N70-29986
TMX-53983

NASA TECHNICAL MEMORANDUM

NASA TM X-53983

FAST CARRY ACCUMULATOR DESIGN

By William C. Mastin
Astrionics Laboratory

February 9, 1970

NASA

George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO. TM X-53983	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.

4. TITLE AND SUBTITLE

Fast Carry Accumulator Design

5. REPORT DATE

February 9, 1970

6. PERFORMING ORGANIZATION CODE

7. AUTHOR(S)

William C. Mastin

8. PERFORMING ORGANIZATION REPORT #

9. PERFORMING ORGANIZATION NAME AND ADDRESS

George G. Marshall Space Flight Center
Marshall Space Flight Center, Alabama 35812

10. WORK UNIT NO.

11. CONTRACT OR GRANT NO.

13. TYPE OF REPORT & PERIOD COVERED

Technical Memorandum

12. SPONSORING AGENCY NAME AND ADDRESS

14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

Prepared by Astrionics Laboratory , Science and Engineering Directorate

16. ABSTRACT

Methods for increasing the speed of binary addition by decreasing carry propagation time are reviewed. Particular attention is given to those methods that would be applicable to an accumulator using ones complements with end-around carry for subtraction. An iterative accumulator using pulse logic for input and carry signals is described. Realizations of gated carry, carry-completion detection, and carry-skip circuits that would be compatible with this accumulator are presented. NAND gates are used in the design of the required combinational networks.

17. KEY WORDS

Digital Accumulation
Gated Carry
Carry-Completion Detection
Carry Skip

18. DISTRIBUTION STATEMENT

19. SECURITY CLASSIF. (of this report)	20. SECURITY CLASSIF. (of this page)	21. NO. OF PAGES	22. PRICE
Unclassified	Unclassified	73	$3.00

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Concluded)

DEFINITION OF SYMBOLS

Symbol	Definition
+	Logical OR function
·	Logical AND function
–	Inversion
⊕	Logical EXCLUSIVE OR function
⊙	Logical COINCIDENCE function
AC	Accumulator register element; T flip-flop
DM	Delay multivibrator
	NAND gate
	NOR gate
Ⓓ	Delay

TECHNICAL MEMORANDUM X-53983

FAST CARRY ACCUMULATOR DESIGN

SUMMARY

Methods for increasing the speed of binary addition by decreasing carry propagation time are reviewed. Particular attention is given to those methods that would be applicable to an accumulator using ones complements with end-around carry for subtraction. An iterative accumulator using pulse logic for input and carry signals is described. Realizations of gated carry, carry-completion detection, and carry-skip circuits that would be compatible with this accumulator are presented. NAND gates are used in the design of the required combinational networks.

INTRODUCTION

Digital techniques are finding application in closed-loop control systems. Sampling permits the control of large amounts of power by sensitive control elements and decreases the loading on sensing devices. The use of sampled data and digital components in control systems facilitates time sharing of portions of the system, which results in both economy and reliability in design. A digital compensator in a control loop is shown in Figure 1, where D(z) represents the transfer function of the compensator, $G_a(s)$ is the transfer function of the system sensing element, and $G_b(s)$ is the transfer function of the system driving element.

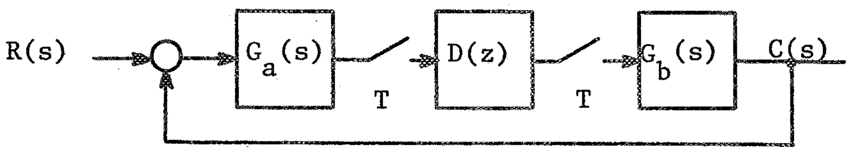


Figure 1. Closed-loop system with sampling [1].

A model of a digital compensator is shown in Figure 2 [1]. A second-order compensator transfer function may be described by [2]

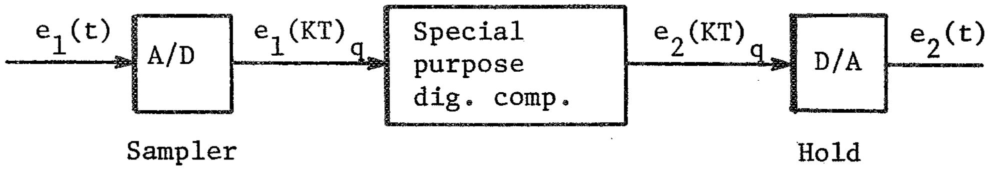


Figure 2. Digital compensator [1].

$$e_2(KT) = e_1(KT) + a_1e_1(KT-T) + a_2e_1(KT-2T) - b_1e_2(KT-T)$$
$$-b_2e_2(KT-2T)$$

where T is the sampling period.

The accuracy with which this equation actually describes the transfer function may be seen to depend upon the speed with which the computer performs the indicated arithmetic operations. If τ is the time required to complete the computation, the output of the compensator is a function of $KT + \tau$. This τ corresponds to a lag in a continuous system and, if large enough, could contribute to stability problems.

Adder circuitry is essential not only for addition, but also for multiplication, subtraction, and division. The speed of the computational process is then heavily dependent upon that of the adder. In general, the basic design problem for arithmetic units is that of achieving high speed and accuracy at low cost. Size, weight, and sensitivity to environment are not generally of great importance in a general purpose computer. These factors, along with a greater emphasis on reliability, do become of great concern in the special purpose computers used in digital compensators.

The most economical adder is probably the serial machine represented in Figure 3. Only one full binary adder (FBA) unit is required to sum the addend and augend words. A typical FBA stage is shown in Figure 4. However, only one element of each word is added in one computation time so the time for the addition of the two words (of equal length) is then nT where n is the number of elements in a word and T is the time required for each addition.

The time required for the addition of the two words can be decreased by the use of a pseudoparallel or simple iterated adder as represented in Figure 5. Here the elements of the n-bit addend and augend are applied to n FBA's instantaneously, and all partial sums $(x_i \oplus y_i)$ are formed during one

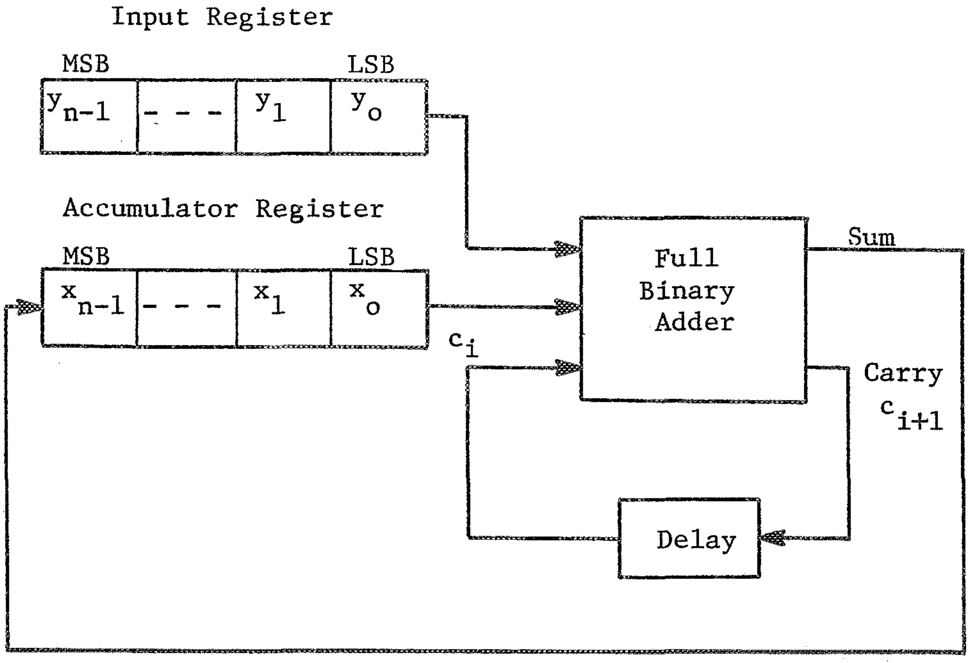


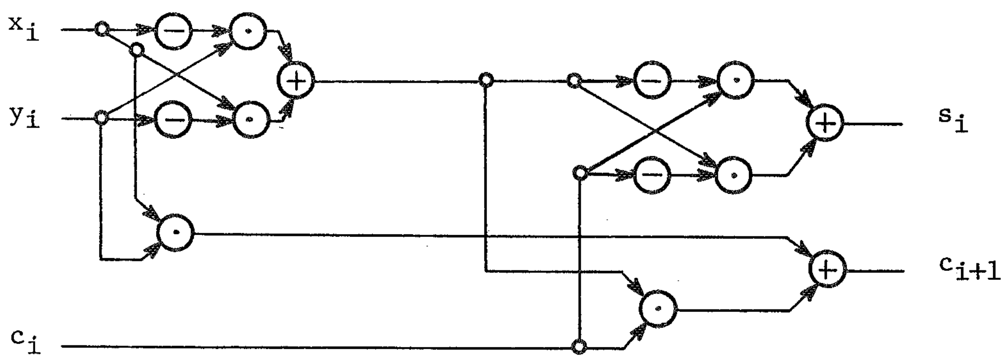
Figure 3. Simple series accumulator.

addition time T. The addition is not completed, however, until all carries generated have been added to the partial sums in the succeeding stages. This carry propagation is essentially a serial process, and in the worst case a carry generated in the first stage would have to propagate through the entire length of the adder.

It is thus recognized that the problem of increasing the speed of the addition process reduces, to a great extent, to that of decreasing carry propagation time. A considerable amount of work has been done in this area and several solutions have been proposed.

REVIEW OF METHODS FOR DECREASING CARRY PROPAGATION TIME

Several methods for increasing the speed of binary addition by decreasing carry propagation time have been presented by Sklansky [3], MacSorely [4], and Lehman and Burla [5]. Those methods and others that might be applicable to accumulators using ones complements with end-around carry for subtraction will be reviewed here.



TRUTH TABLE				
x_i	y_i	c_i	s_i	c_{i+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

s_i

$c_i \backslash x_i y_i$	00	01	11	10
0		1		1
1	1		1	

c_{i+1}

$c_i \backslash x_i y_i$	00	01	11	10
0			1	
1		1	1	1

$$\begin{aligned} s_i &= c_i \overline{x}_i \overline{y}_i + \overline{c}_i \overline{x}_i y_i \\ &\quad + c_i x_i y_i + \overline{c}_i x_i \overline{y}_i \\ &= c_i \oplus x_i \oplus y_i \end{aligned}$$

$$\begin{aligned} c_{i+1} &= c_i x_i + x_i y_i \\ &\quad + c_i y_i \\ &= c_i \overline{x}_i y_i + c_i x_i \overline{y}_i \\ &\quad + c_i x_i y_i + \overline{c}_i x_i y_i \\ &= c_i (x_i \oplus y_i) + x_i y_i \end{aligned}$$

Figure 4. Typical full binary adder stage.

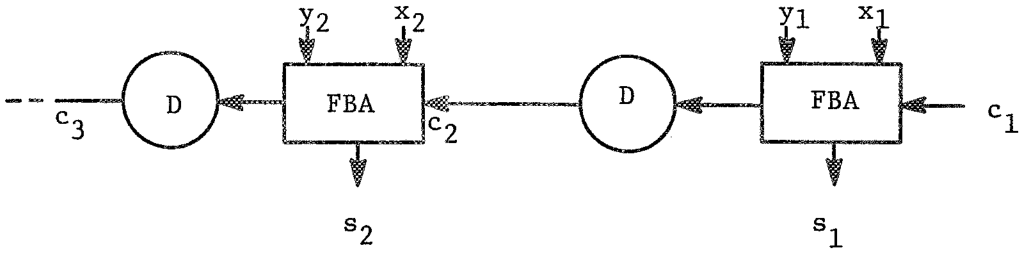


Figure 5. Simple iterated (pseudoparallel) adder.

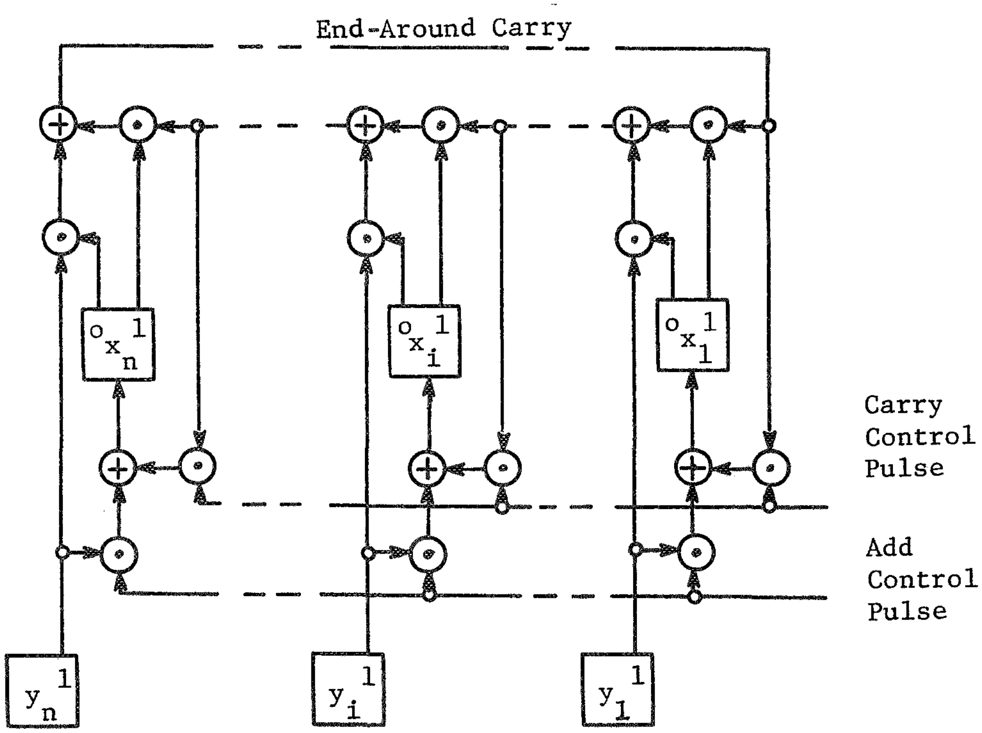
The truth table for carry determination for two-summand binary addition is given in Table 1.

