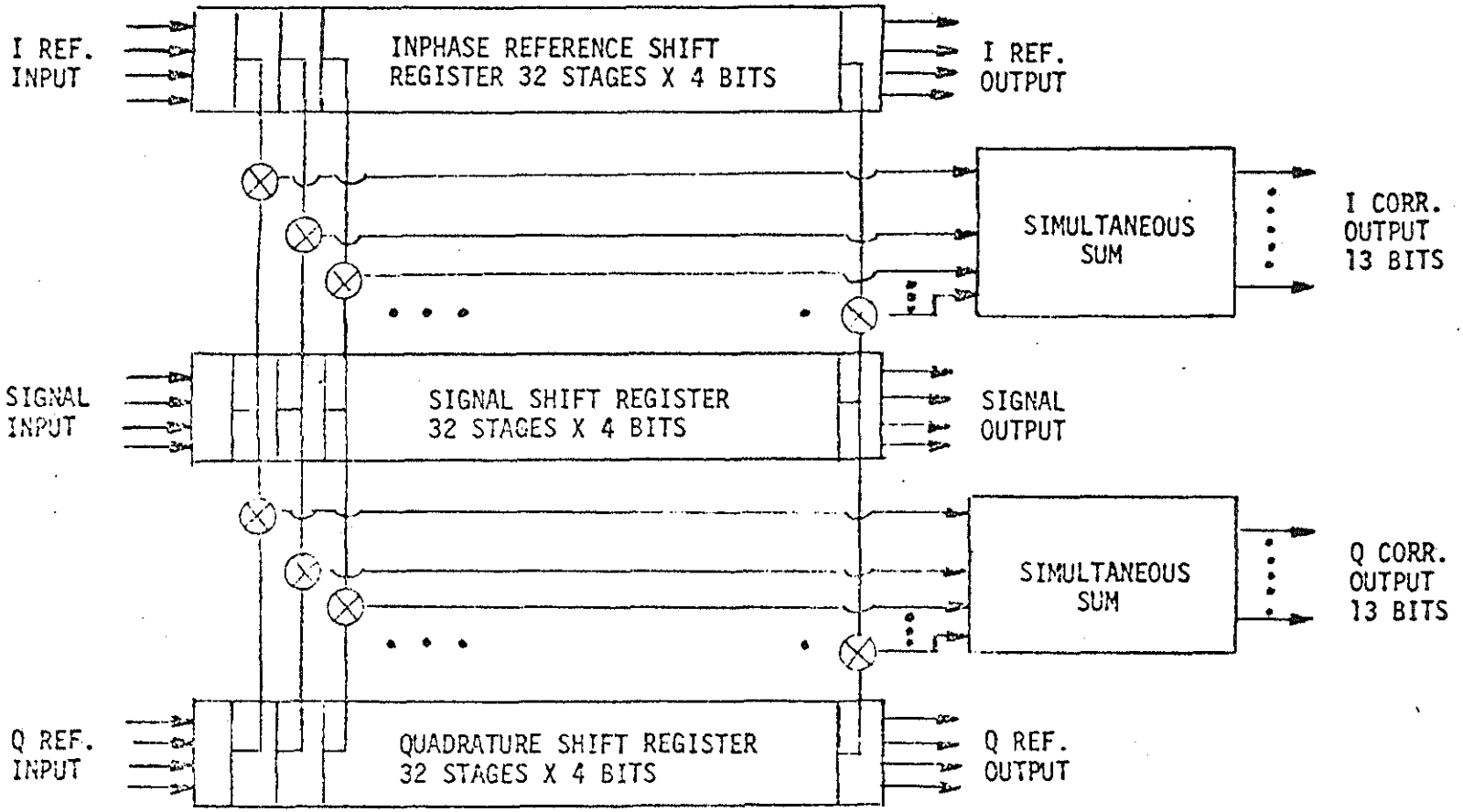

SYSTEM APPLICATION
HARDWARE APPLICATION
CCD MICROELECTRONICS APPLICATION
SPREAD SPECTRUM COMMUNICATION SYSTEMS
VOICE PROCESSING K-L TRANSFORM
MATCHED FILTER COMMUNICATIONS
DIGITAL CORRELATOR
UNISON COUNTER
DIGITAL MATCHED FILTER
DIGITAL CORRELATOR OR DIGITAL MATCHED FILTER MICROELECTRONICS CCD CHIP
MATCHED FILTER SONAR
MATCHED FILTER IFF
MATCHED FILTER RADAR

FIGURE 47



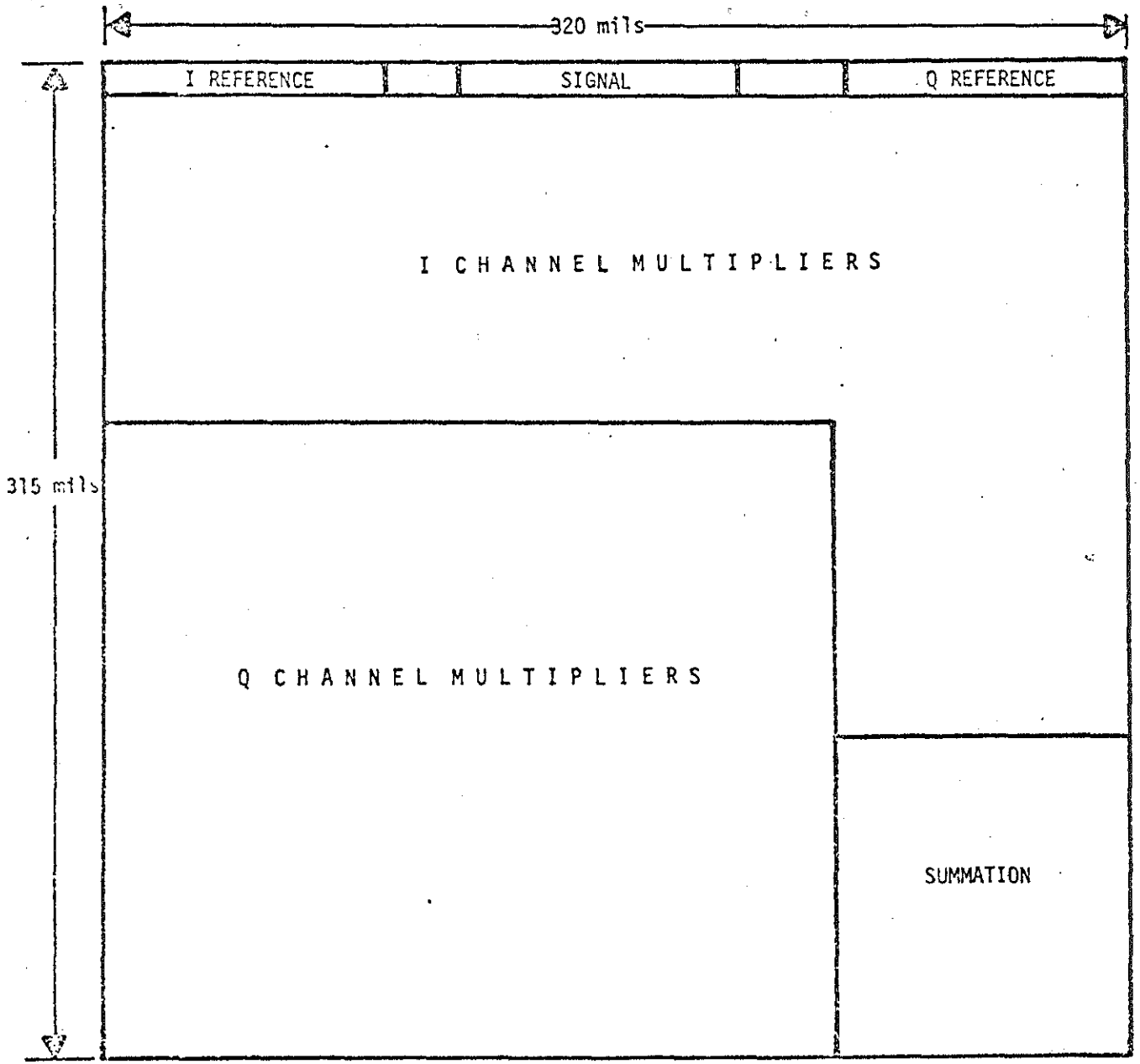
CCD DIGITAL CORRELATOR CONFIGURATION

FIGURE 48

DIGITAL CORRELATOR CHIP CONFIGURATION

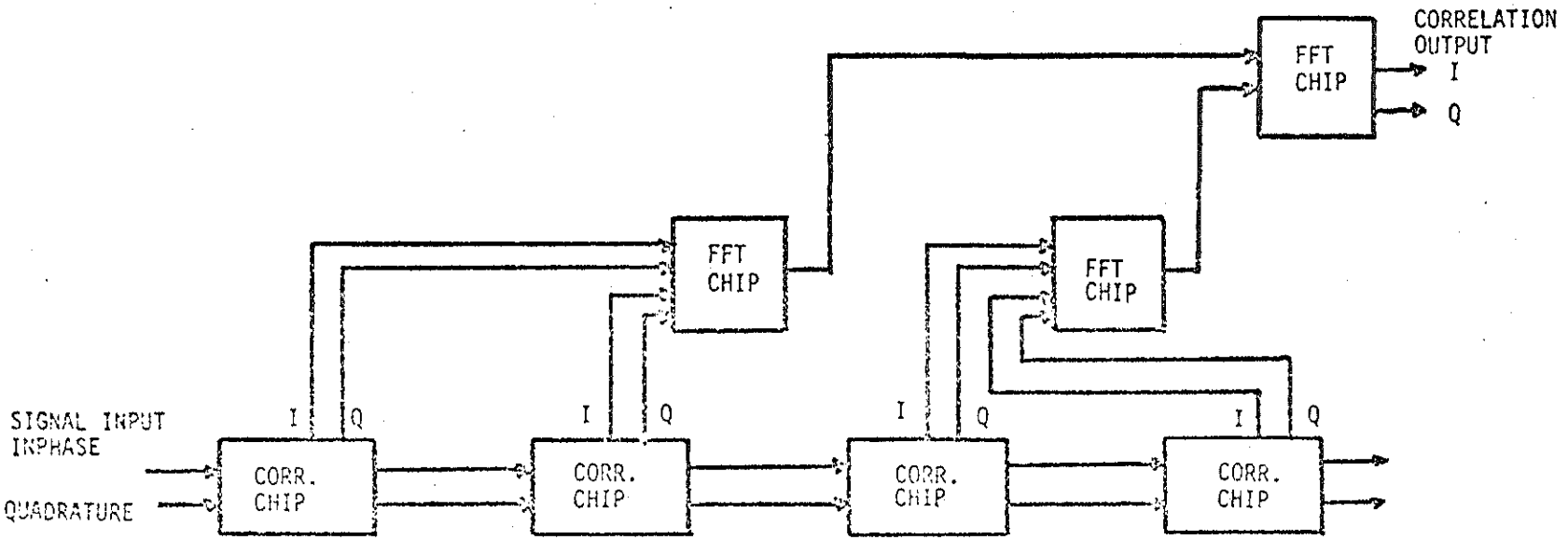
- THREE 32-STAGE SHIFT REGISTERS - 4 BITS/WORD (SIGNAL PLUS I AND Q REFERENCES).
- FORWARD-BACKWARD SHIFTING
- FOUR-BY-FOUR MULTIPLIERS CONNECTED TO EACH SIGNAL-REFERENCE SAMPLE PAIR.
- SIMULTANEOUS SUM OVER 32 PRODUCTS
- DIGITAL CORRELATION OUTPUTS (I AND Q) QUANTIZED TO 13 BITS.
- FFT ARITHMETIC CHIP CAN COMBINE OUTPUTS FOR CASCADE OPERATION

FIGURE 49



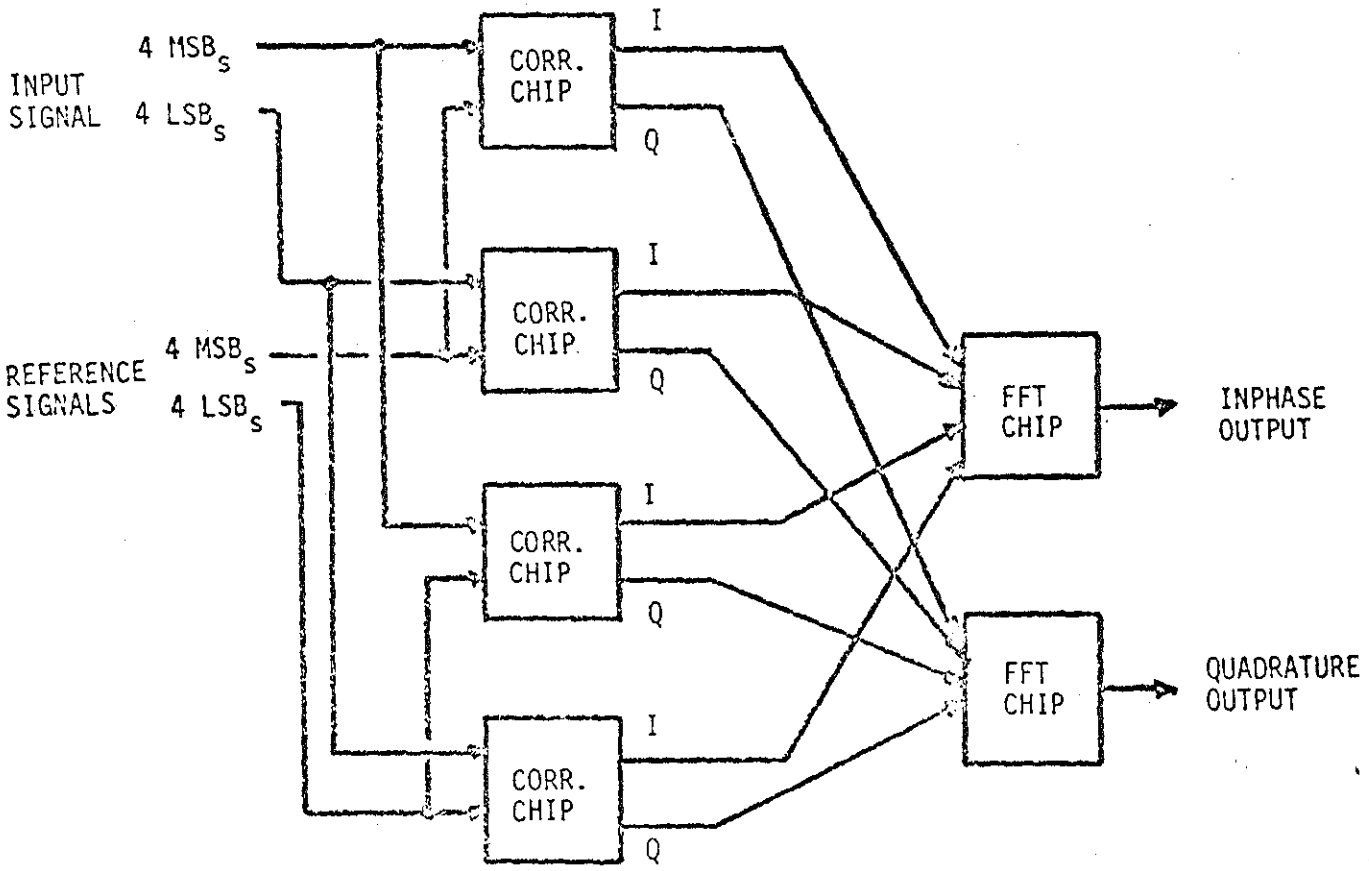
PLAN VIEW OF 32 STAGE CORRELATOR WITH DIGITAL OUTPUT

FIGURE 50



CASCADE OF CORRELATOR CHIPS

FIGURE 51



NONRECURSIVE DIGITAL FILTER - 8 BIT QUANTIZATION

FIGURE 52

Some other systems that we have studied with the digital CCD techniques will now be described. One such system is a voice processor. Figure 53 indicates the various functional blocks necessary to perform Itakura voice processing. On this figure several CCD chips are listed. In fact the CCD chips perform most of the arithmetic needed to realize the algorithm. Figure 54 summarizes the three main CCD chips needed. A full 80% of all the operations in the algorithm are performed by these three chips. Note that the arithmetic chip in fact looks like a quarter butterfly chip; that is, it has a single 16 x 16 multiplier and a 32 + 32 adder on the chip. In fact, with the proper control chip this arithmetic unit could perform the FFT functions described earlier.

Figures 55, 56 and 57 show the various functions required on the three chips being discussed.

Another chip that we have configured for the digital CCD technology involved the use of the Hadamard transform. Hadamard transforms are area transforms commongly used in video data reduction. Figure 58 shows the operations required at two stages typical of a Hadamard transform. We see that additions, subtractions, delays, muxing and select and division operations all must be performed in a single Hadamard section. This pipeline operation is ideally suited for the high density, low power capability of digital CCD's.

Figure 59 indicates the logic necessary in each of the cells of a Hadamard transform. Note that we have to refresh and invert signals, we have to perform ANDing functions in order to do the muxing, we have to perform charge multiplication in order to do the fan out and we have to perform ORing functions in order to achieve some of the other operations necessary. All of this is in addition to the basic addition and subtraction operations needed. Figure 60 indicates how a typical cell would layout in the various areas taken by each of the functions. Note that all of this is achieved in an area which is just 16 x 14 mils on a side.

In order to do the full fast Hadamard transform (FHT), we have to perform a number of stages as indicated in the Figure 61. This figure shows the interconnection of four typical sets of stages for the full transform. Figure 62 indicates an entire fast Hadamard transform chip layout that we have configured. Each of the blocks labeled A, B, C, D, E and F represent, for example, 1/4 of those operations shown on the previous figure. This entire diagram has been conceived as being implemented on a single CCD chip. It can accept up to 10-bit quantized input words and produces up to 13-bit quantized output words. In

CCD VOICE PROCESSOR CHIP CONFIGURATION

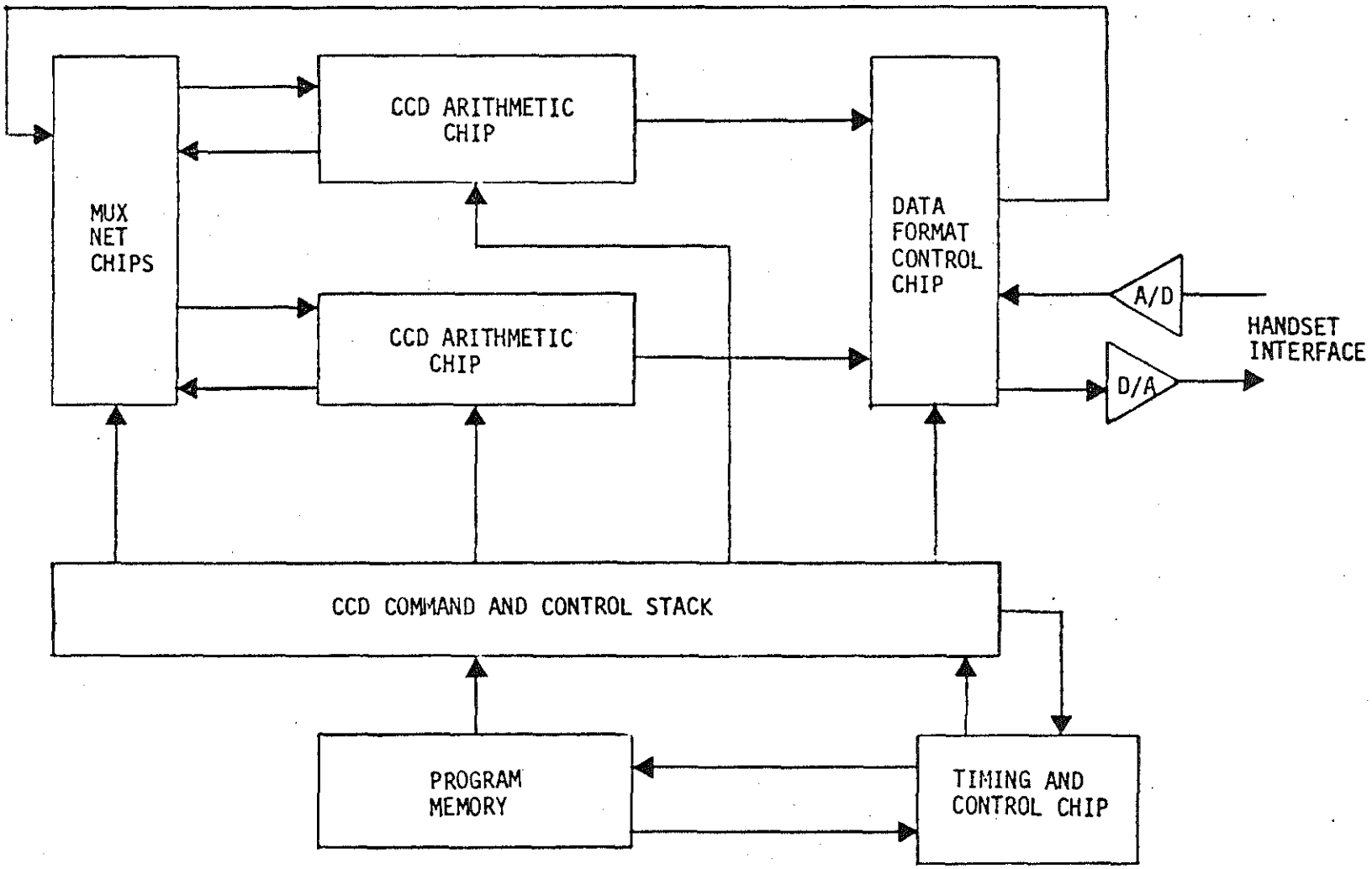


FIGURE 53

THREE DIGITAL CCD CHIPS PERFORM 80 PERCENT OF ALL PROCESSOR OPERATIONS

TRW

- ARITHMETIC UNIT CHIP
  - CONTAINS: (16 X 16) BIT MULTIPLIER; (32 + 32) BIT ADDER; MULTIPLEXING AND CONTROL GATES
  - ESTIMATED CHIP SIZE: $(290)^2$ MILS^2
  - ESTIMATED POWER: 1 WATT AT 5 MHZ
- COMMAND AND CONTROL STACK CHIP
  - CONTAINS: SHIFT REGISTER MEMORY; LOAD CONTROL
  - ESTIMATED CHIP SIZE: $(90)^2$ MILS^2
  - ESTIMATED POWER: 200 MW AT 5 MHZ
- DATA FORMAT AND CONTROL CHIP
  - CONTAINS: SHIFT REGISTER MEMORY; DATA REFORMATTER
  - ESTIMATED CHIP SIZE: $(95)^2$ MILS^2
  - ESTIMATED POWER: 200 MW AT 5 MHZ