TABLE 1. TRUTH TABLE FOR CARRY DETERMINATION

c_i	x_i	y_i	c_{i+1}	
0	0	1	0	Carry out determined solely by carry in.
0	1	0	0	
1	0	1	1	
1	1	0	1	
0	0	0	0	Carry out determined solely by the summands, x_i , y_i .
1	0	0	0	
0	1	1	1	
1	1	1	1	

Gated Carry

Perhaps the simplest method for decreasing carry propagation time is that of gated carry. Table 1 [6] shows that the dependency of the carry out of a stage will fall into two categories; one in which the carry out is determined entirely by the carry in and one in which the carry out is determined only by the summands. Unlike summand bits in a stage may be detected and used to provide a path for carries from its preceding stage directly to its succeeding stage. In a stage with like summands, the above path is interrupted, and a one carry or zero carry is provided for its succeeding stage, depending upon whether the summands are both ones or both zeros. One such scheme using a 2-bit time addition is shown in Figure 6 [7]. A variation of this method using storage elements for the carry produced in each stage is called the stored-carry technique and is shown in Figure 7 [7].

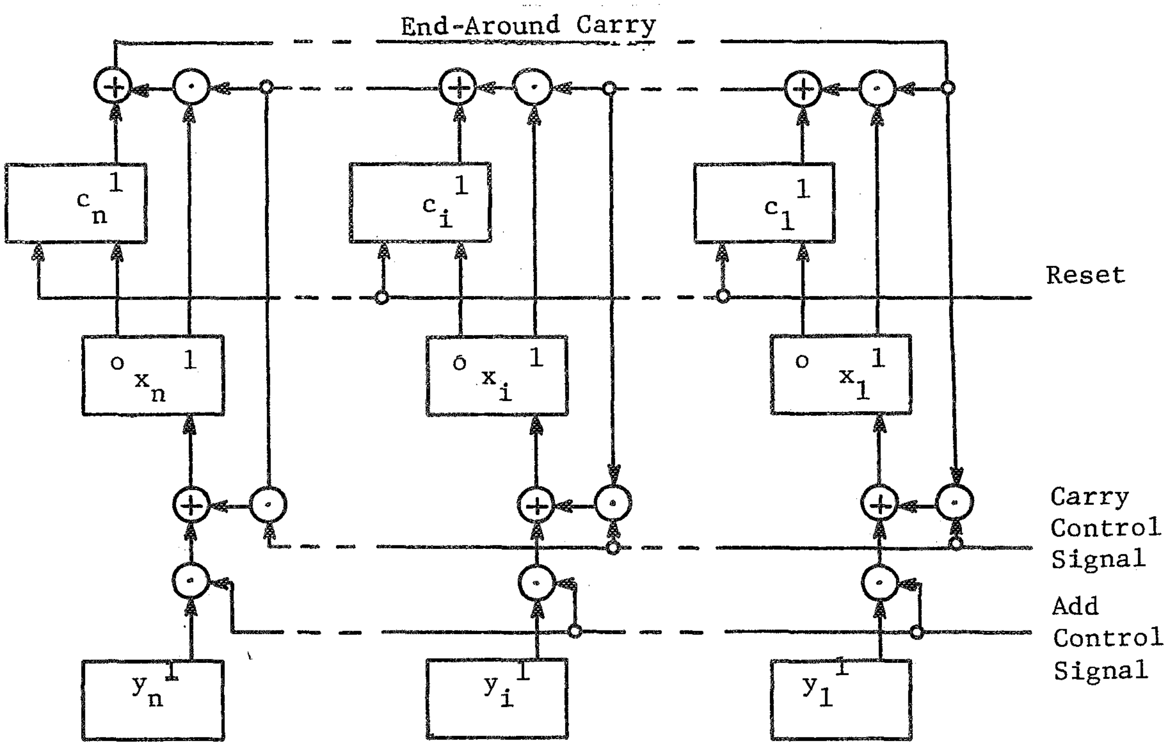


x_i - Element of Accumulator Register

y_i - Element of Input Register

Sum, X+Y, stored in Accumulator Register

Figure 6. Gated carry.



x_i - Element of Accumulator Register

y_i - Element of Input Register

c_i - Element of Carry Register

Sum, X+Y, stored in Accumulator Register

Figure 7. Stored carry.

In either of these techniques, a carry entering the first stage would pass through two gates per stage through the n-1 stage and through two gates in the n^{th} stage to be assimilated. The maximum carry propagation time is then 2(nd), where n is the number of stages and d is the delay time per gate.

Carry-Completion Detection

In methods using fixed-time addition, time for the worst case propagation through the entire accumulator plus a safety factor must be allowed. It can be shown that for the addition of numbers with a random distribution of ones and zeros, the average maximum carry length is less than $\log_2 n$, where n is the word length [8]. For the propagation of both carry and no carry, the average length is $\log_2 5n/4$ [9, 10]. The addition could then be speeded up if only the time needed for the actual carry propagation could be allowed. Gilchrist, Pomerene, and Wong [6] proposed a scheme to detect the presence of a carry and use its completion to initiate the next addition step. The principle of carry-completion detection is illustrated in Figure 8. At the initiation of a carry operation, a one is inserted into the zero-carry line preceding the first adder stage. A one carry or zero carry is then gated or generated in the succeeding stages as determined by the inputs at those stages. The condition for a one carry is

$$c_i^1 = x_i y_i + c_{i-1} \cdot (x_i \oplus y_i) \tag{1}$$

and the condition for a zero carry is

$$c_i^0 = \bar{x}_i \bar{y}_i + c_{i-1} \cdot (\bar{x}_i \oplus \bar{y}_i) \tag{2}$$

Note that both the one- and zero-carry lines must be monitored so that a false completion signal will not be generated. One stage of an accumulator using carry-completion detection is shown in Figure 9 [7]. A method for evaluating the reduction in time achieved by the use of asynchronous addition techniques has been given by Hendrickson [10], who arrived at a value of 90 percent savings for a 100-bit adder.

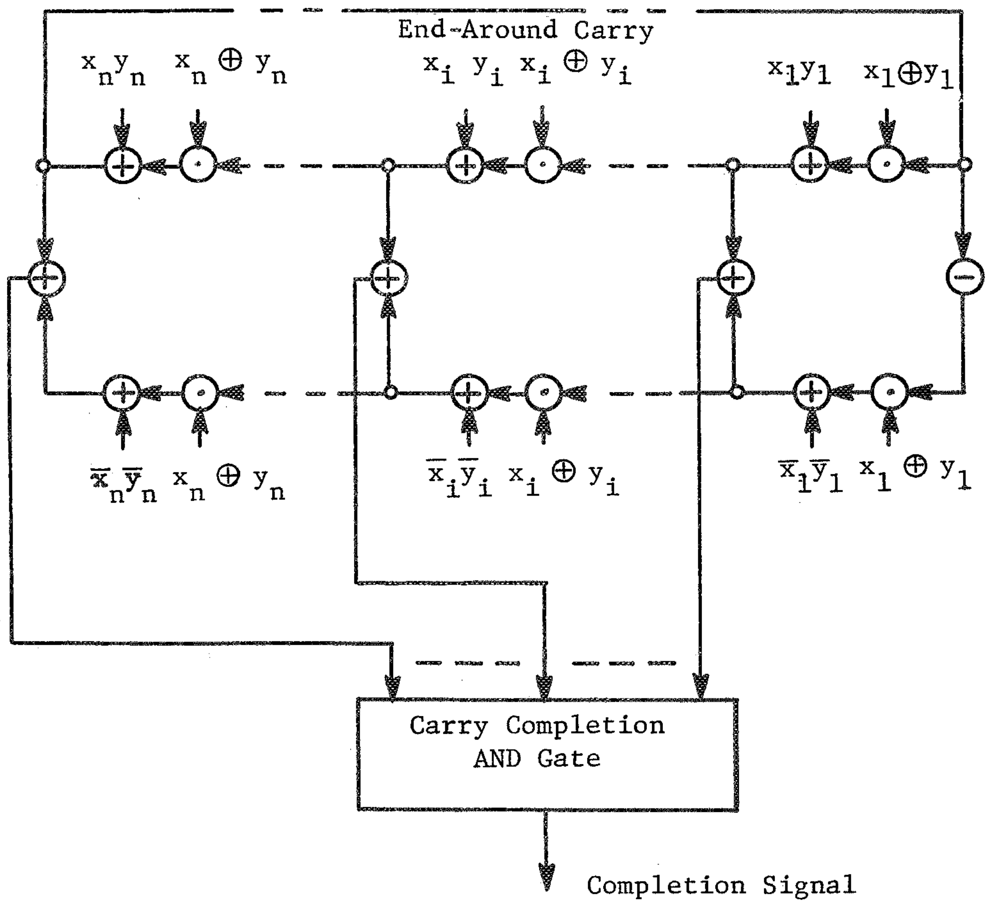


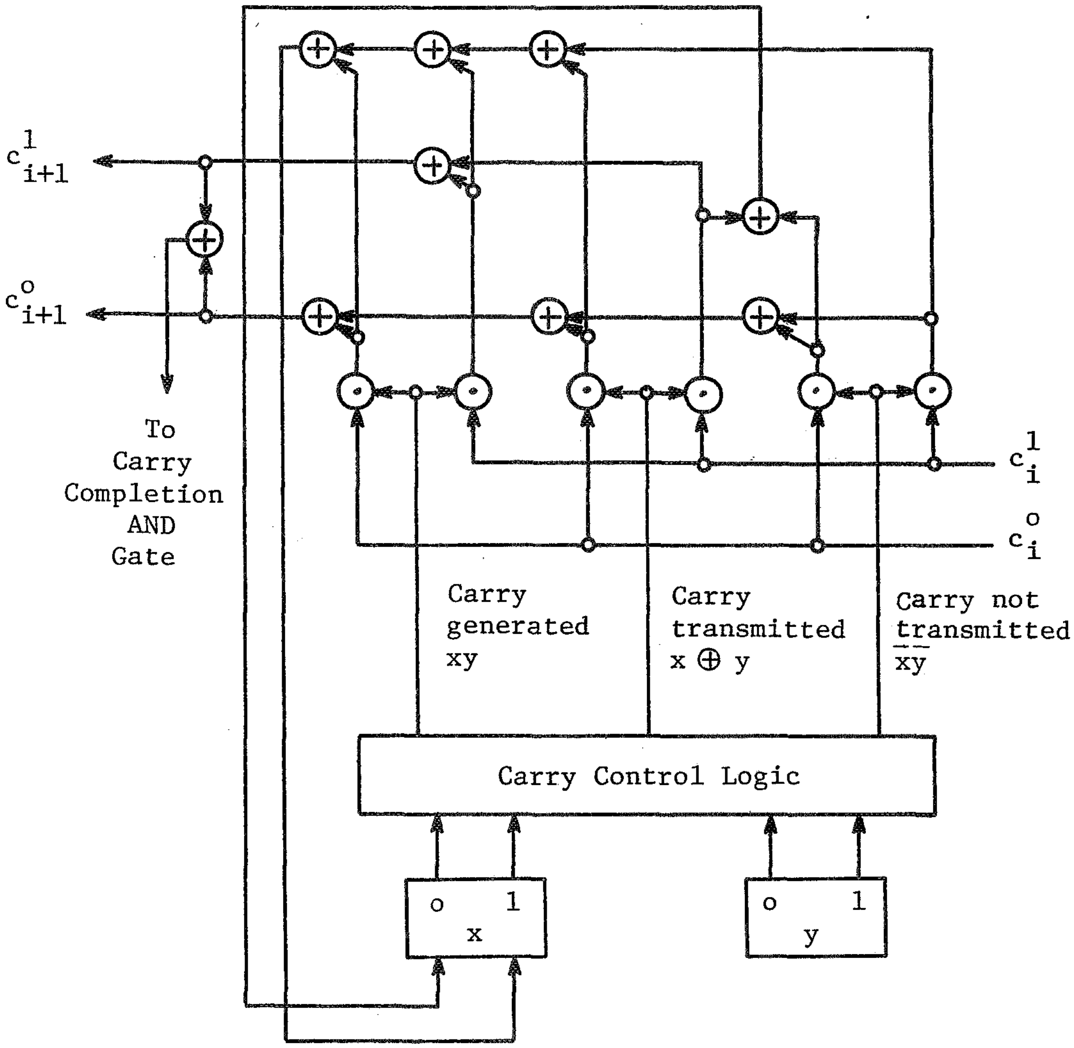
Figure 8. Principle of carry-completion detection.

A NOR gate binary adder with carry-completion detection has been presented by Majerski and Wiweger [11]. One stage of their realization of an adder is shown in Figure 10. The Boolean formulas describing the circuit of Figure 10 are

$$D_i = \overline{x_iy_i + \bar{x}_i\bar{y}_i} \tag{3}$$

$$D_ic_i = \overline{x_iy_i + \bar{x}_i\bar{y}_i + \bar{x}_{i-1}\bar{y}_{i-1} + D_{i-1}\,\bar{c}_{i-1}} \tag{4}$$

$$D_i\bar{c}_i = \overline{x_iy_i + \bar{x}_i\bar{y}_i + x_{i-1}y_{i-1} + D_{i-1}\,c_{i-1}} \tag{5}$$


c^1_{i+1}
c^o_{i+1}
To Carry Completion AND Gate
c^1_i
c^o_i
Carry generated xy
Carry transmitted $x \oplus y$
Carry not transmitted $\bar{x}\bar{y}$
Carry Control Logic
o 1 x
o 1 y

x - Accumulator Register Element

y - Input Register Element

Sum, X+Y, stored in Accumulator Register

Figure 9. Carry-completion detection, one stage.

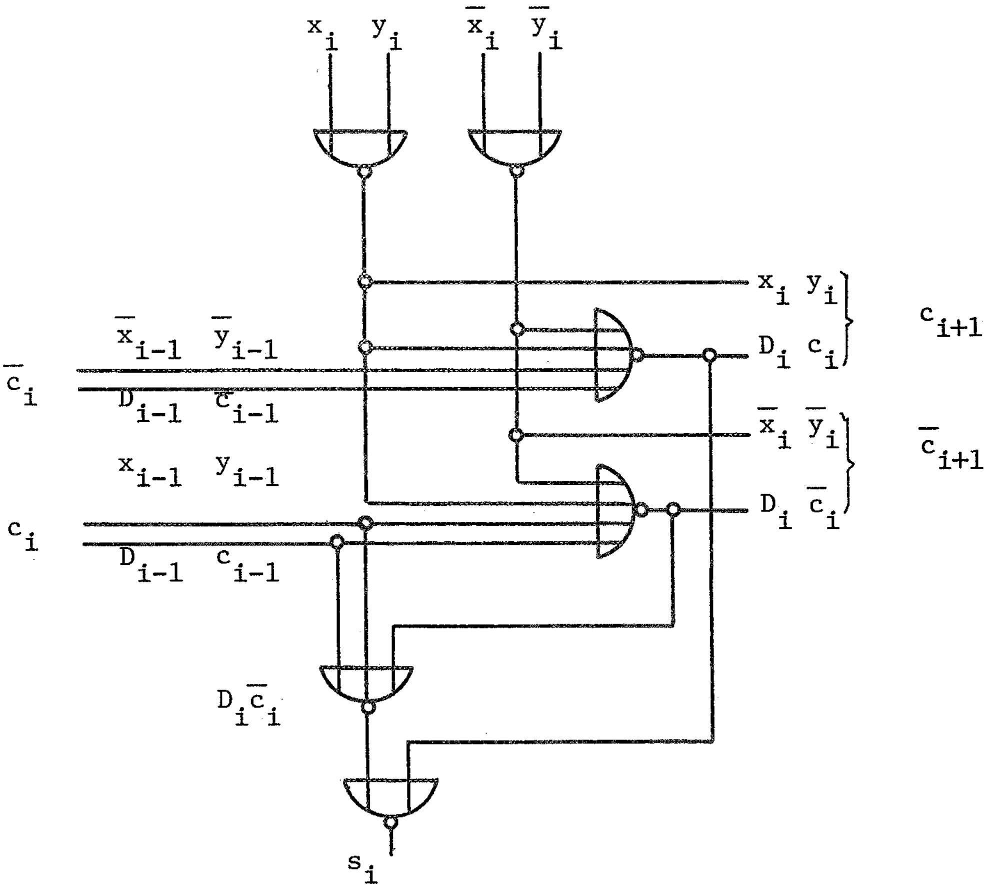


Figure 10. NOR gate adder stage [11].

$$s_i = \overline{D_i c_i + \overline{D}_i \overline{c}_i}$$

$$= \overline{D_i c_i + D_i \overline{c}_i + x_{i-1}\, y_{i-1} + D_{i-1}\, c_{i-1}} \qquad (6)$$

where

$x_i y_i$ $(i = 1, 2, \ldots, n)$ = bits of summands

c_i $(i = 1, 2, \ldots, n+1)$ = carry from i-1 to i position

s_i $(i = 1, 2, \ldots, n)$ = i^{th} bit of sum

Figure 11 shows the adder carry and carry-completion detection circuits of Majerski and Wiweger [11]. An advantage of the adder carry circuit is that a carry must propagate through only one gate per stage, whereas two gates are required in other known designs. The Boolean formulas for the carry-completion detection circuit are

$$H_i = \overline{x_i y_i + \bar{x}_i \bar{y}_i + D_i c_i \bar{h} + D_i \bar{c}_i \bar{h}} \tag{7}$$

$i = 2, 4, 6, \ldots, 2E\ (n/2)$

$$H = \overline{H_2 + H_4 + H_6 + \ldots + H_{2E\ (n/2)}} \tag{8}$$

where h is the carry control signal.

Operation begins with $h = 1$ and the add control signal. After at least 2d, where d is the maximum propagation time of a NOR gate, h is switched to 0. The carry propagation process then begins and lasts until all $D_i c_i$ and $D_i \bar{c}_i$ gates stabilize. The $h = 1$ signal is applied to prevent a false carry-completion signal from being generated. The carry-completion signal may precede the real completion by up to d since only every second position is monitored.

Carry-Skip Techniques

Assuming the complements of both summands are available, an FBA stage may be represented as shown in Figure 12. The functions representing the sum and carry are then

$$s_i = x_i \oplus y_i \oplus c_i \tag{9}$$

$$c_{i+1} = x_i y_i + x_i c_i + y_i c_i \tag{10}$$

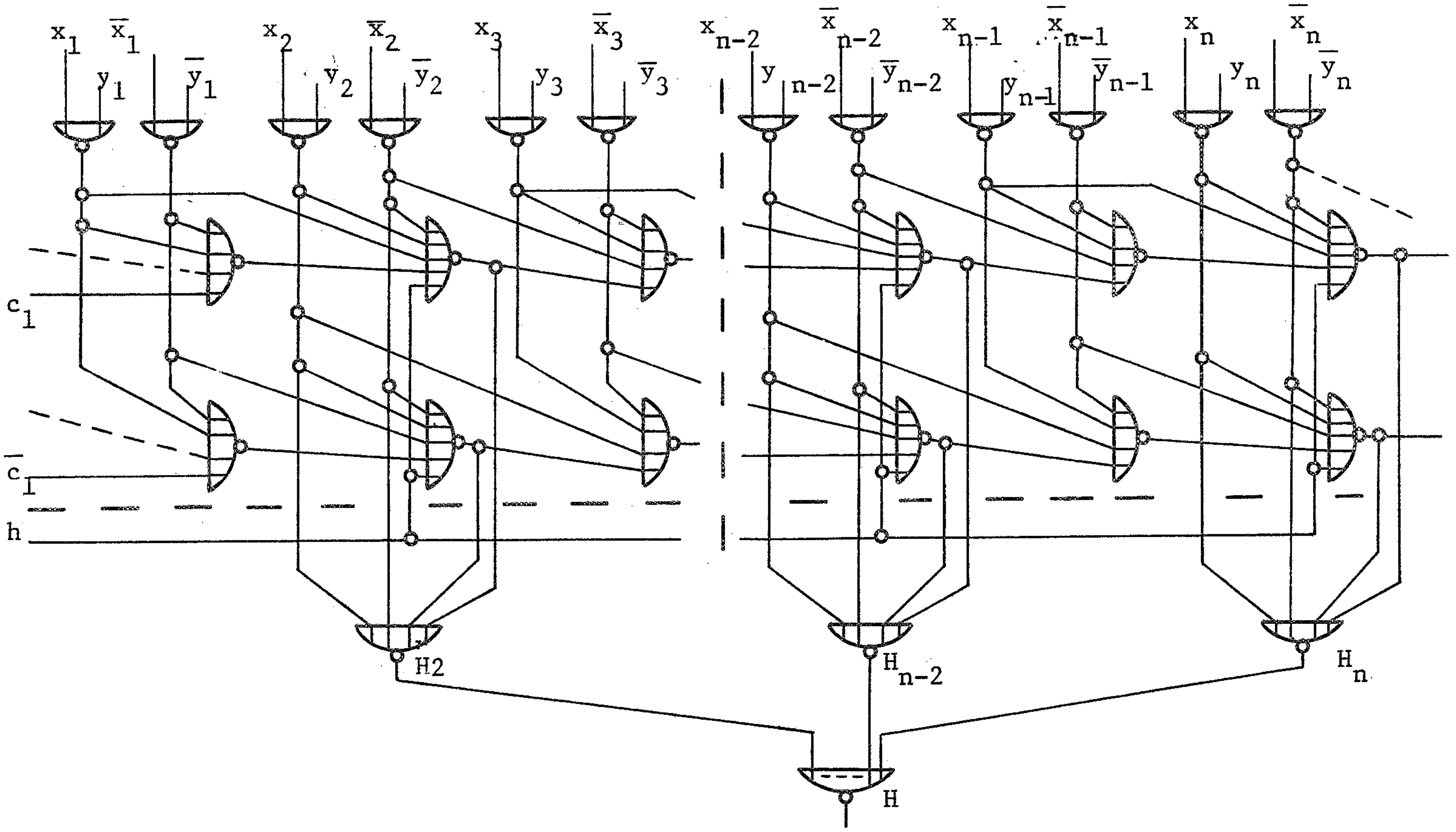


Figure 11. NOR gate realization of adder carry and carry completion detection circuits [11].

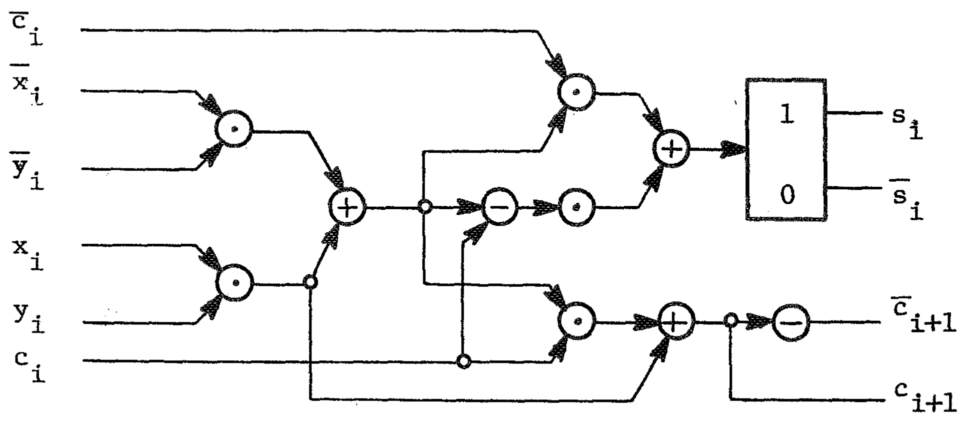


Figure 12. Full binary adder stage.

Let

$$c_{i+1} = r_i, \quad (c_i = r_{i-1}) \tag{11}$$

Equation (10) can be written as

$$r_i = (x_i \oplus y_i)\, c_i + x_i y_i \tag{12}$$

Let

$$t_i = x_i \oplus y_i \ = \ \text{condition for transmission of a carry} \tag{13}$$

$$g_i \ = \ x_i y_i \ = \ \text{condition for generation of a carry.} \tag{14}$$

Then

$$r_i = g_i + t_i c_i \ = \ \text{carry out of a stage.} \tag{15}$$

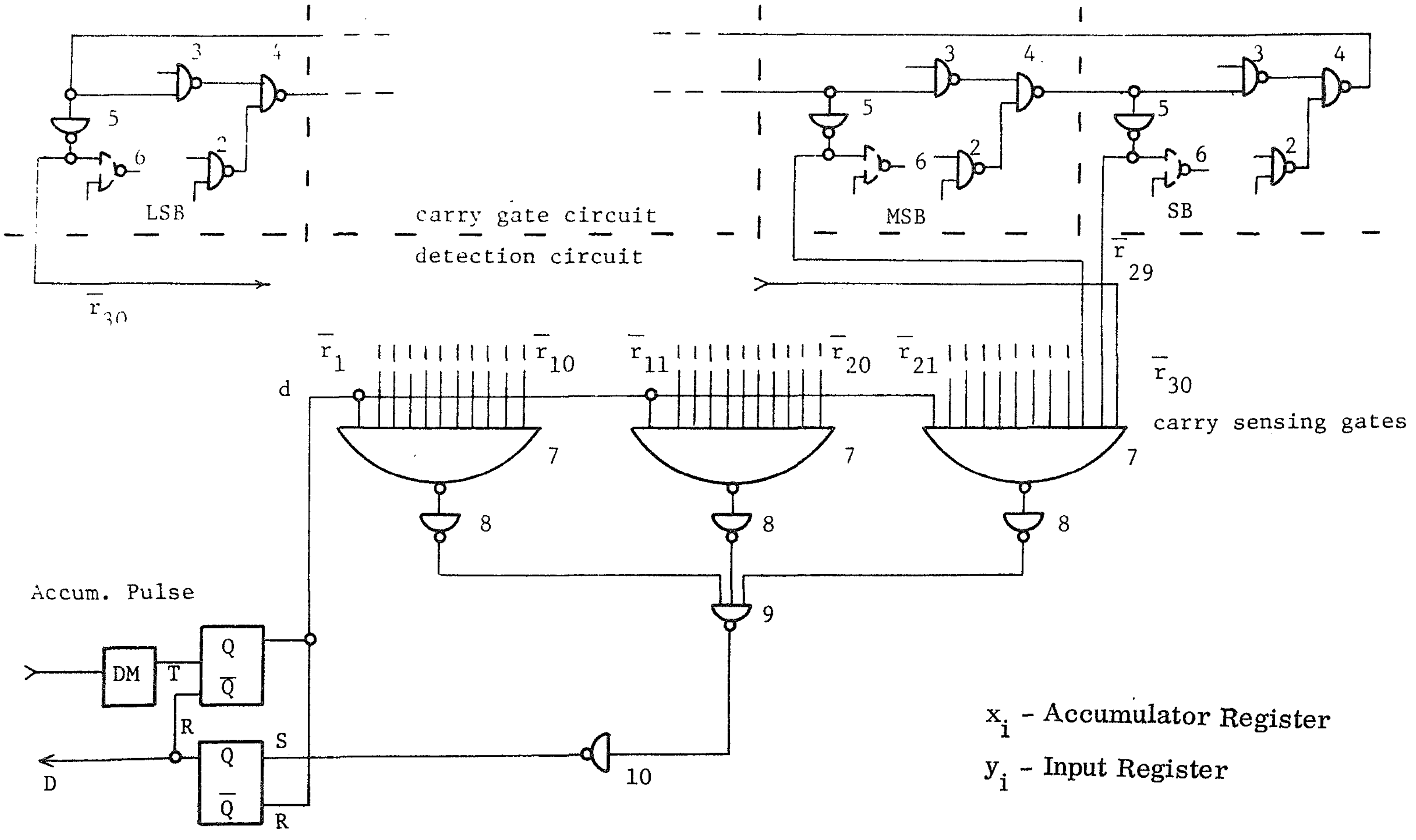
The carry into any stage may then be expanded as follows:

$$\begin{aligned} c_i &= r_{i-1} \\ &= g_{i-1} + t_{i-1} \cdot r_{i-2} \\ &= g_{i-1} + t_{i-1} \cdot g_{i-2} + t_{i-1} \cdot t_{i-2} \cdot r_{i-3} \\ &= g_{i-1} + t_{i-1} \cdot g_{i-2} + t_{i-1} \cdot t_{i-2} \cdot g_{i-3} \\ &\quad + t_{i-1} \cdot t_{i-2} \cdot t_{i-3} \cdot r_{i-4} \ [4] \ . \end{aligned} \tag{16}$$

The expansion of r may be carried as far back as desired. The limit for an n-stage adder is an expression for c_n containing r_1.

Application of this principle to a section of an adder is illustrated in Figure 13 [4]. Allowing the i to take on successive values in equation (4), and omitting all terms with negative subscripts, it is seen that each stage of the adder will require one i-input OR gate and i AND gates having one through i inputs. Thus the number of circuit elements would become prohibitive for adders of more than a few stages. However, the maximum carry path between any two stages is two levels and it is four levels for a complete addition.

The adder can be broken up into groups of stages connected in the manner described. The carry into each group would be designated c_{gi}, the carry out of a group designated r_{gi}, and the group containing the lower-order stages designated group 1. Assume that six stages would be a reasonable number of stages to be connected with full carry-look-ahead. If the six-stage groups are now connected in series with $c_{gi} = r_{g(i-1)}$, a carry will require four levels to be generated, two levels to be transmitted through each intermediate group, and four levels to reach and produce a sum in the final group. For 6-bit groups then, the maximum carry path length would be 4 + (2n/6), where n is the number of stages. For a 30-bit adder, this would be 14. This technique may be extended by providing carry-look-ahead circuitry between groups and even further by dividing the groups into sections and providing look-ahead between sections. A carry-look-ahead, or carry-skip, network for three groups is illustrated in Figure 14.



x_i - Accumulator Register

y_i - Input Register

Figure 13. Three-bit adder group with carry prediction [4].

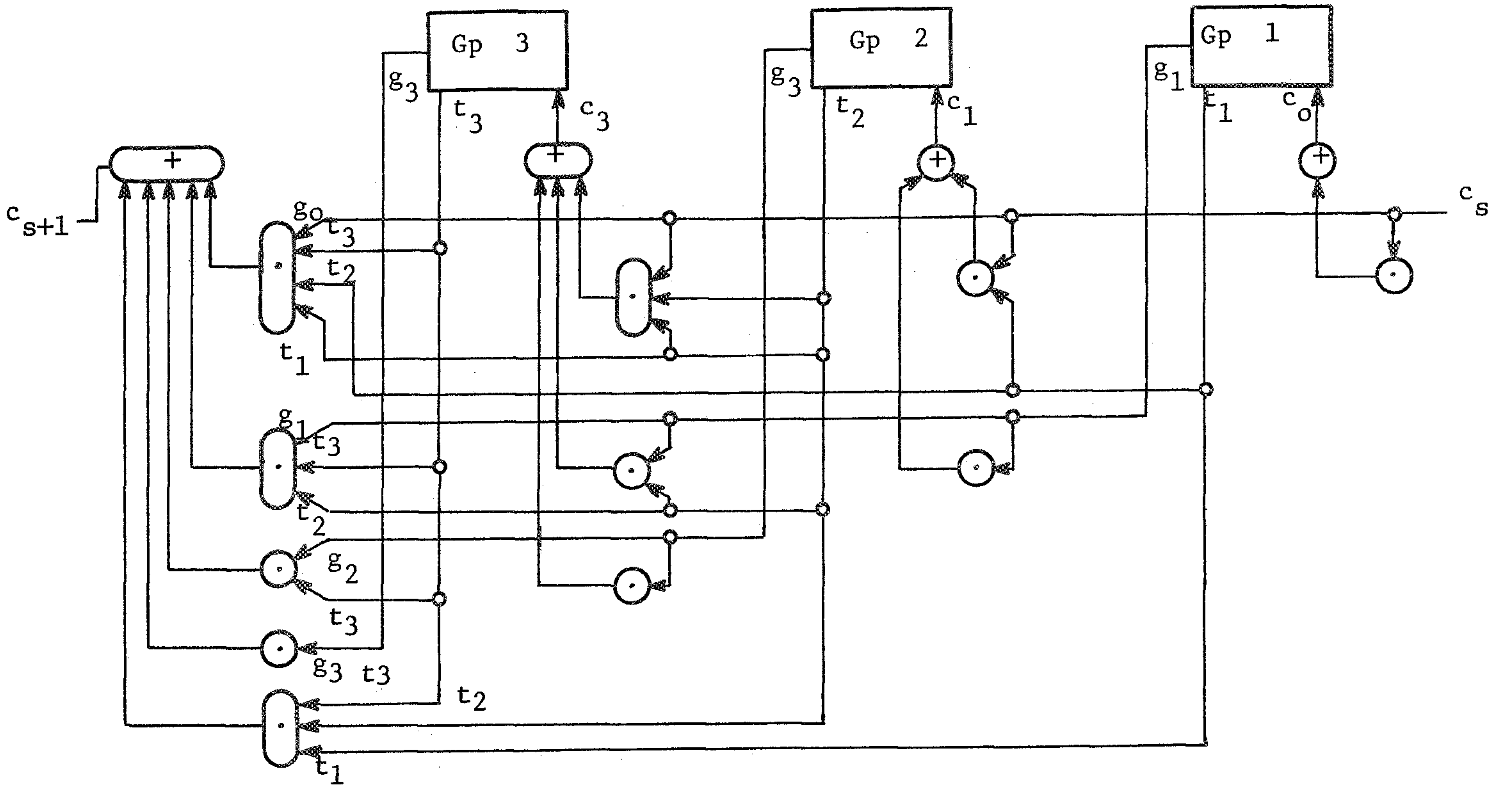


Figure 14. Carry-look-ahead for three groups.

Majerski [12] has proposed and implemented a NOR gate realization of a carry-skip circuit. For use of this circuit the skips may comprise only odd numbers of adder stages and their distribution relative to one another must always be an even number of stages. The adder with which this circuit is used is an asynchronous binary NOR gate implementation, of which the structure is given by the following equations:

For odd adder stages,

$$\bar{s}_i = \overline{\overline{D_i c_i + \bar{x}_{i-1}\,\bar{y}_{i-1} + D_{i-1}\bar{c}_{i-1}} + \overline{x_i y_i + \bar{x}_i \bar{y}_i + D_i c_i}} \quad , \qquad (17)$$

$$D_i c_i = \overline{x_i y_i + \bar{x}_i \bar{y}_i + \bar{x}_{i-1}\,\bar{y}_{i-1} + D_{i-1}\,\bar{c}_{i-1} + h} \quad , \qquad (18)$$

where h is the carry control signal;

for even adder stages,

$$s_i = \overline{\overline{D_i \bar{c}_i + x_{i-1} y_{i-1} + D_{i-1} c_{i-1}} + \overline{x_i y_i + \bar{x}_i \bar{y}_i + D_i \bar{c}_i}} \quad , \qquad (19)$$

$$D_i \bar{c}_i = \overline{x_i y_i + \bar{x}_i \bar{y}_i + x_{i-1} y_{i-1} + D_{i-1} c_{i-1}} \quad , \qquad (20)$$

where

$$D_i = \overline{x_i y_i + \bar{x}_i \bar{y}_i} \qquad (21)$$

Carry propagation begins with a change in h from 1 to 0 and ends with the stabilization of the states of all gates.

The carry-skip circuit, comprising k stages from j + 1 to j + k of an adder, is described by the function:

$$c_{j,k} = \overline{D_j \bar{c}_j + \bar{x}_j \bar{y}_j + \bar{x}_{j+1} \bar{y}_{j+1} + \ldots + \bar{x}_{j+k} \bar{y}_{j+k}} \qquad (22)$$

where j is an even number and k is an odd number. The carry and skip circuits for $k = 3$ are shown in Figure 15 [12].

Majerski's equations for determining optimal and economical skip distributions in adders using his design are presented in the Appendix.

DESCRIPTION OF A SIMPLE ITERATIVE ACCUMULATOR

The parallel binary accumulator shown in reduced block diagram form in Figure 16 has been used by the Auburn Research Foundation in the design of a digital compensator [2]. It was selected because of its automatic carry feature and relatively simple control circuitry. Its properties will be briefly described here and used as a basis on which to judge improvements in carry propagation time resulting from the application of fast carry techniques.

The blocks labeled AC are T flip-flop storage elements of the accumulator register, and those labeled DM are delay multivibrators used for carry generation. The NAND gate pairs in the lower part of the figure form the augend-complementing structure. The input from the complementing-gate structure is a negative-going pulse when the input is positive. The negation output of the DM is used. The input to the i^{th}-stage flip-flop (AC) is then

$$\overline{\bar{y}_i \cdot \bar{c}_i} = y_i + c_i \qquad (23)$$

The output equation of the i^{th} flip-flop is

$$x_i' = t_i \bar{x}_i + \bar{t}_i x_i \quad , \qquad (24)$$

where t_i is the input.

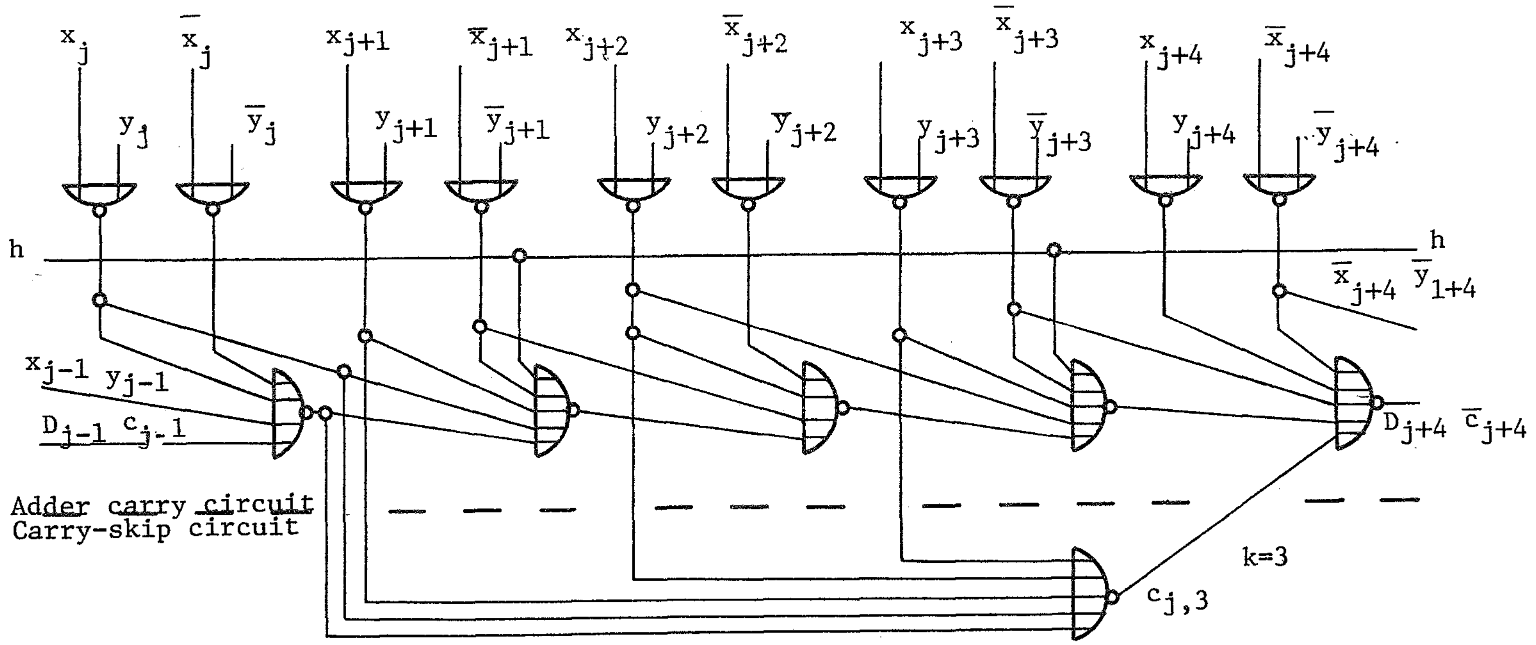


Figure 15. Carry-skip circuit [12].