FIGURE 54

ARITHMETIC UNIT CHIP

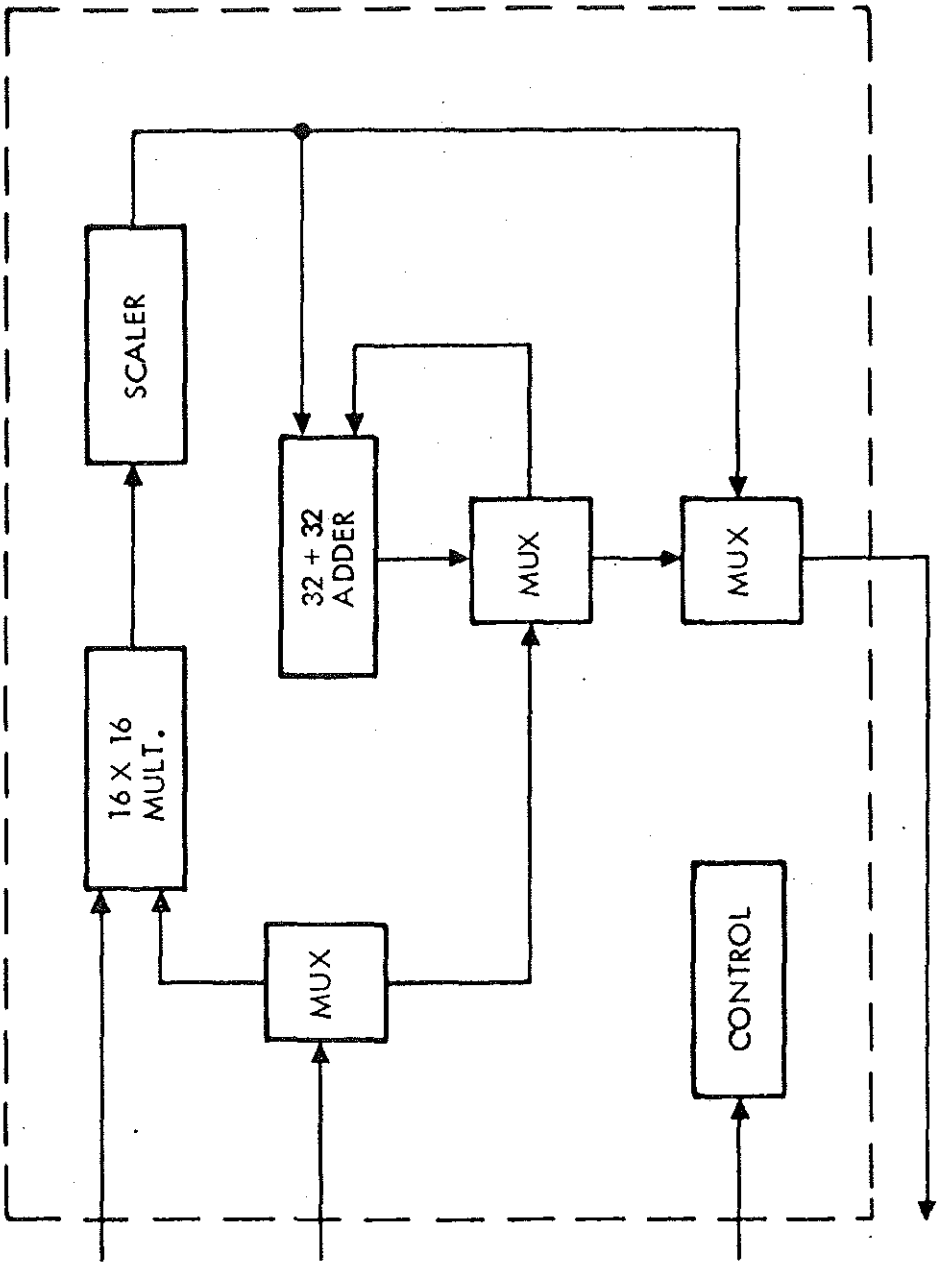


FIGURE 55

DATA FORMAT AND CONTROL CHIP

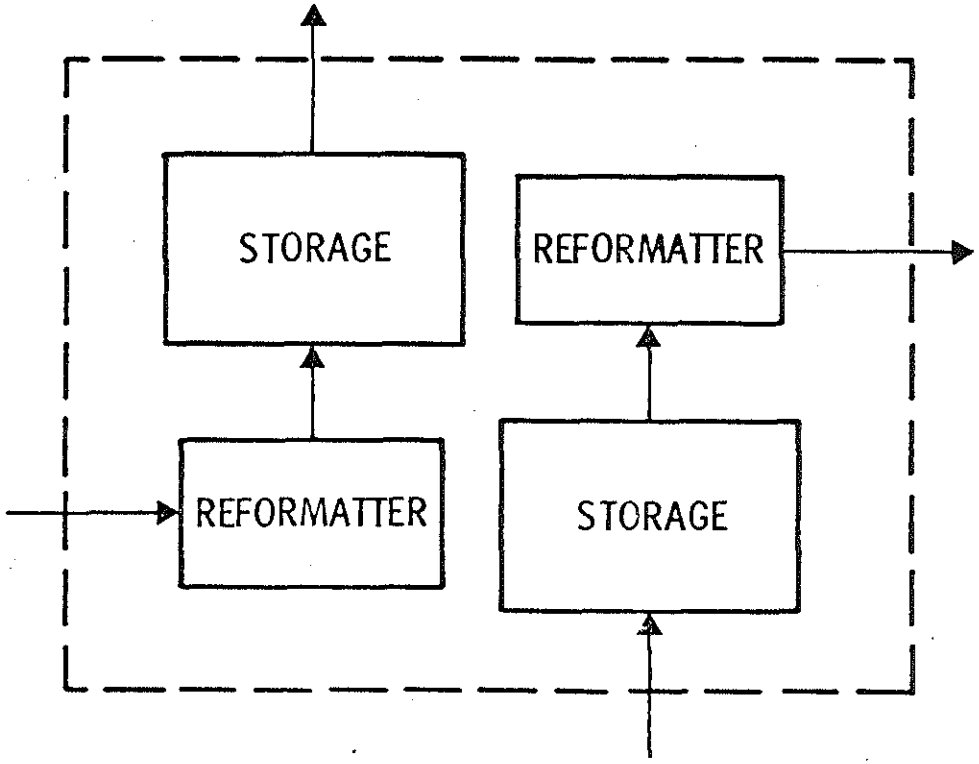


FIGURE 56

COMMAND AND CONTROL STACK CHIP

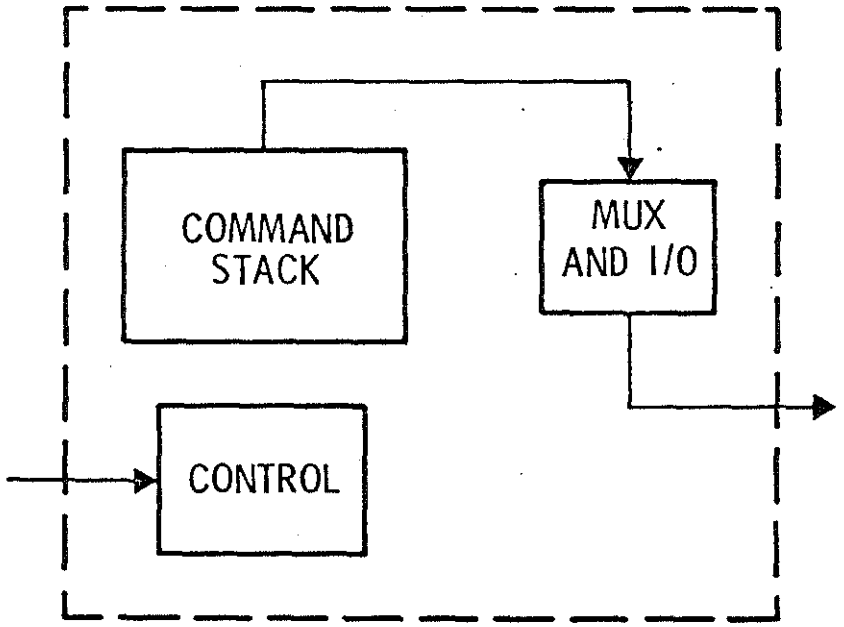
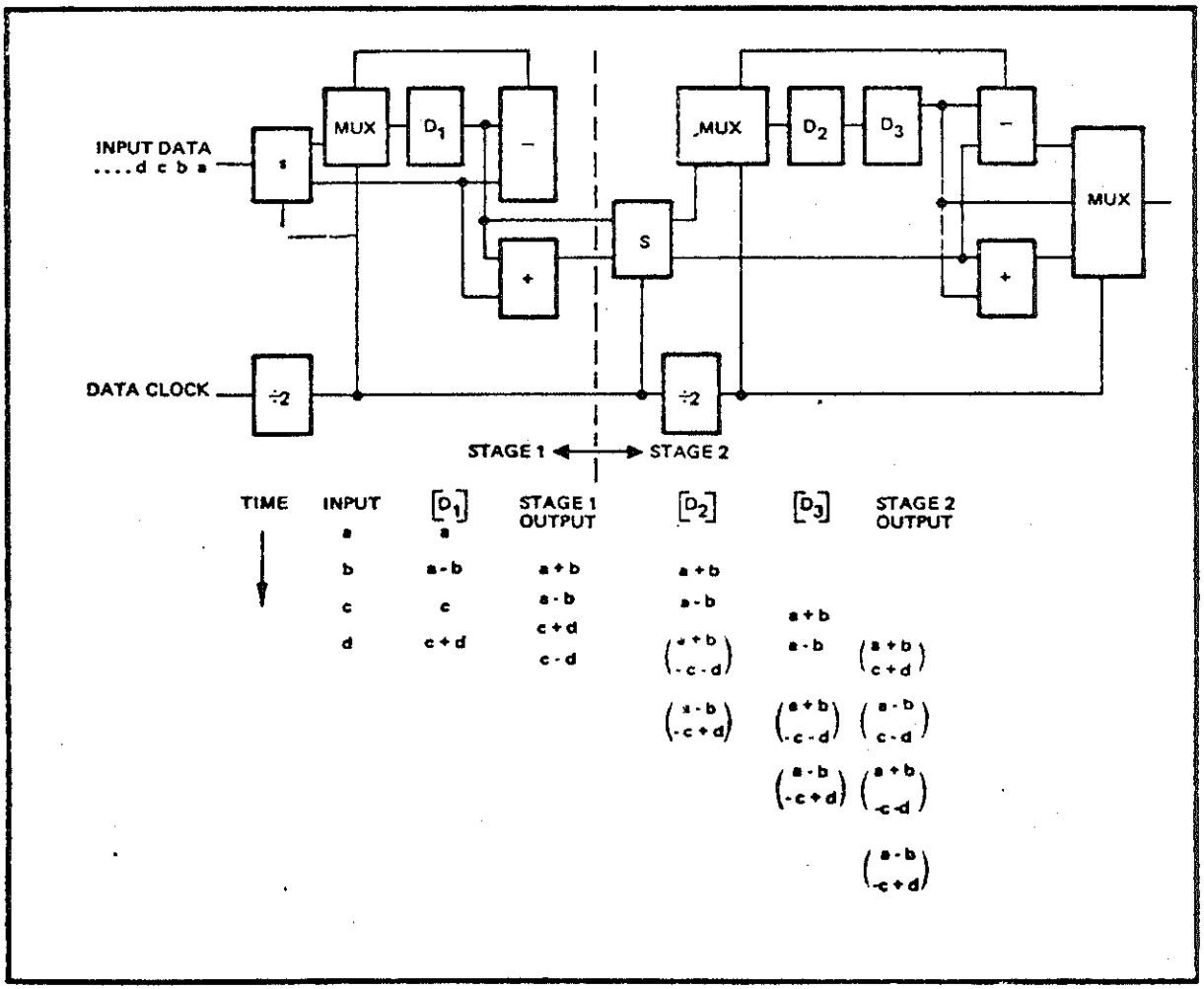
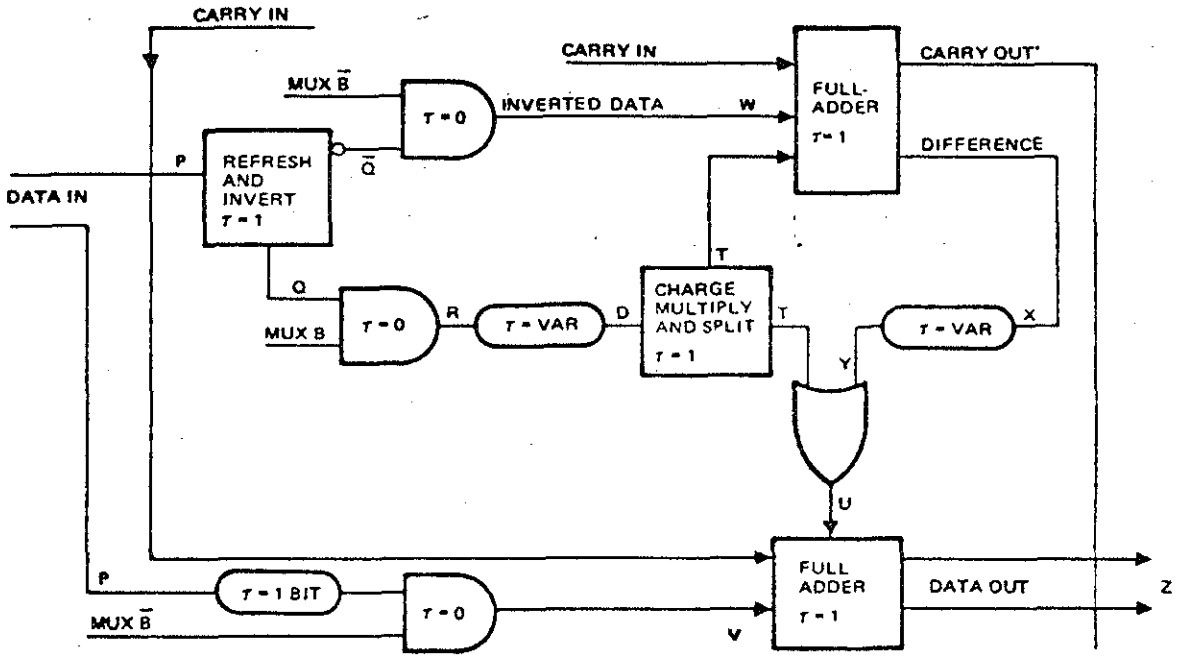


FIGURE 57



EXAMPLE OF THE PIPELINE HADAMARD TRANSFORM

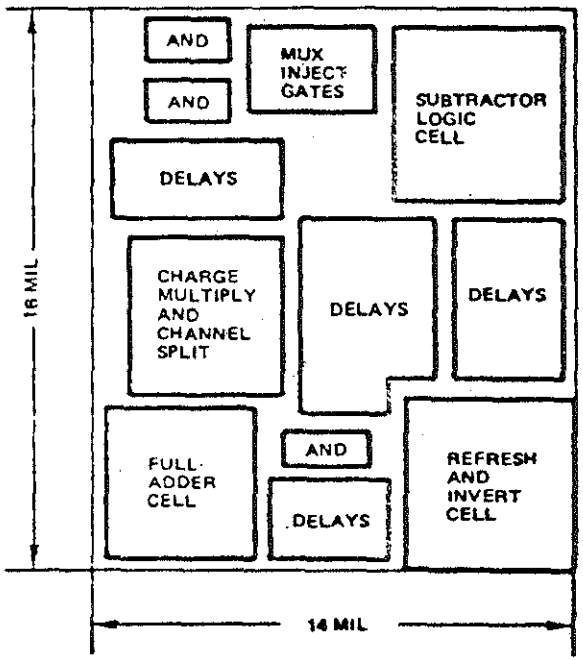
FIGURE 58



DCCD Implementation of a Typical FHT Cell.

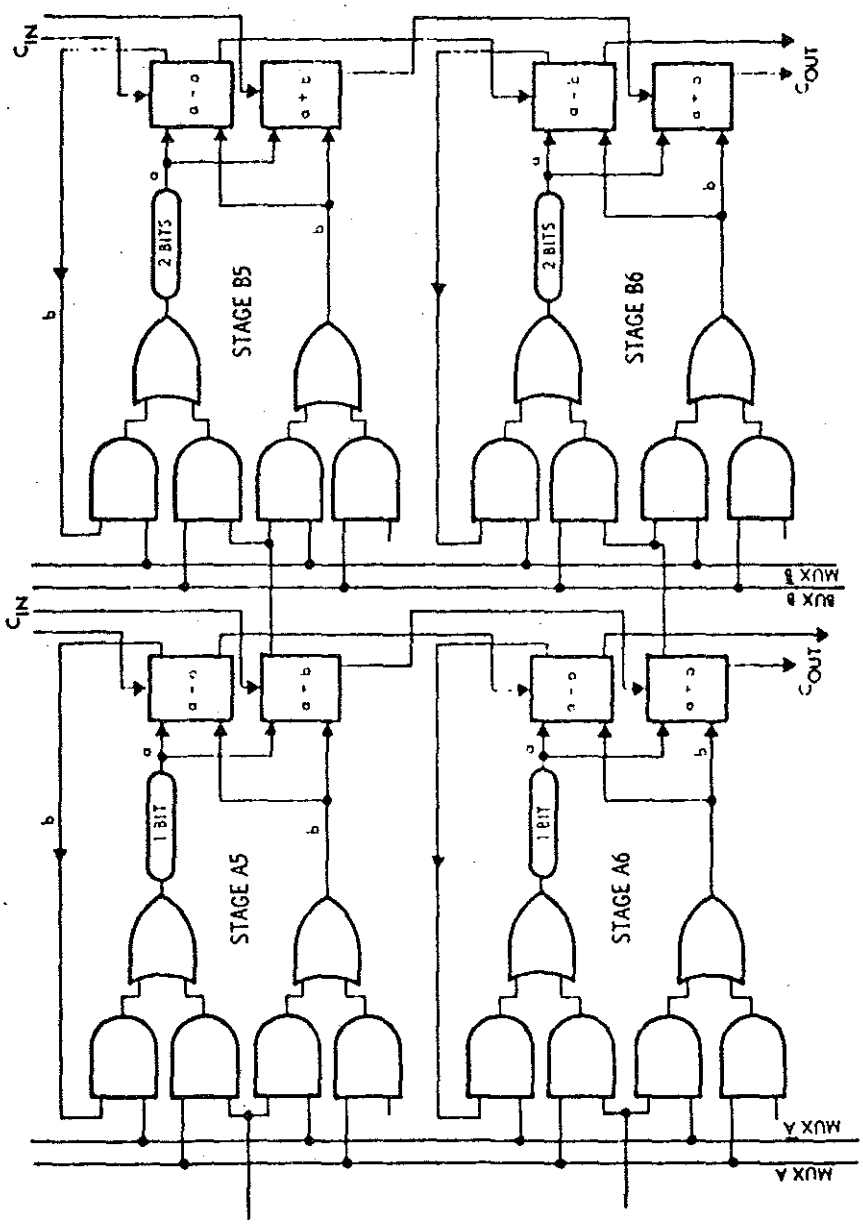
For a cell in the A stage, τ = 0, for the B stage τ = 1 bit delay, for the C stage τ = 3-bit, for the D stage τ = 7 bits, etc.

FIGURE 59



LOGIC CELL LAYOUT OF A TYPICAL FIT CELL

FIGURE 60



Four Typical FHT Cells

FIGURE 61

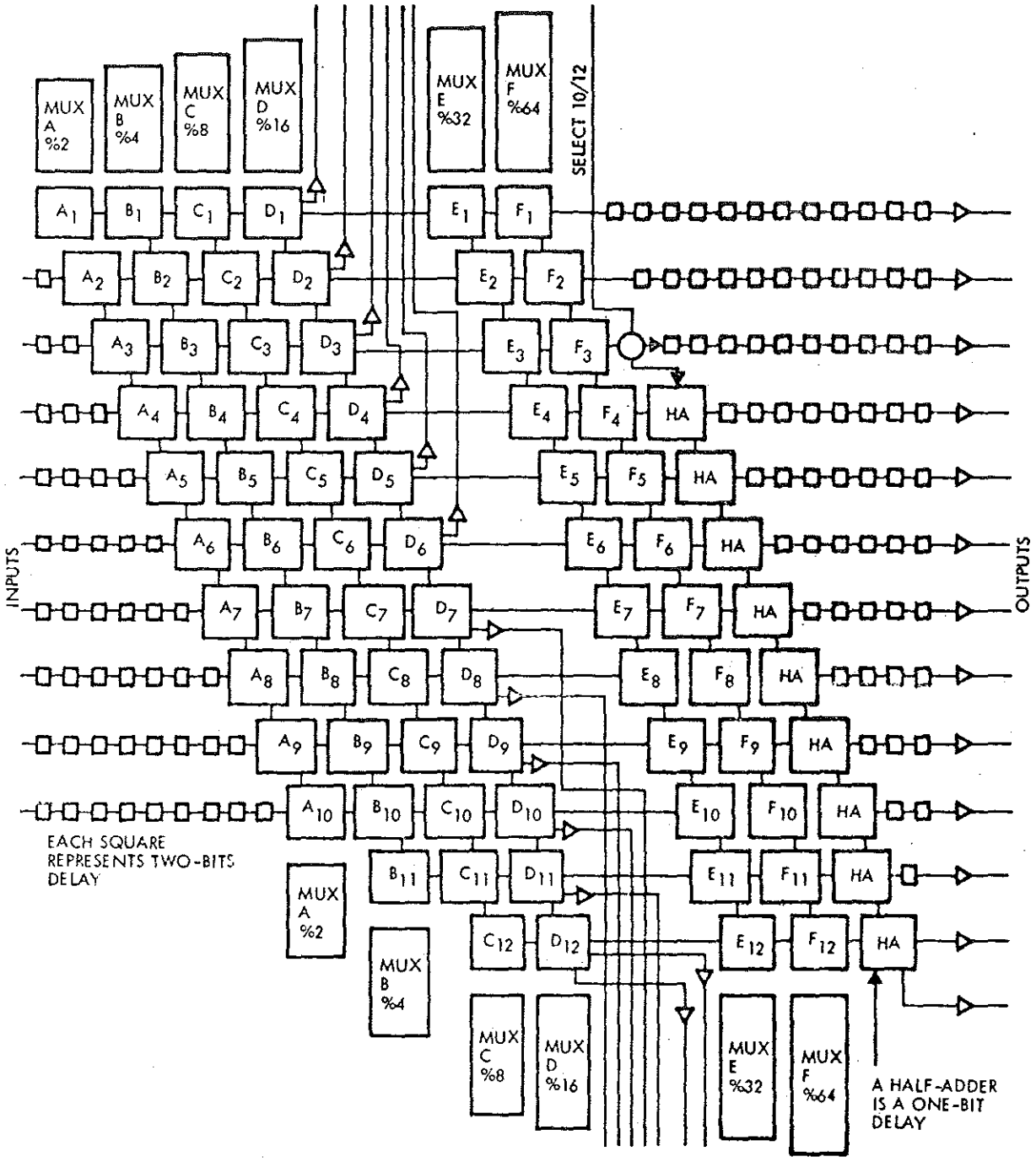


FIGURE 62

BLOCK DIAGRAM OF THE FAST HADAMARD TRANSFORM (FHT) CHIP

addition to the arithmetic operations, various multiplexing and division functions are performed by the peripheral circuits indicated at the top and bottom of the figure. One last feature is indicated here: a selectable output midway through the operation.

These previous examples give some idea of the complexity achievable with the digital CCD technology. Once again we should reiterate that it is ideally suited to pipeline operations, has a distinct advantage in low power and high density, and it is dependent only on circuit designs, not device processing.

Let us now make some observations relative to digital CCD memory. One question that repeatedly comes up is can we use tuned drivers to reduce the power of digital memories? Of course we can use turned drivers, but there are certain problems that result. Figure 63 summarizes the situation. If we use inductively tuned sine wave drivers, we consume the least power but it is very difficult to produce these in integrated circuit form and the control of the amplitude of the clock voltages is difficult. Switching to resistive sine wave drivers we find that we raise the power consumption slightly. We gain in that we can produce them in integrated form but we still have difficulty controlling the amplitude. Resistive square wave drivers on the other hand, which are most commonly used, have the highest power consumption but are easily producible in integrated form and have the best amplitude control. Overall it appears that except for the most restrictive cases it is best to use resistive square wave drivers and accept the high power dissipation in order to gain the accuracy of amplitude control.

Figure 64 summarizes some of the power considerations and shows that as a rule of thumb the power required is about a micowatt per square mil of gate area at a megahertz. We know of course that the power is directly proportional to the frequency. It is desirable to operate any memory chip as at low frequency as possible consistant with leakage current requirements and the need to refresh the data. In Figure 65 we do some typical power consumption calculations and we see that the total power required by the drivers is roughly about 30 times the power required on the chip as was noted earlier.

In order to maximize the density of memory structures and to minimize the capacitance per bit (therefore minimizing the power) we have worked for some time with a self aligned structure known as the offset gate. Figure 66 summarizes the realization steps of this technique. It fundamentally amount to the interaction of several mask layers, each mask layer having a feature of length L and the final device having features less than L due to the interaction. This

CLOCK DRIVER CONSIDERATIONS

Type \ Comments	Power Consumption	Producible in Integrated Form	Amplitude Control
Inductively Tuned Sinewave	Lowest	No	Difficult
Resistive Sinewave	Intermediate	Yes	Difficult
Resistive Squarewave	Highest	Yes	Easy

FIGURE 63

MEMORY CHIP POWER CONSIDERATIONS

- POWER/BIT ON CHIP IS PROPORTIONAL TO CLOCK FREQUENCY (1 $\mu W/MIL^2$ at 1 MHz)
- CLOCK DRIVER POWER/BIT IS PROPORTIONAL TO CLOCK FREQUENCY AND CAPACITANCE/BIT
- CLOCK DRIVER POWER 1 BIT > ON CHIP POWER/BIT
- THEREFORE IT IS DESIRABLE TO OPERATE AT AS LOW A FREQUENCY AS POSSIBLE CONSISTENT WITH REGENERATION REQUIREMENTS

FIGURE 64

TYPICAL POWER CONSUMPTION

- On Chip Power/Bit = $f \cdot Q \cdot \Delta V$

  $= (100 \times 10^3 \text{ Hertz}) \cdot (10^5 \text{ electrons} \times 1.6 \times 10^{-19} \frac{\text{coul}}{\text{elec}})$ (5 volts)

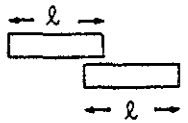
  $= 8 \times 10^{-9}$ watts

- Off Chip Power/Bit $\approx$ 30X (On Chip Power/Bit)
  - Clock Driver Power/Bit = $f \cdot C \cdot (\Delta V)^2 = (10^5 \text{ Hertz})\ (8 \times 100^{-14} f)$ (5 volts)^2

    $= 2 \times 10^{-7}$ watts
  - Controlling Logic
  - Power Supply Losses
- Total Power $\approx$ Off Chip Power

FIGURE 65

BASIC CCD STRUCTURE

- ACHIEVE HIGH DENSITY AND SELF ALIGNED DEVICES BY USING THE INTERACTION AMONG SEVERAL MASK LAYERS
- CAN PRODUCE DEVICES WITH STRUCTURAL DETAIL THAT IS FINER THAN MASK DETAIL
- 

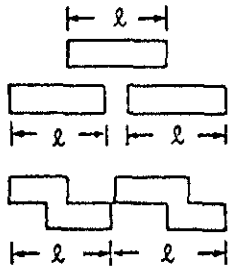
REGION OF OVERLAP $< \ell$
- ESSENTIALS: 1) CHARGE STORAGE CAPABILITY, 2) CHARGE FLOW DIRECTIONALITY CONTROL
- 

FIGURE 66

allows us to achieve a number of desirable features. This device however, is strictly a two phase device as can be seen on the bottom of Figure 66.