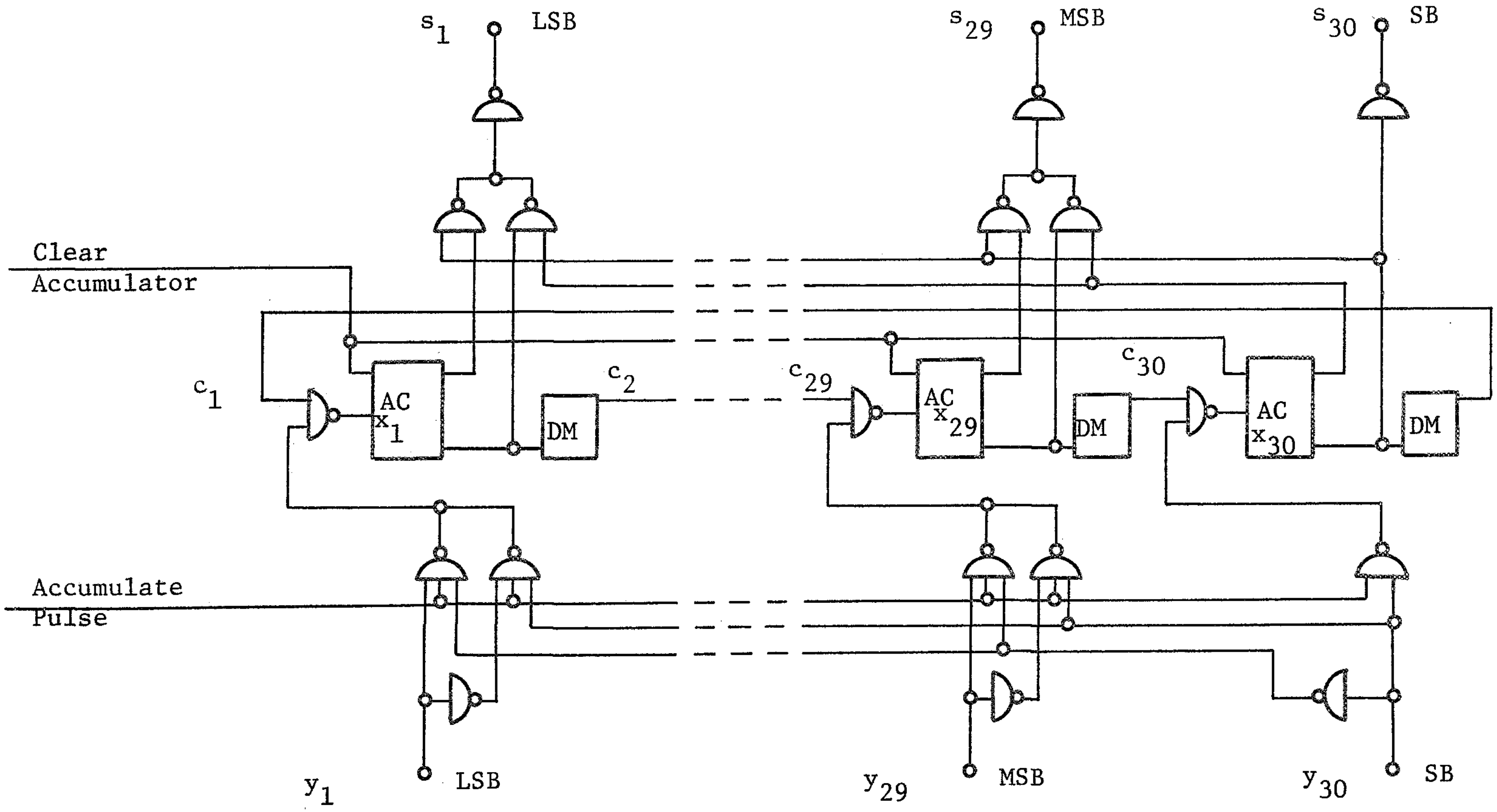


Figure 16. A simple iterative accumulator [2].

The total addition for a stage takes place in two steps. In the first step,

$$t_i = y_i$$

Then

$$\begin{aligned} x'_{i+1} &= y_i\bar{x}_i + \bar{y}_i x_i \\ &= y_i \oplus x_i \qquad (25) \end{aligned}$$

The complement output of the flip-flop is the function

$$\overline{y_i \oplus x_i} = y_i \odot x_i \quad . \qquad (26)$$

In the second step of the addition, the input of the i^{th} flip-flop is c_i. The final output is then

$$\begin{aligned} x'_{i+2} &= c_i\ \overline{(y_i \oplus x_i)} + \bar{c}_i\ (y_i \oplus x_i) \\ &= c_i \oplus y_i \oplus x_i \qquad (27) \end{aligned}$$

The accumulate operation is initiated by the application of the accumulate pulse. The contents of each stage of the augend, or input, register are gated to the input of the corresponding stage of the addend, or accumulator, register. This operation is simultaneous for all stages within the limits of the variation in propagation time between NAND gates. If the augend bit is a logic one, with the sign bit a logic zero (positive), the accumulator flip-flop changes state. Thus if a logic one from the augend stage is added to a logic one in the accumulator stage, the partial sum, a logic zero, is stored in the accumulator. The transition of the accumulator stage from a one to a zero state triggers the delay multivibrator and generates a carry, which is routed to the succeeding stage. If a logic one from the augend stage is added to a logic zero in the corresponding accumulator stage, the partial sum, a logic one, is stored in the accumulator. The transition of the accumulator stage from a zero state to a one state does not trigger the delay multivibrator, and no carry is generated. At this point the accumulator contains the results of the operation $x_i \oplus y_i$. The carry generated in the preceding stage is then added to the contents of each stage of the accumulater, forming another partial sum and possibly generating other carries. The delay time of the multivibrator is such that all partial sums are formed before the carries are generated. A timing diagram for the operation of one stage of the accumulator is shown in Figure 17. The AC registers trigger when there is a change from the one to the zero level. Therefore, the trailing edge of the accumulate pulse is used as the zero-time reference. At the end of the accumulate operation, the augend in the input register has been added to the addend in the accumulator register, and the sum has been stored in the accumulator register.

Typical delay times of the elements used in the accumulator are given in Table 2 [13]. The DM pulse width is set at 100 ns to assure operation of all circuits to which it is applied. It can be seen from the values in this table and by referring to Figure 16 that a maximum delay time of 120 ns is required from the end of the accumulate pulse to the formation of partial sums in the accumulator register. The time required to form a carry and transmit it to the succeeding register is a 30-ns delay for the multivibrator, a 100-ns pulse width of the multivibrator, and a 30-ns delay for the input gate, yielding 160 ns. The total time for the generation and transmission of the first carry is then 280 ns. The time for the generation and transmission of each secondary carry (generated by the assimilation of a carry) would be 220 ns, since the delay time of the complementation gate and one input gate would then be omitted. The worst case carry propagation time for a 30-stage accumulator of this design would then be 280 ns for the generation of a primary carry in the first stage, 6160 ns (28×220) for transmission through the intermediate stages, and 60 ns for assimilation in the 30th stage, yielding 6.50 μs.

The relative cost of the circuit elements used in the accumulator is given in Table 3 [13], using a 2-input NAND gate as a basis. Inverters will be considered as 2-input NAND gates with the inputs common. A considerable variation in cost is found, but these figures should be realistic enough for comparative evaluation of designs.

The cost figure for the portion of this accumulator dealing with carry propagation is then 630.

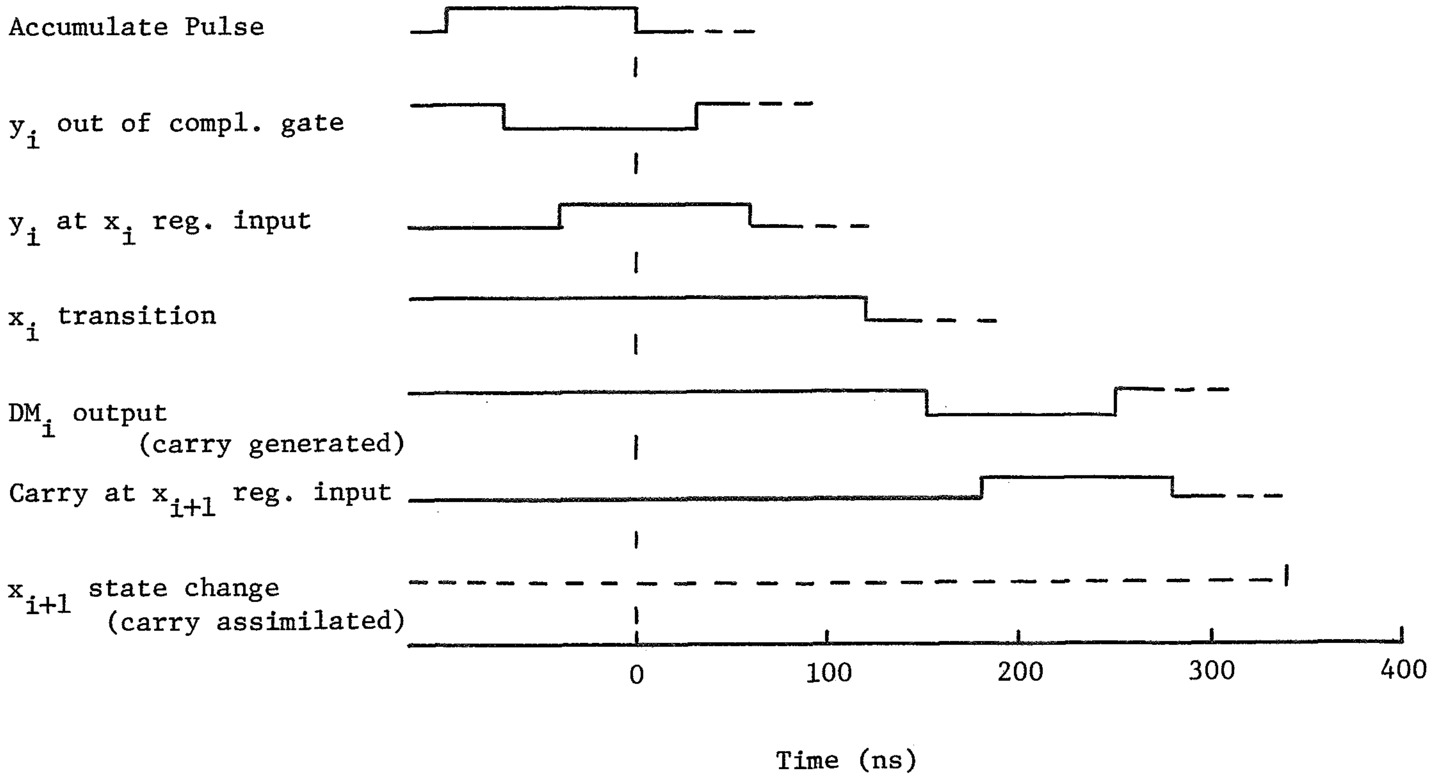

Accumulate Pulse
y_i out of compl. gate
y_i at x_i reg. input
x_i transition
DM_i output (carry generated)
Carry at x_{i+1} reg. input
x_{i+1} state change (carry assimilated)
0
100
200
300
400
Time (ns)

Figure 17. Timing diagram for carry generation and assimilation.

TABLE 2. LOGIC ELEMENT DELAY TIMES

Logic Element	Typical Delay Time	Maximum Delay Time (ns)
NAND gate	25 ns	30
AC register:		
T input	45 ns	60
DC input	65 ns	80
Delay multivibrator:		
Assertion delay		60
Negation delay		30
Minimum Pulse width	50 — 300 ns, 300 ns — 3μs	

TABLE 3. RELATIVE COST OF LOGIC ELEMENTS

Logic Element	Relative Cost
2-input NAND gate	1
3-input NAND gate	2
4-input NAND gate	4
6-input NAND gate	6
12-input NAND gate	12
Flip-flop	8
Delay multivibrator	12

RAPID ACCUMULATOR REALIZATIONS AND ANALYSIS

The methods previously reviewed were investigated with regard to their application to the accumulator. This accumulator utilizes pulse logic for the input and carry signals. NAND gates were used in the design of combinational circuits because of their availability in integrated circuit form and their use in the basic accumulator.

The techniques presented by Majerski and Wiweger [11, 12] offer faster propagation times than those given in the following sections when used with adders of their design, but were not directly applicable to this accumulator because of its use of pulse logic.

Both one- and two-layer carry-skip methods were investigated. It was found that two-layer skips offered no appreciable speed advantage over one-layer skips for this accumulator.

The results of the investigation are presented in the following sections.