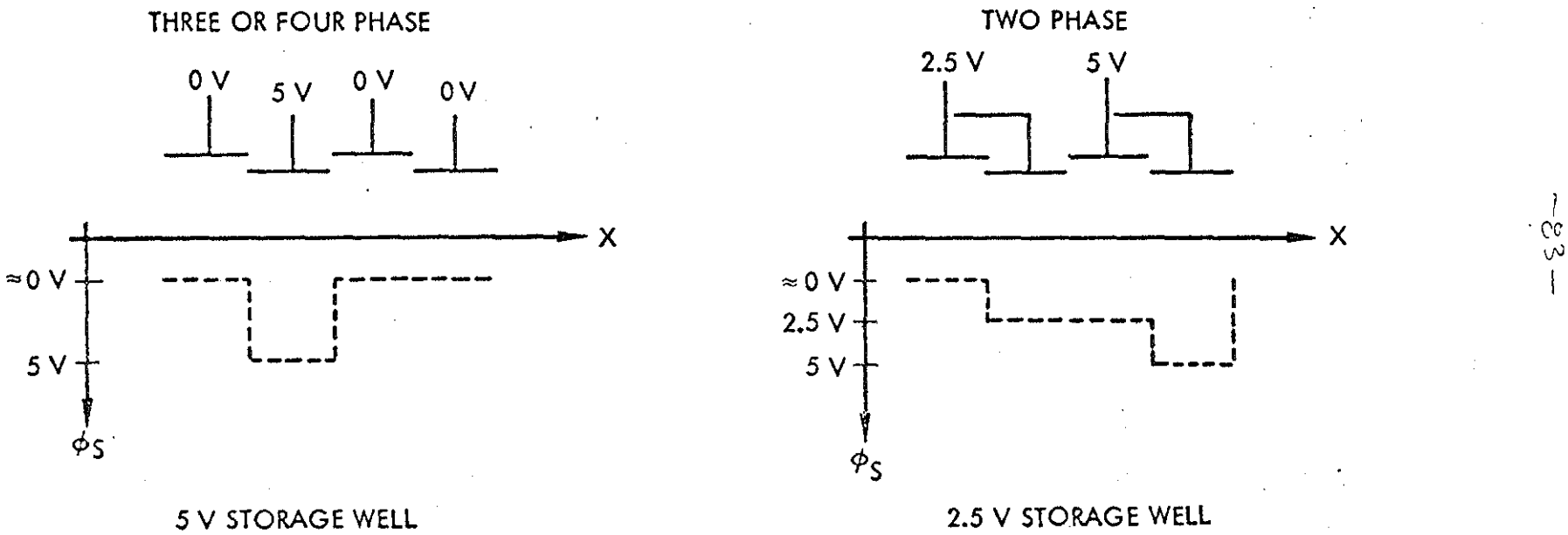
Figure 67 shows that we can achieve minimum supply voltages if we use a three or four phase operation compared to a two phase operation. This figure assumes zero volt threshold but sill illustrates the point well. Larger storage can be achieved with smaller voltage swings for three or four phase operations. However this consideration is often over ridden by the advantages of density achievable with the offet gate.

Figure 68 shows a cross section view of the length required for one bit of a standard two phase structure in part A and a high density offset gate structure in part B. This is displayed by the graphical relationship shown on the right hand portion of the figure. Note that the actual area required for a standard realization is dependent upon the ratio of the minimum feature size, L, to the alignment tolerances, r, that one is willing to live with.

Figure 69 summarizes this statement and indicates the relationship between the chip size and chip capacity for a particular r and L ratio. Note the advantage that the offset gate structure has over the standard structure. This advantage gets larger as the shift register capacity increases.

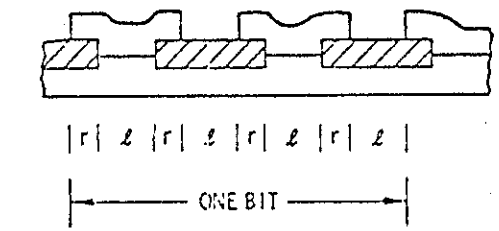
In general, CCD memory structures are made from a serial-parallel-serial shift register organization. This structure is shown in Figure 70. Note that the data arrives and is shifted along a serial register. When this register is full, a parallel transfer is performed and the data is inserted into the parallel registers. Data is continually read into the input register. When the parallel registers are filled, the data is then shifted to the output register and read out in a serial fashion. This allows us to achieve several things. The total storage capacity of this structure is approximately $N_s \times N_p$ bits. However, the total number of transfers is merly $N_s + N_p$ bits. In this way we maximize the storage while minimizing the number of transfers. This allows us to overcome one of the draw backs of charge transfer devices and that is the transfer loss per bit. One other important feature is realized with this structure: the majority of the bits are shifting at a much slower rate than the input. In fact, if the input frequency is F than most of the bits are shifting at F/N_s . That is the frequency of the parallel portion of the register. Thus from our previous comments we see that thepower required by this structure is significantly reduced.

THREE OR FOUR PHASE OPERATION IS BETTER FOR MINIMUM SUPPLY VOLTAGES

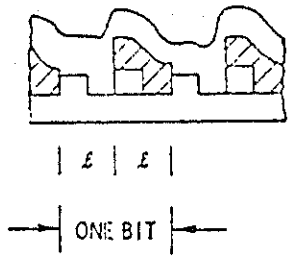


(ASSUMING ZERO VOLT THRESHOLD)

FIGURE 67

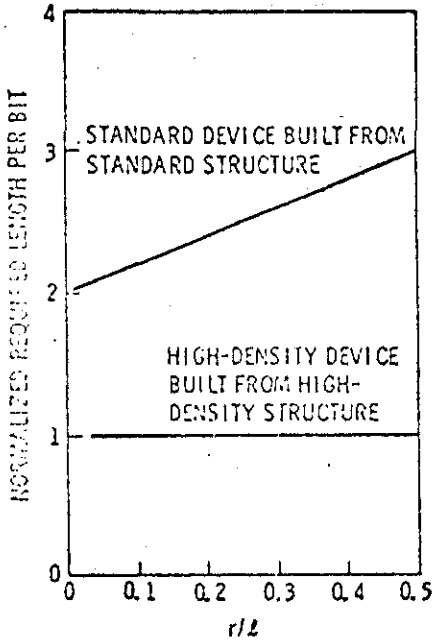


a) STANDARD GATE STRUCTURE



b) HIGH-DENSITY GATE STRUCTURE

CROSS-SECTIONAL VIEWS OF STANDARD AND HIGH DENSITY GATE STRUCTURE



NORMALIZED REQUIRED LENGTH PER BIT AS A FUNCTION OF (r/ℓ)

FIGURE 68

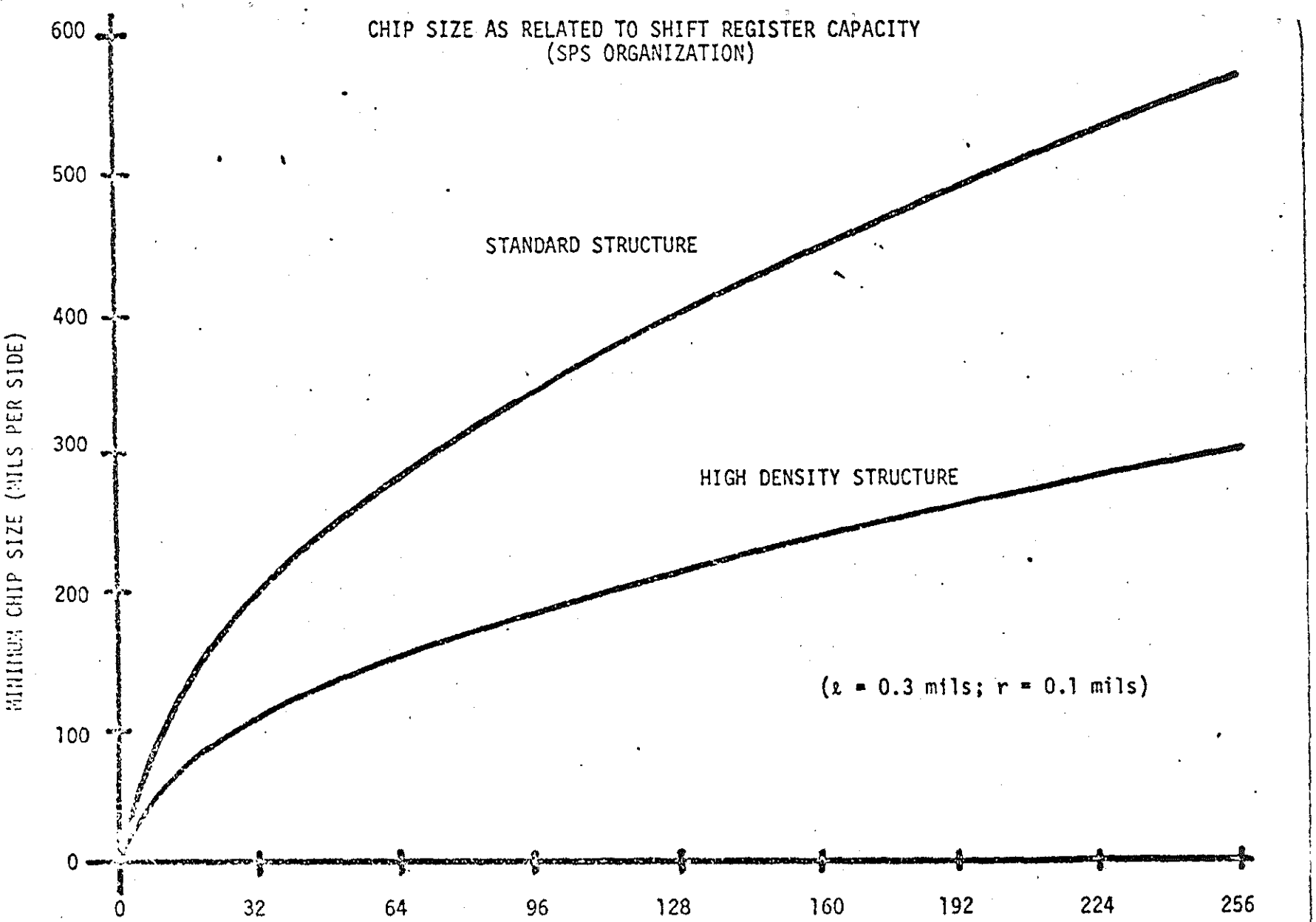
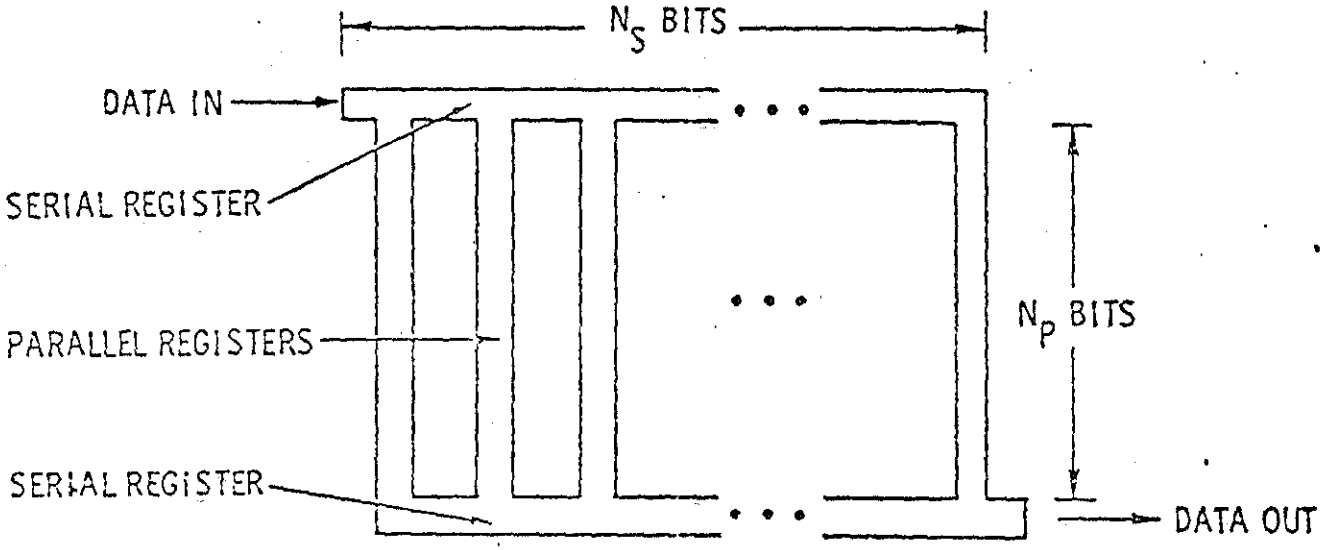

CHIP SIZE AS RELATED TO SHIFT REGISTER CAPACITY
(SPS ORGANIZATION)
600
500
400
300
200
100
0
MINIMUM CHIP SIZE (MILS PER SIDE)
STANDARD STRUCTURE
HIGH DENSITY STRUCTURE
(ℓ = 0.3 mils; r = 0.1 mils)
0
32
64
96
128
160
192
224
256
SHIFT REGISTER CAPACITY (KILOBITS)

FIGURE 69



ORGANIZATION OF A SERIAL-PARALLEL-SERIAL SHIFT REGISTER

FIGURE 70

Figure 71 shows a memory realized with such a structure. This figure is a 16K bit SPS memory realized with 5 micron photolithography, thereby producing bit densities over 6-bits per square mil of silicon area.

When we consider shift register operation we must be worried about charge transfer loss and its implication for bit error rate. Figure 72 indicates the four regions that we need to be concerned with when we look at losses. There is the perpendicular edge labeled 1 in this figure. That is the edge of the gate that is in line with the charge flow. The parallel edge, labeled 2 in the figure, is the edge perpendicular to the charge flow. That edge may be more significant than any other in determining the overall loss for small structures; proportionally the parallel edge begins to assume a large fraction of the area of the total storage region.

The effect of these edges depends a great deal upon the fast surface states associated with the device structure. Fast surface states are a name applied to trapping centers in the silicon. Figure 73 indicates how these traps can affect the operation of the device. On the left hand side we see the conduction and valiance band and a quasi-Fermi level where we have all states from the conduction band to that Fermi level filled under the full well conditions. On the right hand side of the figure we see a corresponding diagram for an empty well case where the distence between the conduction band and the quasi-Fermi level is much less. Now then we see that there is a region of state labeled II in that figure which can be covered and uncovered during each device cycle. It is precisely that density of state which will affect our operation and which will cause losses to our transfer. To wind up this section on memories, let us do some calculations on different effects of noise on the memory device. Figure 74 shows a listing for some assumed baseline conditions for a long shift register. We have assumed such items as surface state density and the number of carriers input for 1's and 0's. This example is for a shift register of 1000 transfers. The following Figure 75 indicates the various noise contribution under the assumed baseline conditions of Figure 74. The figure that follows, shows the trend in the bit error rate as the fast surface states the number of transfers the storage well size and so on are all varied about their nominal values. We see that certain elements have a rather strong affect on the error rate. Note that one of the least sensitive parameters would be absolute temperature.

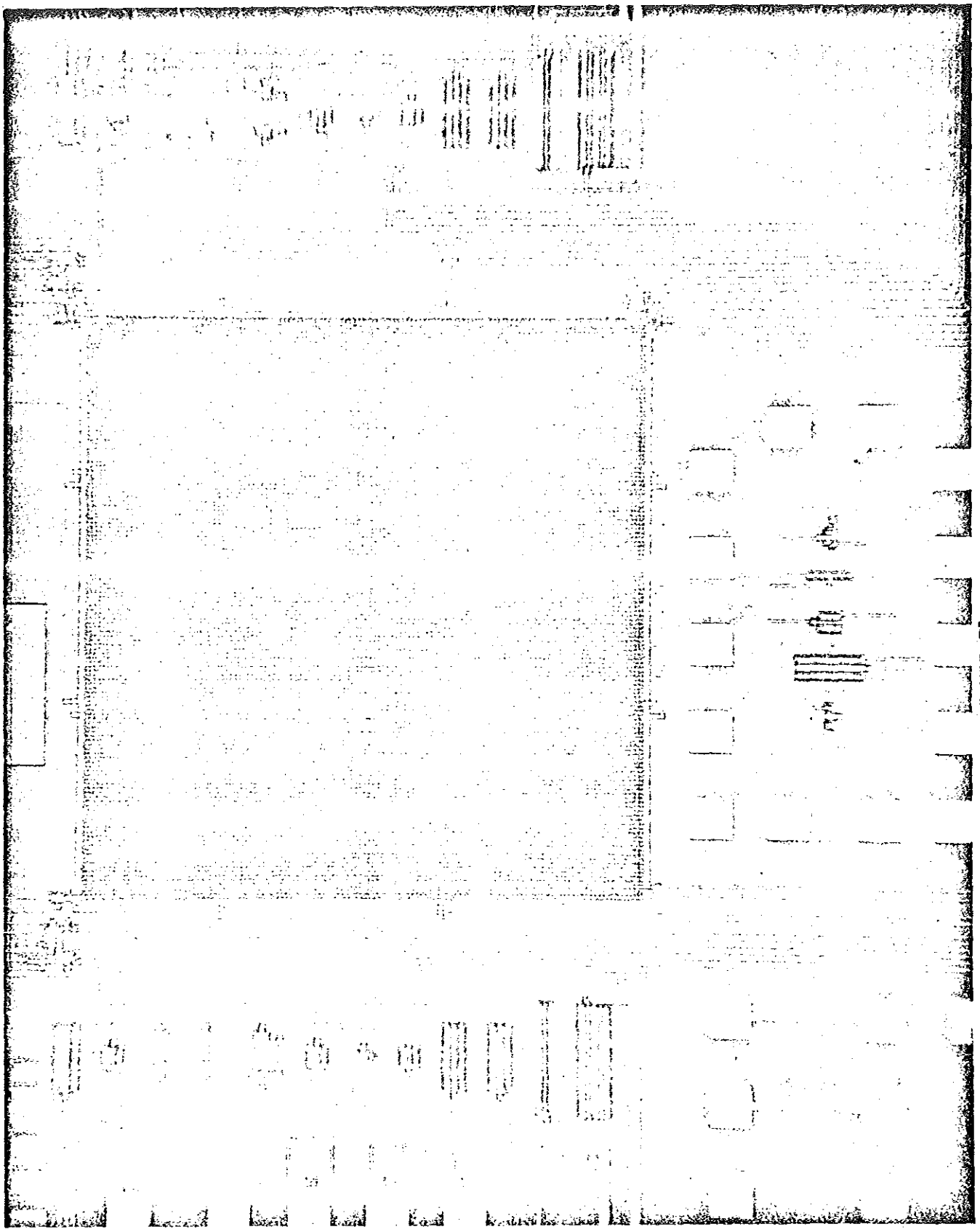
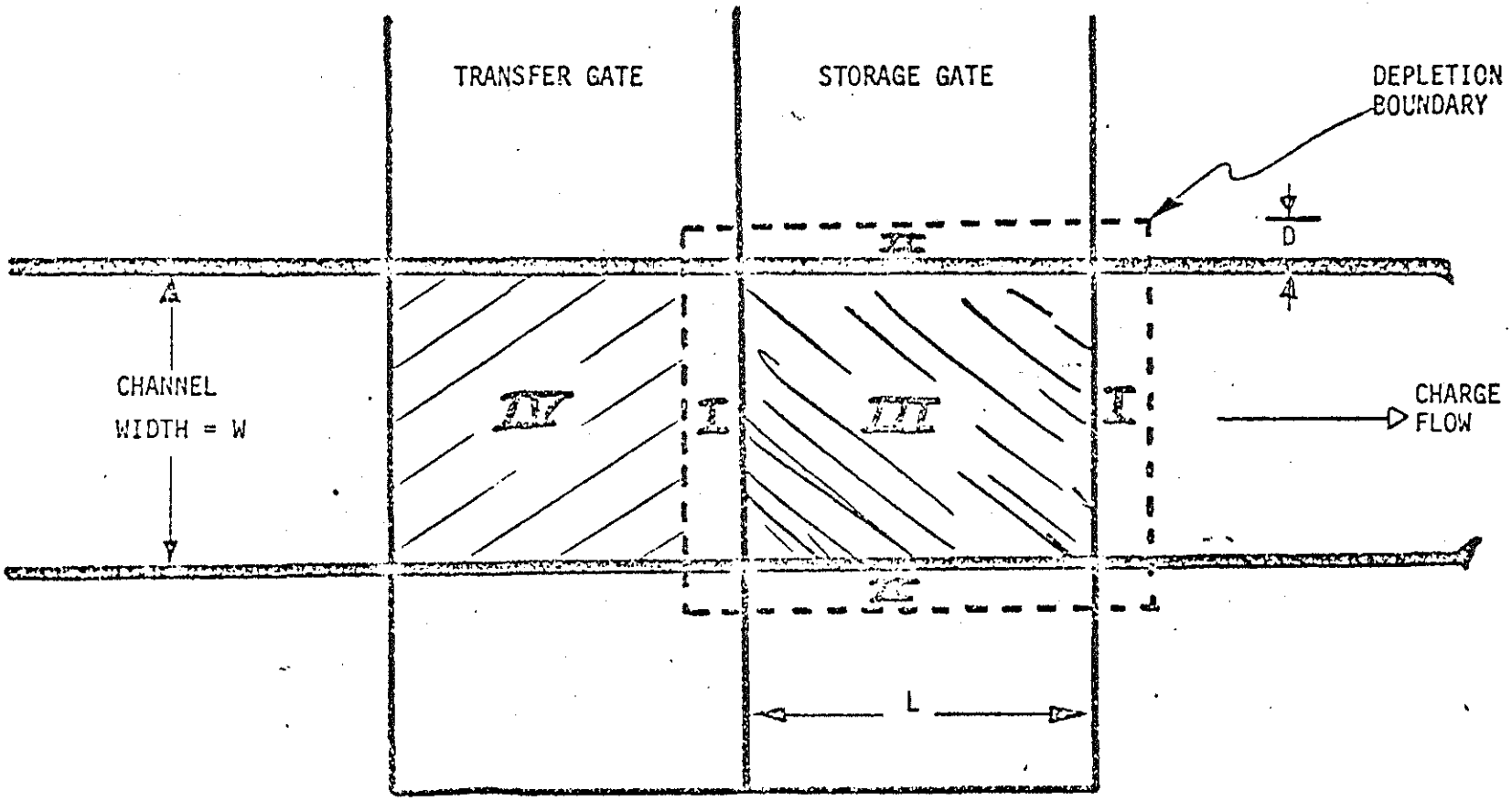


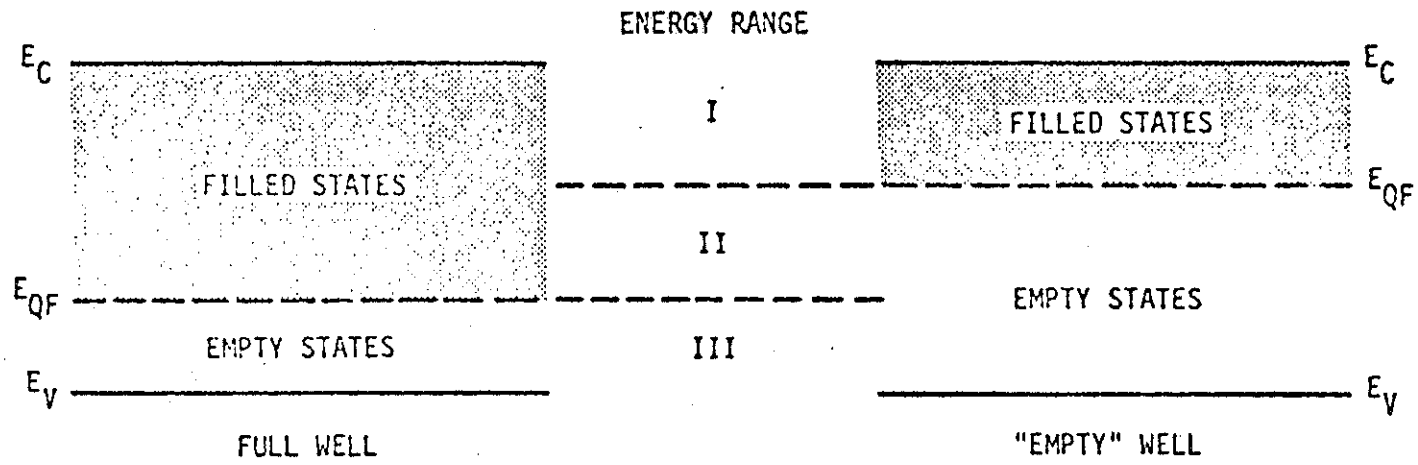
FIGURE 71

REGIONS OF POSSIBLE CHARGE LOSS DUE TO N_{ss} RELATED EFFECTS



I PERPENDICULAR EDGE
II PARALLEL EDGE
III STORAGE GATE
IV TRANSFER GATE

FIGURE 72



E_V — VALANCE BAND EDGE
E_C — CONDUCTION BAND EDGE
E_{QF} — QUASI-FERMI LEVEL

Hypothentical Example of Storage Well Quasi-Fermi Level Position for "Empty" and Full Well

FIGURE 73

ASSUMED BASELINE CONDITIONS

SURFACE STATE DENSITY	10^{10} $(CM^2\text{-}EV)^{-1}$
NUMBER OF CARRIERS INPUT FOR ONE	13672
NUMBER OF CARRIERS INPUT FOR ZERO	6836
STORAGE WELL DEPTH	5 VOLTS
NUMBER OF TRANSFERS	1000
INEFFICIENCY PER TRANSFER	10^{-4}
BANDWIDTH OF OUTPUT AMPLIFIER	10^6 HZ
TRANSCONDUCTANCE OF OUTPUT MOSFET	300 MICROMHOS
PHOTOLITHOGRAPHY	5 MICRONS
TEMPERATURE	300^oK

FIGURE 74

NOISE CONTRIBUTIONS UNDER BASELINE CONDITIONS

NOISE SOURCE	ELECTRONS RMS
FAST SURFACE STATES	301
TRANSFER PROCESS	357
FAT ZERO	14
FLOATING DIFFUSION	44
MOSFET	2
LEAKAGE CURRENT	37
TOTAL NOISE	471
BIT ERROR RATE	1.9×10^{-7}

FIGURE 75

One way to overcome all of these errors is to include on the memory chip error correction coding. Figure 82 indicates some typical error correction codes and the improvement in bit error rate that can be achieved. This technique of including error correction coding on a memory chip will become mandatory as the number of bits per chip becomes larger and larger due to the system demands.