Gated Carry

It is seen from Table 1 that the carry out of a stage depends solely upon the carry in when the summands are unlike. The function representing unlike summands is

$$x_i \oplus y_i = x_i\bar{y}_i + \bar{x}_iy_i$$

The carry out of a stage depends only on the summands when they are both alike. The function representing like summands is

$$x_i \odot y_i = x_iy_i + \bar{x}_i\bar{y}_i$$

In the accumulator previously described, the above two functions are available from the i^{th}-stage flip-flop of the accumulator at the true and complement outputs, respectively. The carry from the i^{th} stage is generated when the complement output goes to the logic-one level. The functions necessary for applying the gated-carry technique presented are available. The assimilation of a carry in a stage for which the function $x_i \oplus y_i$ is true will produce another carry, which poses no problem with level logic. In the accumulator, extraneous pulses would be inserted into the carry propagation line. The suppression of these extraneous pulses must be considered in applying the gated-carry technique.

Let r_i be the carry inserted into the carry propagation line by the i^{th} stage, and let

$$c_i = r_{i-1} \tag{28}$$

Let g_{i-1} be the carry pulse generated by the condition $x_{i-1}y_{i-1}$, which results when both summands of the i-1 stage are at the logic-one level. The function for c_i must then be

$$c_i = r_{i-2} \cdot (y_{i-1} \oplus x_{i-1}) + eg_{i-1} \quad , \tag{29}$$

where e is a carry-generation enable pulse. This pulse must be present for the generation of a carry in step one of the addition process, and absent to prevent the generation of an extraneous carry when a carry is assimilated.

A circuit to realize this c_i function is shown in Figure 18. The equation for c_i, written from the figure, is

$$c_i = \overline{\overline{\overline{\overline{r_{i-2} \cdot (x_{i-1}y_{i-1})} \cdot \overline{eg_{i-1}}}}}$$

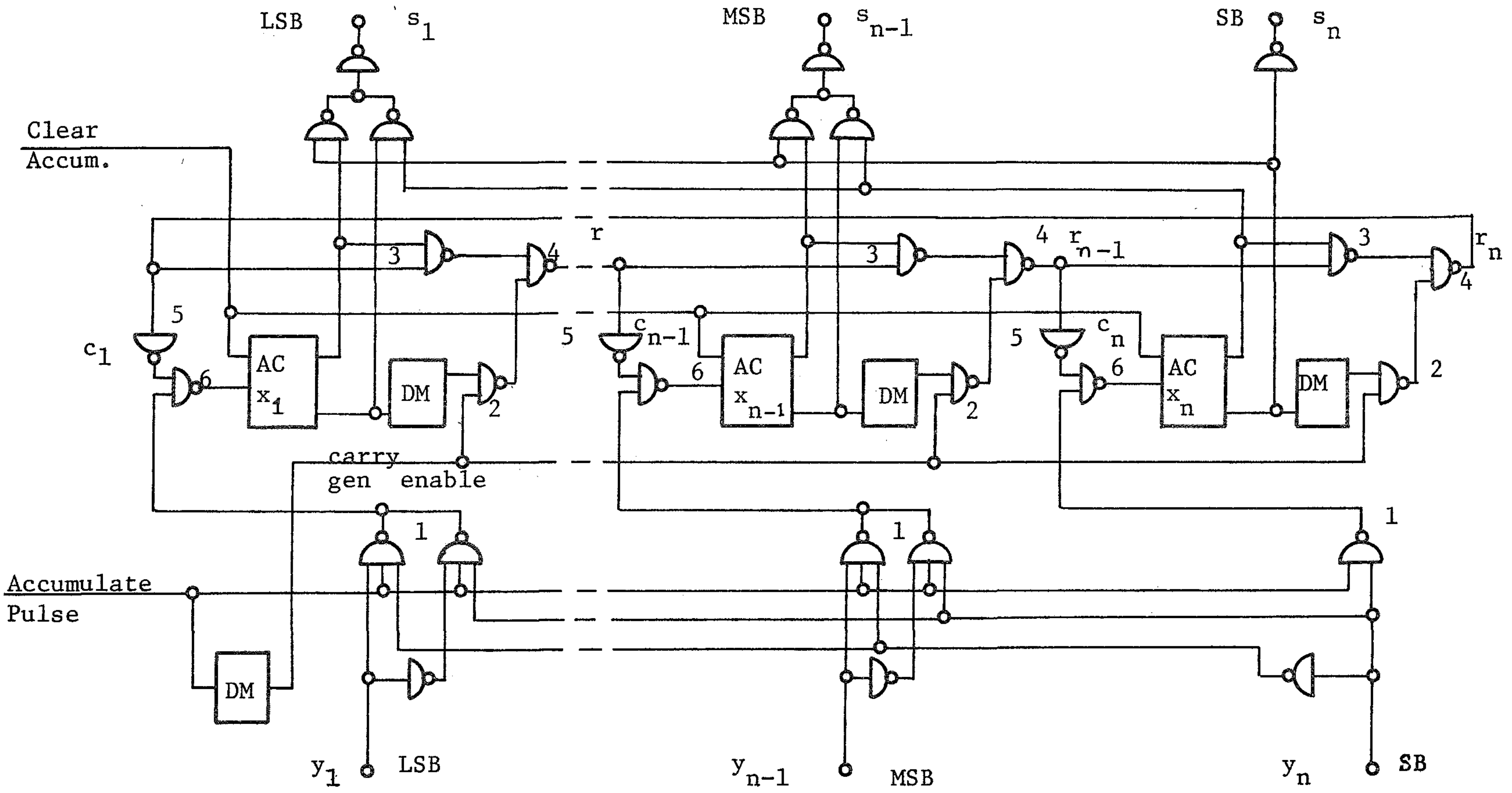


Figure 18. Accumulator with gated carry.

Applying DeMorgan's theorem,

$$c_i = r_{i-2} \cdot (y_{i-1} \oplus x_{i-1}) + eg_{i-1} \qquad ,$$

which is the desired result.

The accumulation step is initiated with the accumulate pulse. If x_i and y_i are logic ones, the AC complement output changes state from a zero to a one, triggering the DM and generating a carry pulse. The carry pulse enters the carry propagation line through the gate labeled 4 in Figure 18. If x_i and y_i are both at the logic-zero level, the complement output of the AC is at the logic-one level and does not change state and no carry is generated. This is the situation in which the carry out is determined solely by the summands. The true output of the AC is then at the logic-zero level and attains this level 120 ns before a carry is generated in the preceding stages. This level is then applied to the gate labeled 3 in Figure 18, blocking the transmission of any carry generated in the preceding stages. A carry entering the i^{th} stage is routed through gates 5 and 6 to the AC input, causing a change in state. The assimilation of the carry then removes the transmission block, but does so 120 ns after the carry pulse has ceased to be present at gate 3.

If x_i and y_i are different, $x_i \oplus y_i$ is true, and the true output of the AC takes on the logic-one level. If x_i is one and y_i is zero, there is no AC state change. If y_i is one and x_i is zero, the AC would change state from a complement logic-one level to a true logic-one level. In either case, no carry would be generated. The resulting situation is that the carry out is determined solely by the carry in. The true level is applied to gate 3, enabling the transmission of an incoming carry. A carry into the stage is then assimilated through gates 5 and 6, causing a state change of the AC. This removes the gating level to gate 3, 120 ns after the carry is transmitted. No new carry is generated by the stage because of the removal of the carry-generation enabling level, e, from gate 2. The enabling level, e, is generated by a DM, which is actuated by the accumulate pulse. The DM pulse width is set at 370 ns to gate all primary carries generated and to block all secondary carries generated.

A timing diagram for one stage of the accumulator is shown in Figure 19. Assuming the necessary conditions, the time required to insert a carry into the propagation line after the end of the accumulate pulse is a 30-ns delay each for the complementing gate and gate 6, a 60-ns delay for the state change of the AC, a 60-ns delay for the DM, a 100-ns delay for the DM pulse width, a 30-ns delay for gate 2, and 30 ns for the insertion-gate delay, or a total of 340 ns. There is a delay of 30 ns per gate per stage, or 60 ns per stage through the carry propagation line. It requires 120 ns to assimilate a carry in a stage. The equation for determining the total accumulation time for two binary numbers is

$$A = (460 + 60\,n') \quad , \qquad (30)$$

where n' is the number of stages through which a carry must be propagated before assimilation. The total time required to generate a carry in the first stage of a 30-stage accumulator, propagate it through the entire length, and assimilate it in the last stage is

$$A = [460 + 60(28)]\ \text{ns} = 2.14\ \mu\text{s}$$

The equipment that must be added to the accumulator previously described to facilitate carry gating consists of four 2-input NAND gates per stage and one DM unit to generate the carry-generation enable pulse, e. The cost figure for the portion of this accumulator concerned with carry propagation is then 756.

The gated-carry design gives a speed increase of 6.50 μs/2.14 μs or 3.37 to 1 over the simple iterative accumulator. The cost increase is 21 percent.

Carry-Completion Detection

The gated-carry method produces an appreciable decrease in the maximum carry-propagation time. To operate the accumulator on a fixed-time basis, maximum delays with an added safety factor must be considered, and the accumulation rate must always allow time for the maximum length of carry propagation.

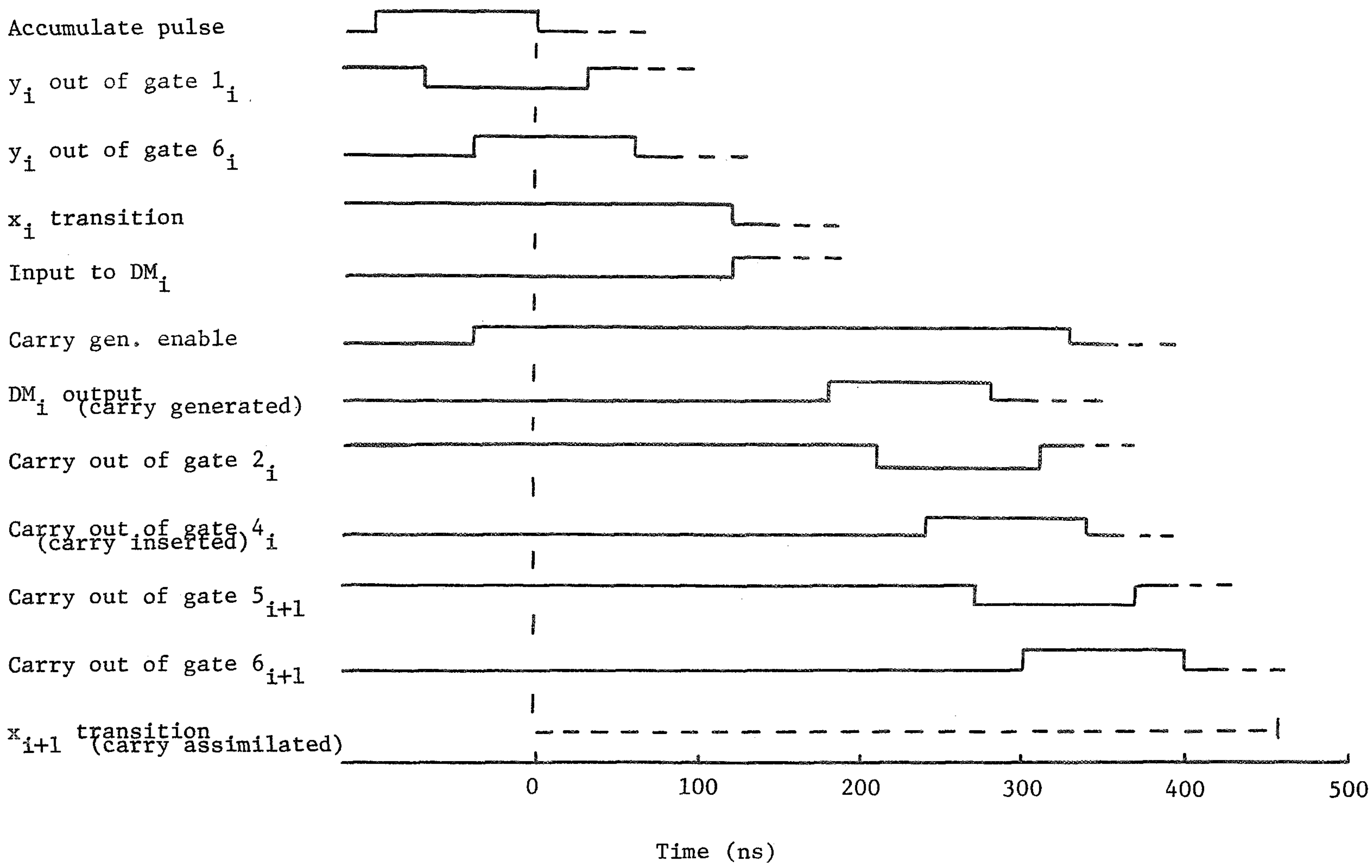


Figure 19. Timing diagram for carry generation and assimilation (gated carry).