BIT ERROR RATE AS A FUNCTION OF SURFACE STATE DENSITY

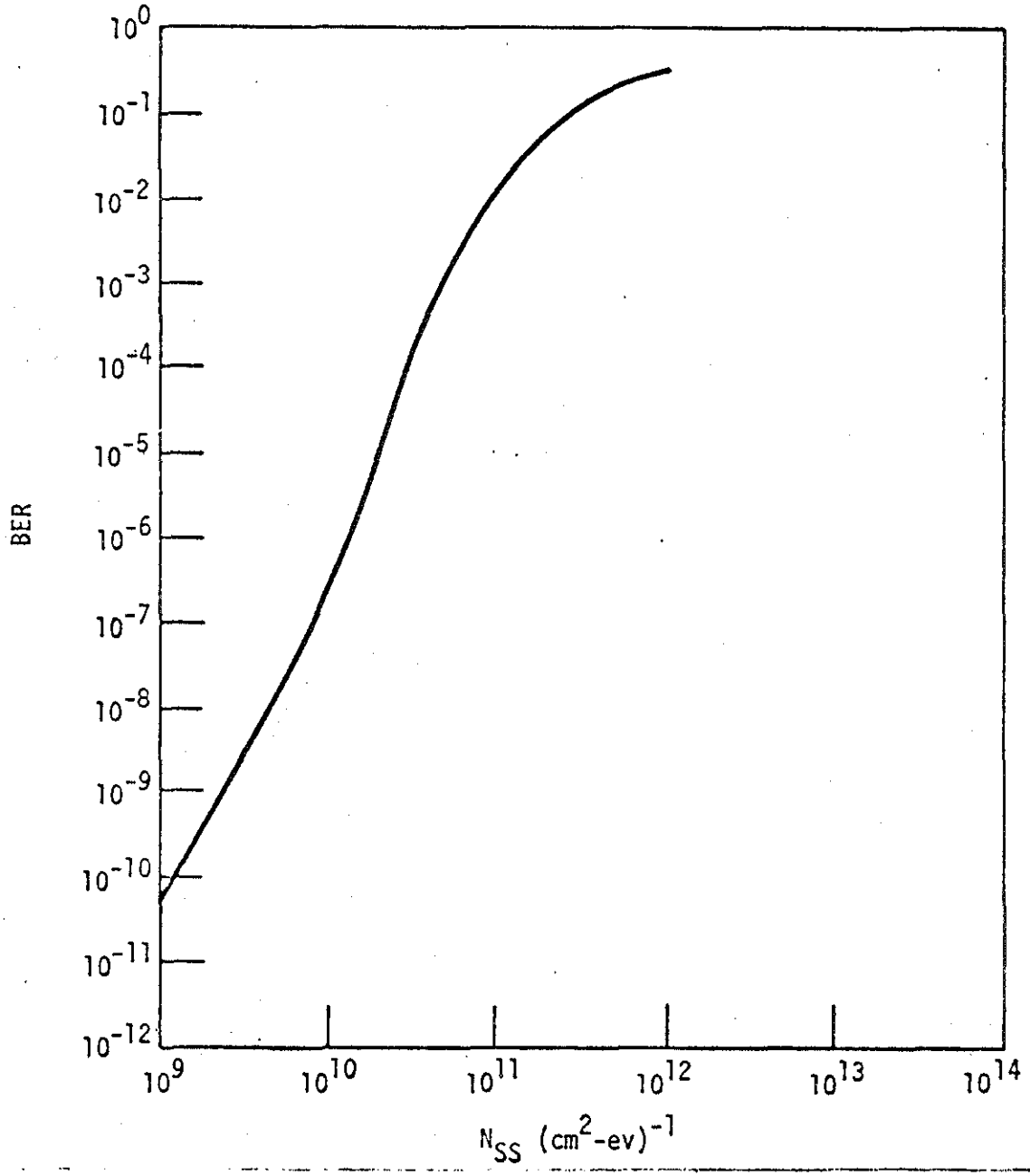


FIGURE 76

B IT ERROR RATE AS A FUNCTION OF THE NUMBER OF TRANSFERS

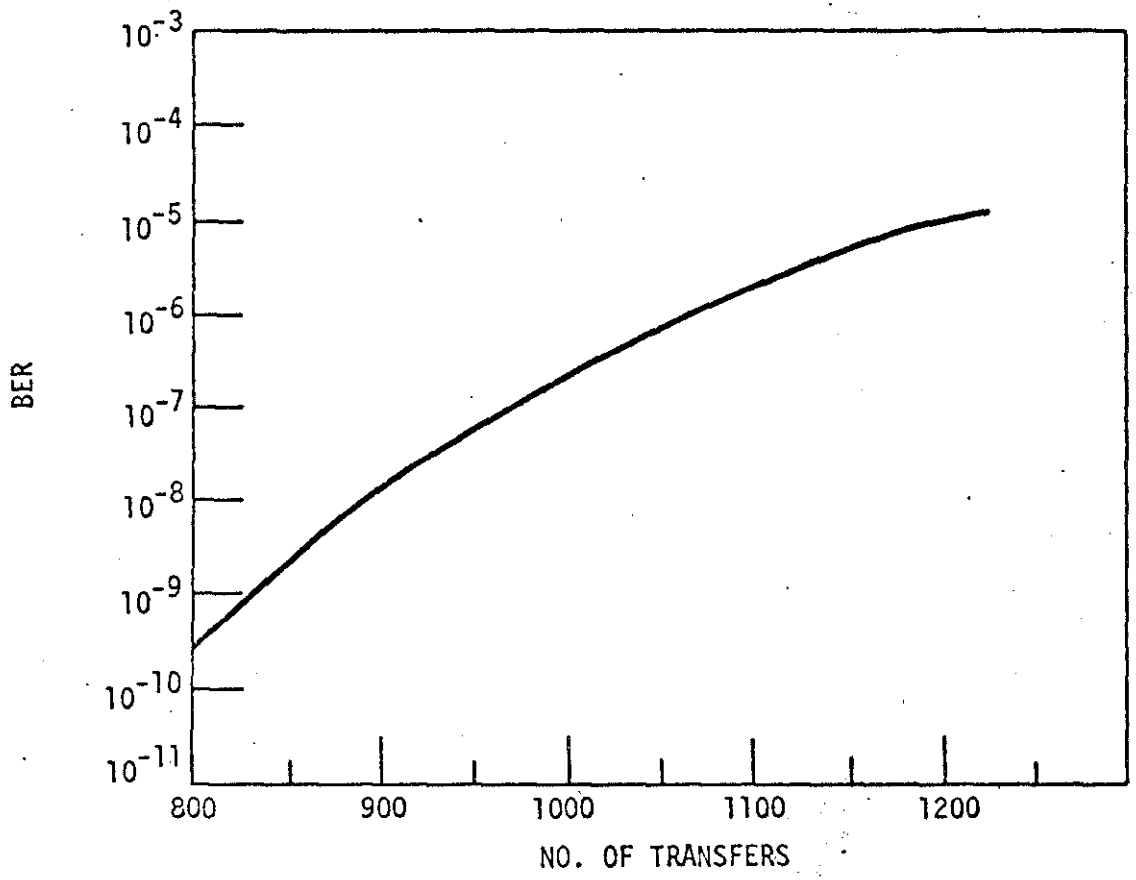


FIGURE 77

BIT ERROR RATE AS A FUNCTION OF STORAGE WELL DEPTH

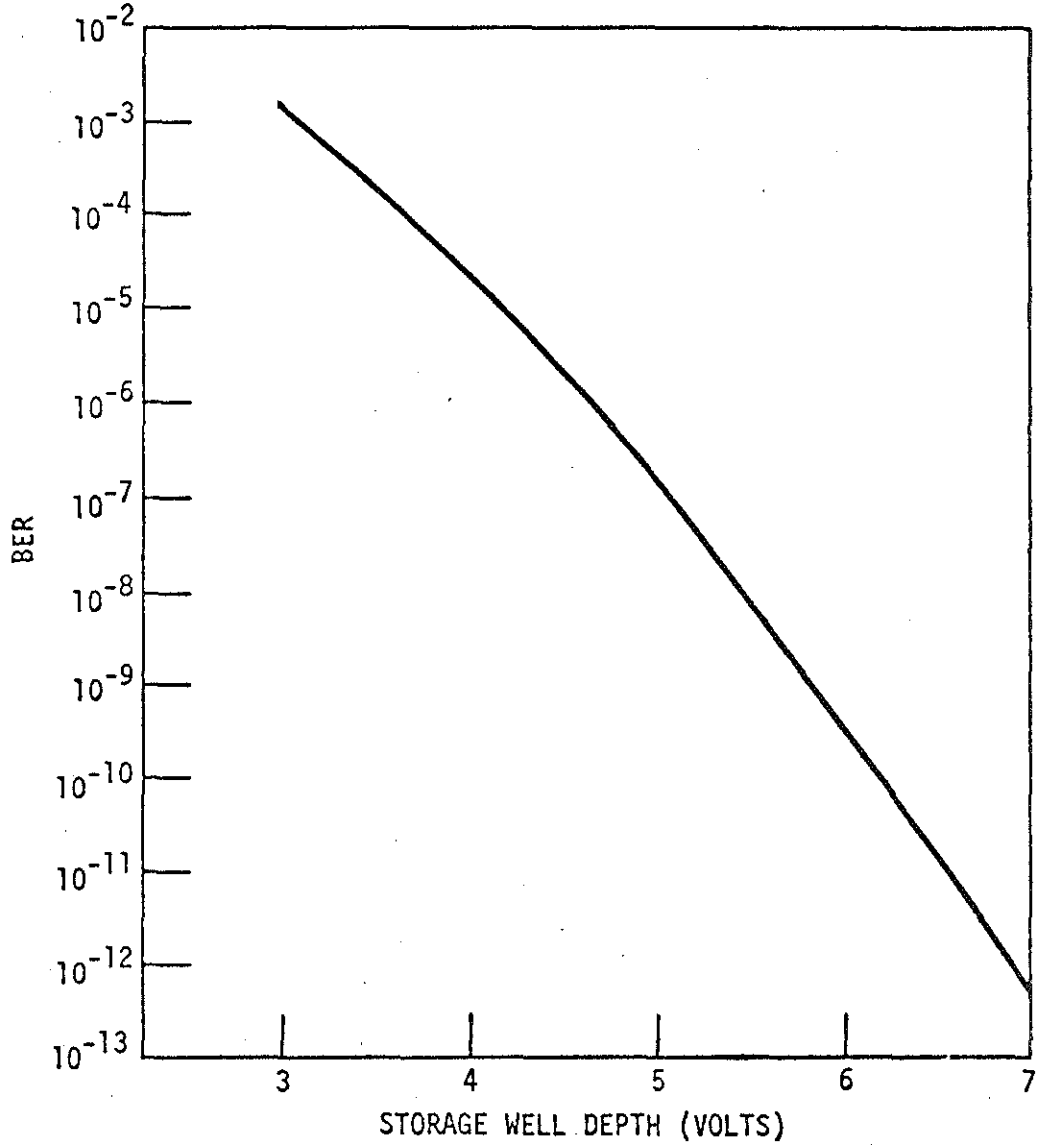


FIGURE 78

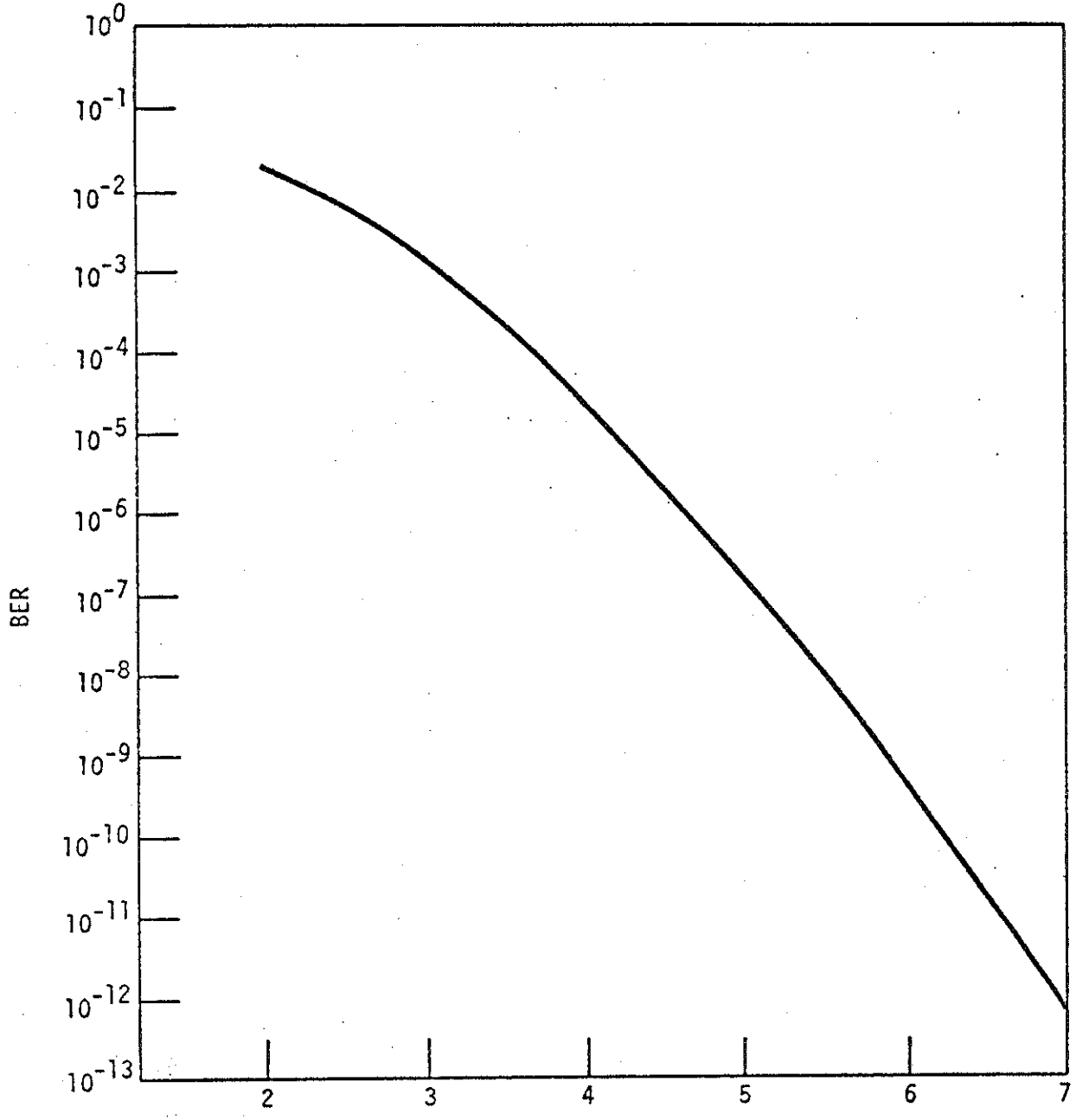

BER
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
10^{-9}
10^{-10}
10^{-11}
10^{-12}
10^{-13}
2
3
4
5
6
7
PHOTOLITHOGRAPHY (MICRONS)

FIGURE 79

BIT ERROR RATE AS A FUNCTION OF THE ONE/ZERO CARRIER RATIO