As pointed out previously, the average carry propagation length is $\log_2 n$. This value is 4.91 for an n of 30, so more rapid accumulation could be accomplished using a variable rate, with the rate controlled by the completion of carry propagation. Let r_i, $i = 1, 2, \ldots, 30$, be the carry pulse inserted into the carry-propagation line by the i^{th} stage. Carry propagation is complete when a pulse is not present at any of the insertion points. The required condition for carry completion, D, is then

$$D = \bar{r}_1 \cdot \bar{r}_2 \cdot \bar{r}_3 \cdot \quad \cdots \quad \cdot r_{30} \qquad (31)$$

A carry-completion-detection circuit that could be used with the gated-carry accumulator is shown in Figure 20. From the figure, the carry-completion signal is a positive level described by the equation

$$D = \overline{\overline{\overline{\overline{d \cdot \bar{r}_1 \cdot \bar{r}_2 \cdot \quad \cdots \quad \cdot \bar{r}_{10}}} \cdot \overline{\overline{d \cdot \bar{r}_{11} \cdot \bar{r}_{12} \cdot \quad \cdots \quad \cdot \bar{r}_{20}}}}}$$

$$\cdot \overline{\overline{\overline{\overline{d \cdot \bar{r}_{21} \cdot \bar{r}_{22} \cdot \quad \cdots \quad \cdot \bar{r}_{30}}}}}$$

Applying DeMorgan's theorem gives

$$D = d \cdot \bar{r}_1 \cdot \bar{r}_2 \cdot \quad \cdots \quad \cdot \bar{r}_{30} \qquad ,$$

where d is the detection-enable level. This level is generated from the accumulate pulse by means of a DM and a T flip-flop. The delay of the DM is set to ensure that any carry being propagated will be present at the output of the gates labeled 5 before the detection system becomes operative. This prevents the generation of a false completion signal. The completion detection signal then resets the flip-flop in preparation for the next accumulate pulse, which will also be controlled by D. The timing diagram for this operation is shown in Figure 21. The overlap of carry signal at adjoining sensing points along the carry propagation line is shown in Figure 22 and is seen to be 40 ns.

The timing diagram for the generation and assimilation of a carry is shown in Figure 23. The maximum delay times of circuit elements were used

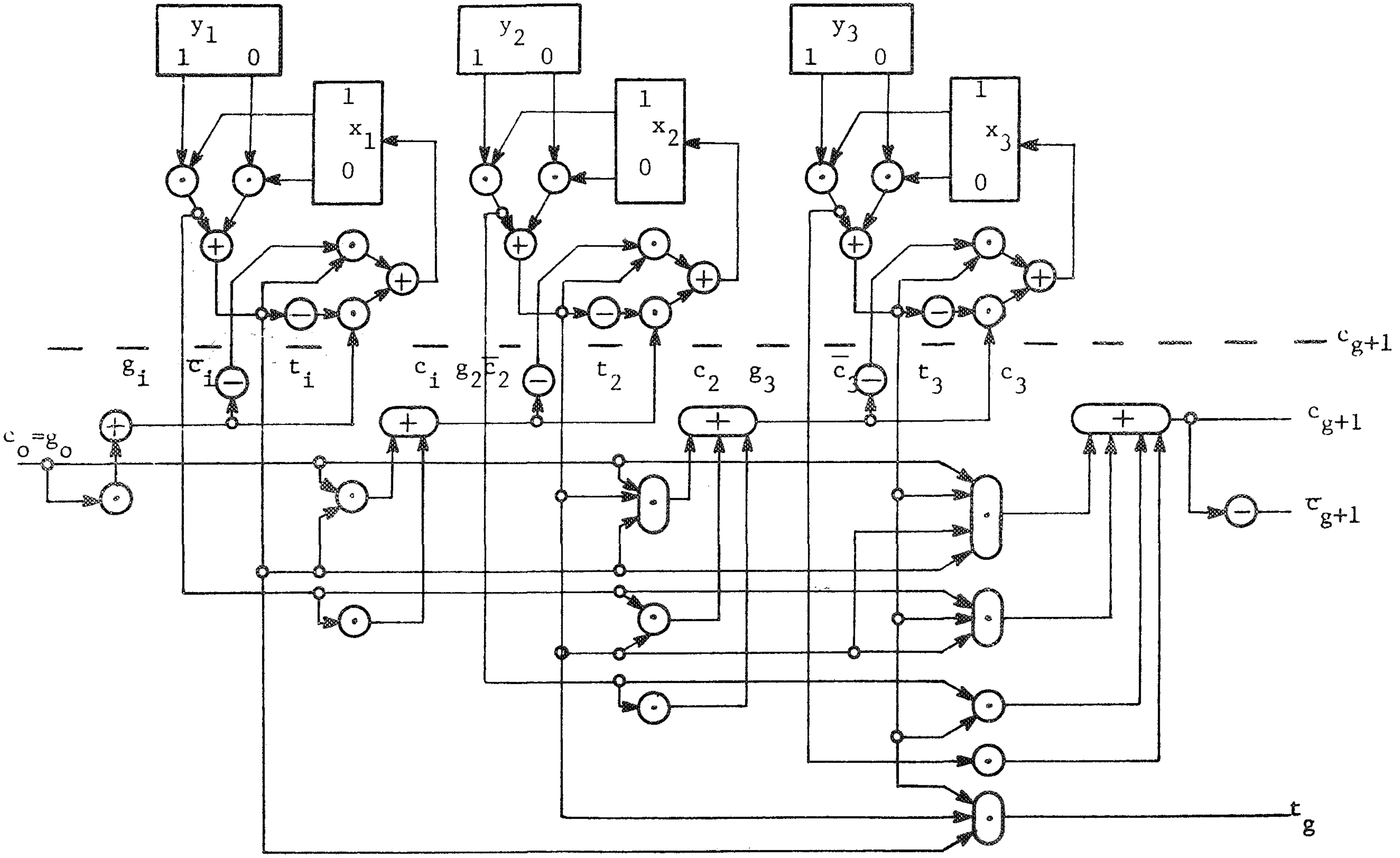


Figure 20. Carry-completion detection circuit.

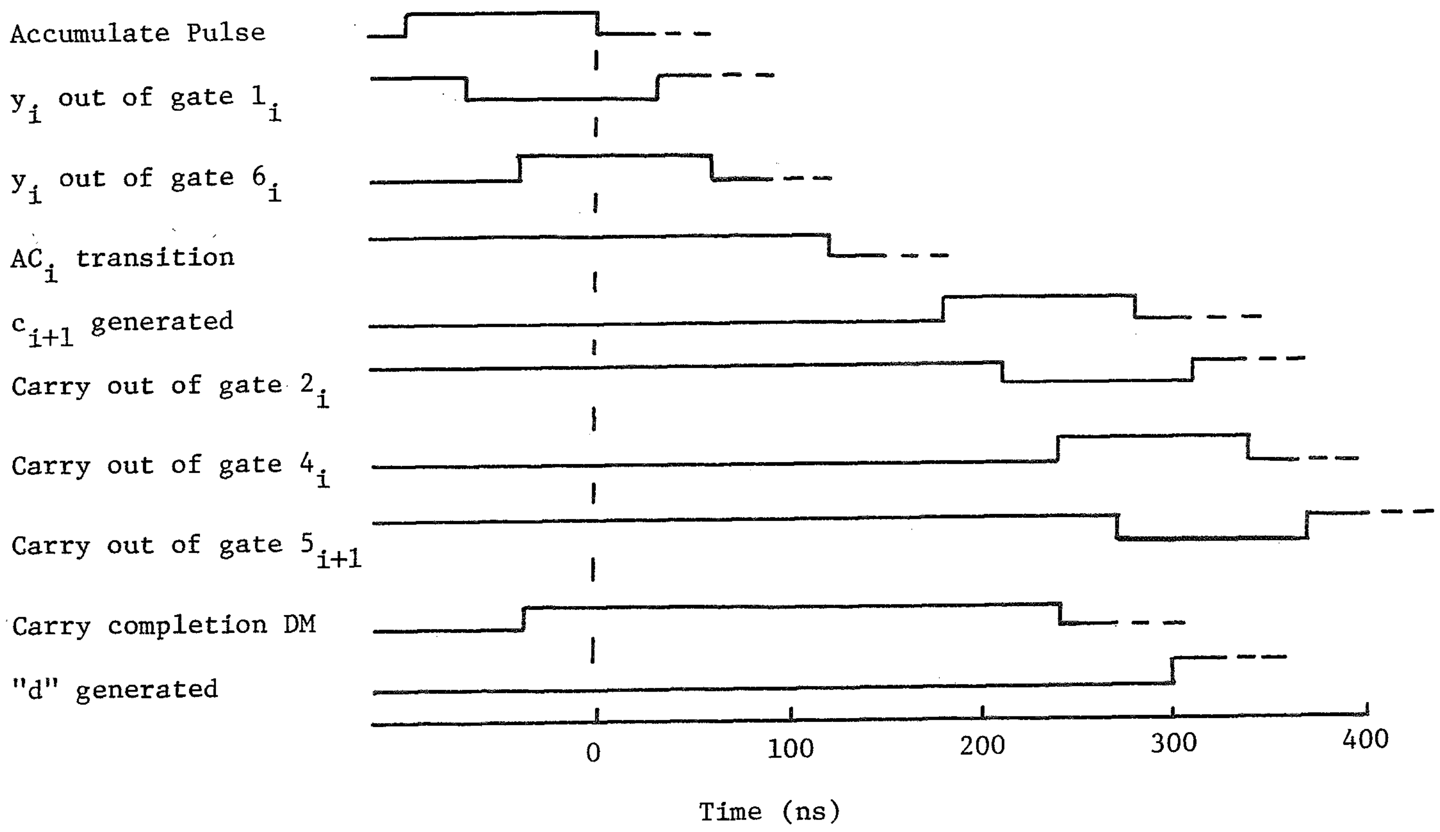


Figure 21. Timing diagram for detection enable signal generation.

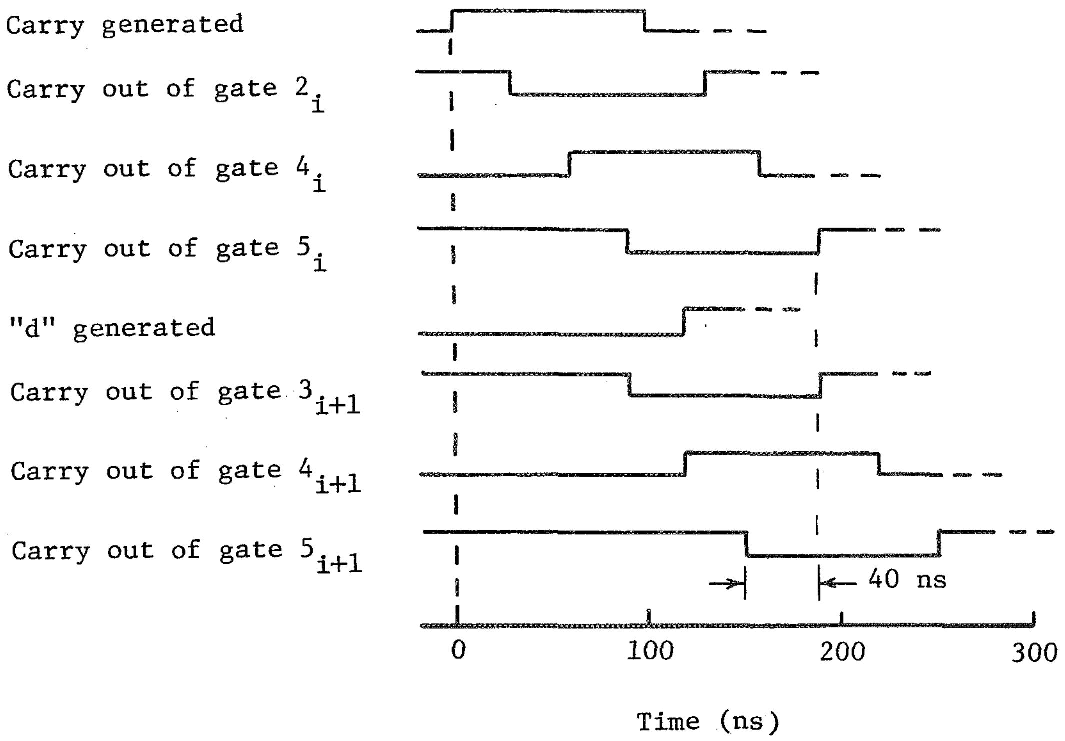


Figure 22. Continuity of carry signal at sensing gates.