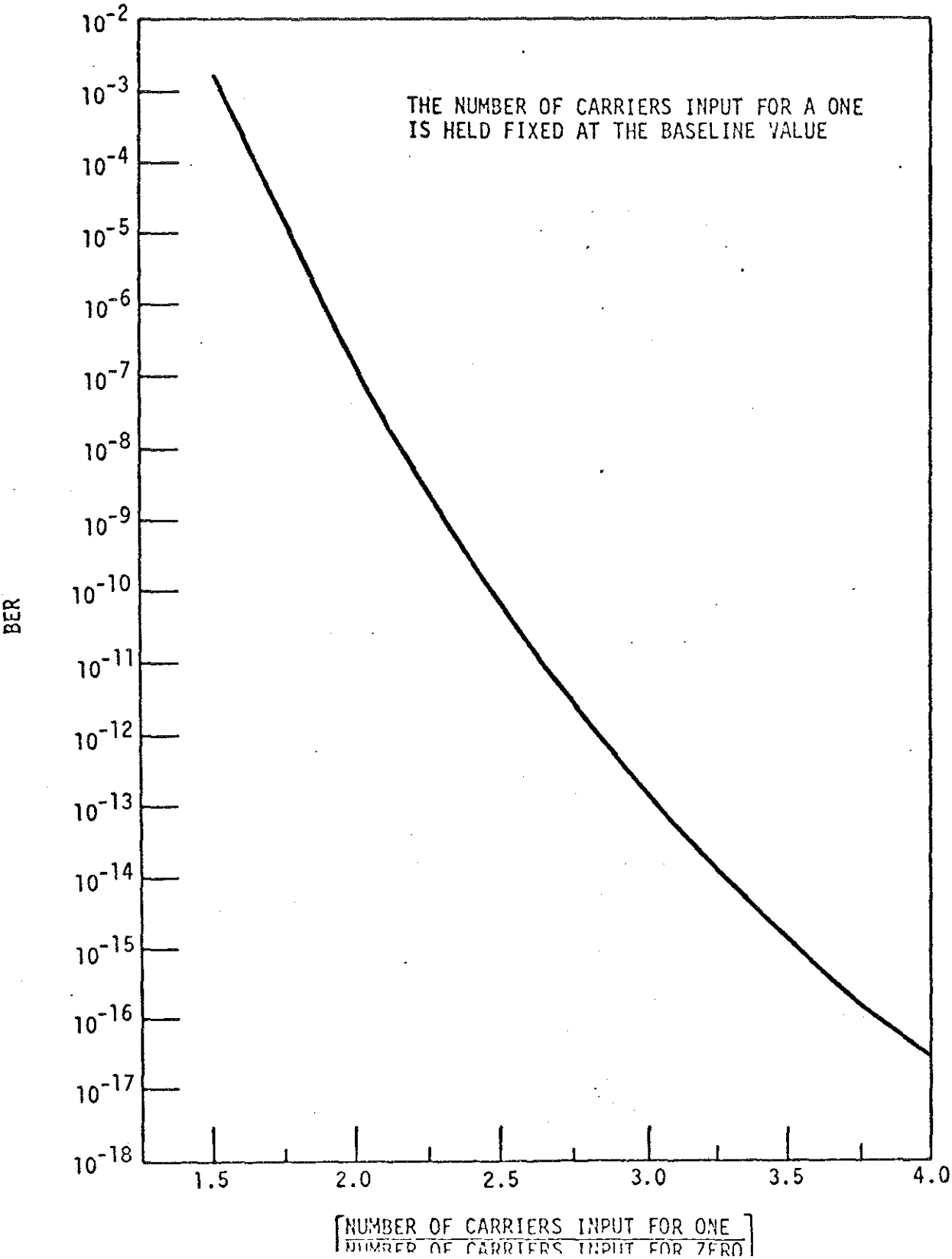


FIGURE 80

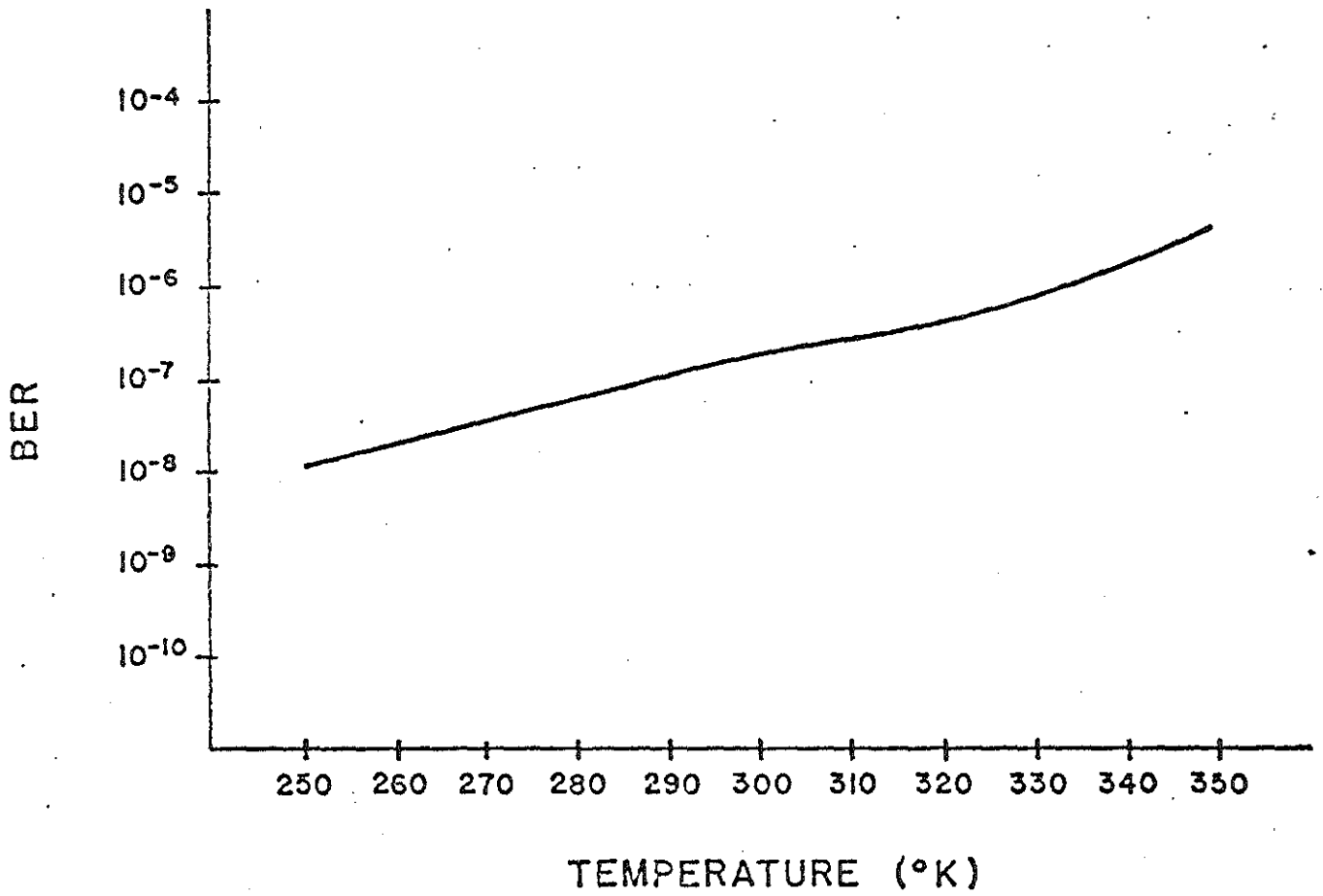

BER
10-4
10-5
10-6
10-7
10-8
10-9
10-10
250 260 270 280 290 300 310 320 330 340 350
TEMPERATURE (°K)

FIGURE 81

BIT ERROR RATE AS A FUNCTION OF CODE TYPE

CODE	TYPE	NUMBER OF CORRECTABLE ERRORS	BIT ERROR RATE		
NONE	---	0	10^{-4}	10^{-5}	10^{-7}
(15,11)	HAMMING	1	2×10^{-7}	2×10^{-9}	2×10^{-13}
(31,26)	HAMMING	1	4.6×10^{-7}	4×10^{-9}	3×10^{-13}
(31,21)	BCH	2	7.5×10^{-10}	10^{-12}	
(60,48)	SHORTENED BCH	2	2.5×10^{-9}	2.5×10^{-12}	
(63,51)	BCH	2	3×10^{-9}	3×10^{-12}	3×10^{-15}

FIGURE 82