in defining this diagram. It is seen that 340 ns are required to insert a carry into the carry propagation line. The time required to assimilate a carry after it has arrived at a stage is 120 ns. The equation for accumulation time is

$$A = (460 + 60n') \text{ ns} \tag{32}$$

where n' is the number of stages through which a carry must be propagated before it is assimilated. The time required to generate a carry and propagate it through the entire length of a 30-stage accumulator is

$$A = [460 + 60(28)] \text{ ns}$$

$$= 2.14\ \mu s$$

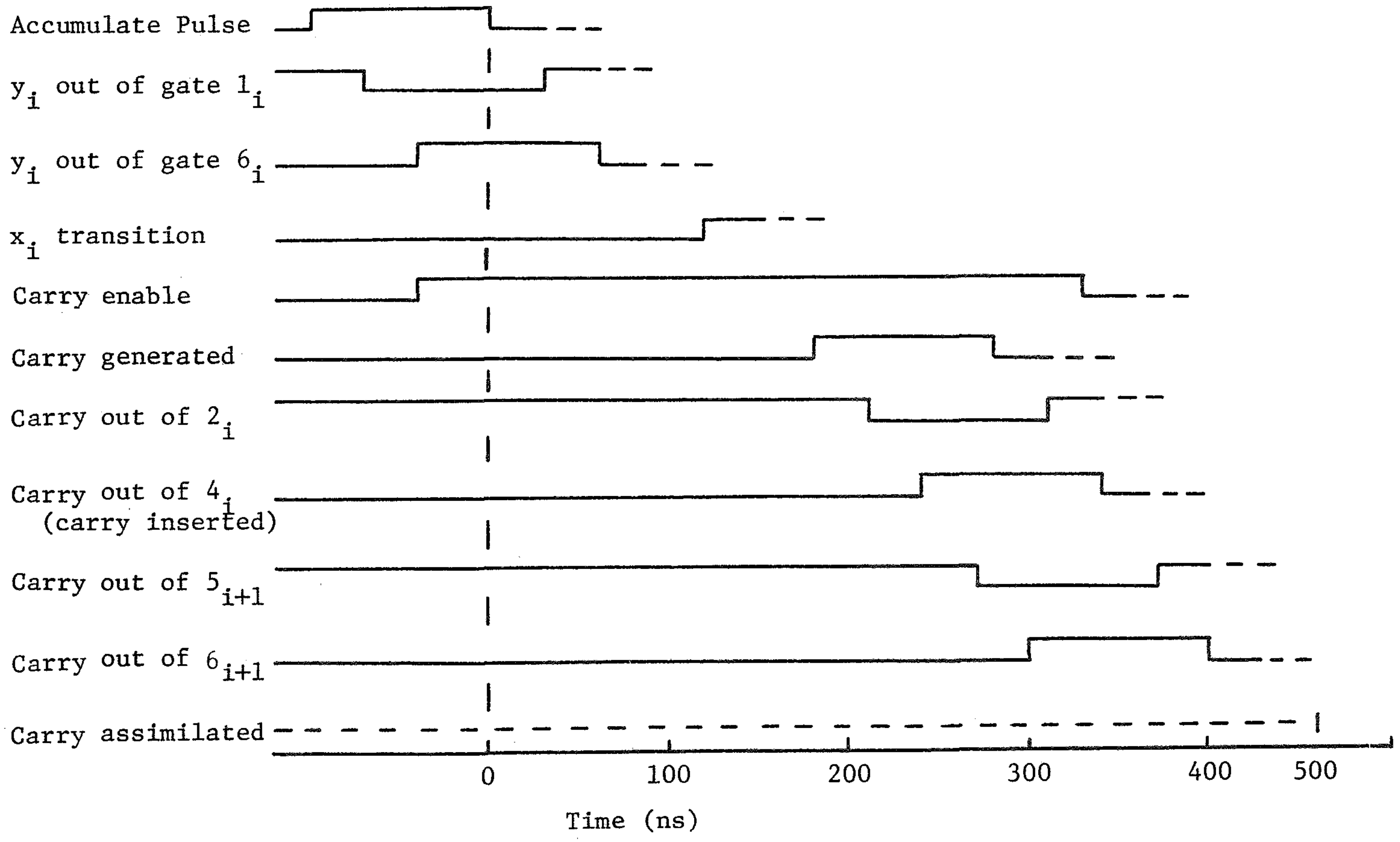


Figure 23. Timing diagram for carry generation and assimilation in carry-completion detection realization.

The carry propagation complete signal is generated 210 ns after the last carry has ceased to be present at a sensing point, as shown in Figure 24. The final accumulator-register transition is completed 90 ns from that time. The difference, 120 ns, must be added to the 2. 14 μs to yield the total maximum accumulation time of 2. 26 μs. For a one-stage propagation, x_{i-1} would change state 460 ns after the end of the accumulate pulse. The total time for the accumulation step in this case is 460 ns + 120 ns = 580 ns. If no carries are generated in an accumulation, the limiting factors on speed are the generation of the detection enable signal, d , and its propagation through the completion detection circuit. The minimum accumulation time is
300 ns + 210 ns = 510 ns.

For the theoretical average propagation length of 4. 91 (or 5) stages, the accumulation time is

340 ns for carry generation and insertion into carry propagation line

+ 5 × 60 ns for propagation through 5 stages

+ 30 ns for the delay of gate 5

+ 210 ns for completion circuit delay

= 880 ns .

Compared with the accumulation time of 6. 50 μs which must always be allowed in the accumulator previously described, this is an increase in speed of 7. 39 to 1 for the addition of a large number of summands with random bit distribution.

The circuit elements that must be added to the gated-carry accumulator to implement the carry-completion detection feature are three 11-input NAND gates, one 3-input NAND gate, four 2-input NAND gates, two flip-flops, and one delay multivibrator. From Table 3 it can be seen that the cost figure will increase by 62, yielding a total of 818. This is a 29. 8-percent cost increase over that of the simple iterative accumulator.

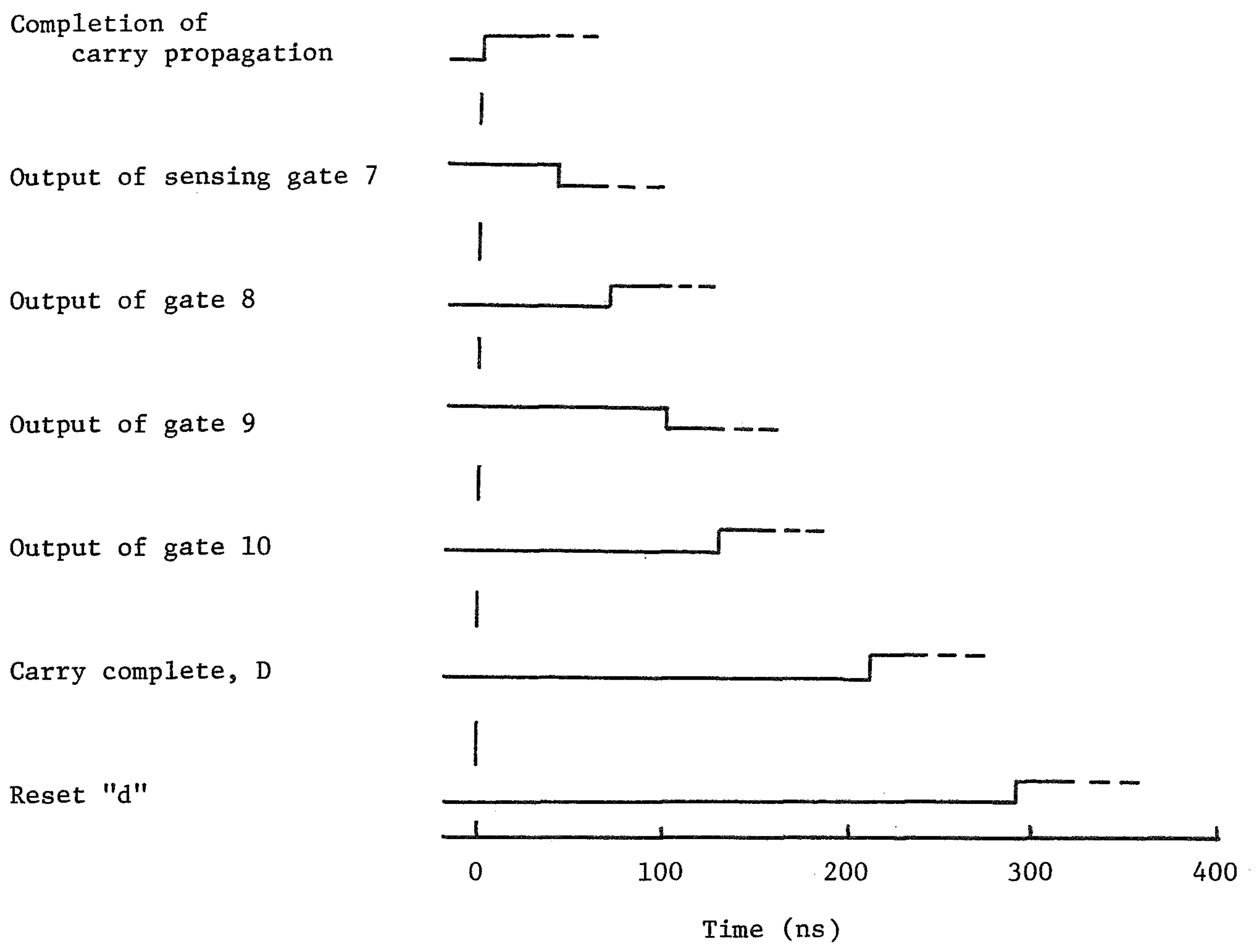


Figure 24. Timing diagram for carry-completion signal generation.

Carry-Skip Techniques

The accumulator under investigation uses ones complements representation of negative numbers with end-around carry, so a skip distribution using equal size groups is considered [6]. The accumulator of n bits is divided into k groups of m bits each so that

$$mk = n \tag{33}$$

The greatest carry propagation time results when the conditions for the summands are

$$x_1 \cdot y_1 \quad , \tag{34}$$

$$x_1 \oplus y_1 \ , \quad i = 2, 3, . . . , n-1, \tag{35}$$

and

$$\overline{x}_{30} \cdot \overline{y}_{30} \tag{36}$$

The time required for propagation is then

$$\begin{aligned} \tau' &= [1 + (m-1) + (k-2) + (m-1)] \text{ t. u.} \\ &= [2m + k-3] \text{ t. u.} \\ &= (2m + n/m-3) \text{ t. u.} \end{aligned} \tag{37}$$

where t. u. is the propagation time of two NAND gates [6]. Differentiating equation (37) with respect to m and equating to zero yields

$$2\text{-}n/m^2 = 0 \quad ,$$

$$m^2 = n/2 \quad ,$$

$$m^2 = \sqrt{n/2} \quad , \tag{38}$$

or

$$m^2 = km/2 \quad , \tag{39}$$

$$k \;= 2m$$

for the minimum propagation time. For $n = 30$, the approximations $m = 5$ and $k = 6$ are used. The group containing the lower-order bits is called group 1.

The maximum propagation time for a 30-bit accumulator is

$$\tau' = [1 + (5-1) + (6-1) + (5-1)] \times 60 \text{ ns}$$

$$= 780 \text{ ns}$$

The time required to generate a carry was 340 ns, and the time to assimilate a carry after it arrived at a stage was 120 ns. The maximum accumulation time for a 30-stage accumulator using one-layer carry-skips is then

$$A = (460 + \tau'_{30}) \text{ ns}$$

$$= 1.24 \ \mu\text{s} \quad .$$

Equations (11) through (16) will be rewritten here using notation for a five-stage group. Let

$$r_{k,i} = (x_{k,i} \oplus y_{k,i})\, c_{k,i} + x_{k,i} \cdot y_{k,i} \tag{40}$$

be the carry signal produced by the i^{th} stage of the k^{th} group, where $y_{k,i}$ and $x_{k,i}$ are the summands of the i^{th} stage of the k^{th} group and $c_{k,i}$ is the carry into that stage and

$$c_{k,i} = r_{k,i-1} \tag{41}$$

Let

$$t_{k,i} = x_{k,i} \oplus y_{k,i} \tag{42}$$

be the condition for the transmission of a carry through a stage. This signal is present at the true output of the k, i^{th} -stage accumulator register 110 ns after the accumulate pulse has ended (Fig. 23).

Let

$$g_{k,i} = x_{k,i} \cdot y_{k,i} \tag{43}$$

be the condition for the generation of a carry in the k, i^{th} stage. This signal is generated by the k, i^{th} DM when the complement output of the k, i^{th} accumulator register becomes a one. Figure 23 shows that $g_{k,i}$ is generated 280 ns after the end of the accumulate pulse. The carry produced by the k^{th} group is then

$$\begin{aligned} r_k = g_{k,5} &+ t_{k,5} \cdot g_{k,4} + t_{k,5} \cdot g_{k,3} \\ &+ t_{k,5} \cdot t_{k,4} \cdot t_{k,3} \cdot g_{k,2} \\ &+ t_{k,5} \cdot t_{k,4} \cdot t_{k,3} \cdot t_{k,2} \cdot g_{k,1} \\ &+ t_{k,5} \cdot t_{k,4} \cdot t_{k,3} \cdot t_{k,2} \cdot t_{k,1} \cdot c_k \end{aligned} \tag{44}$$

Equation (44) is satisfactory for carry levels, but with carry pulses 100 ns in width, Figure 25 shows that for $t_1 \cdot t_2 \cdot t_3 \cdot t_4 \cdot t_5$ true, each c_k pulse into a group will result in two r_k pulses out of that group. For pulse carries, equation (44) must then be changed to

$$
\begin{aligned}
r_k = {} & \overline{(t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})} \cdot (g_{k,5} \\
& + t_{k,5} \cdot g_{k,4} \cdot t_{k,5} \cdot t_{k,4} \cdot g_{k,3} \\
& + t_{k,5} \cdot t_{k,4} \cdot t_{k,3} \cdot g_{k,2} \\
& + t_{k,5} \cdot t_{k,4} \cdot t_{k,3} \cdot t_{k,2} \cdot g_{k,1}) \\
& + (t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})\, c_k \\
= {} & \overline{(t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})}\, g_{k,5} \\
& + \overline{(t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})} \cdot (t_{k,5} \cdot g_{k,4} \\
& + t_{k,5} \cdot t_{k,4} \cdot g_{k,3} + t_{k,5} \cdot t_{k,4} \cdot t_{k,3} \cdot f_{k,2} \\
& + t_{k,5} \cdot t_{k,4} \cdot t_{k,3} \cdot t_{k,2} \cdot g_{k,1}) \\
& + t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})\, c_k
\end{aligned}
\tag{45}
$$

The carries that must be produced by the stages within the group are described by the following equations:

$$
r_{k,1} = c_k \cdot t_{k,1} + g_{k,1} \tag{46}
$$

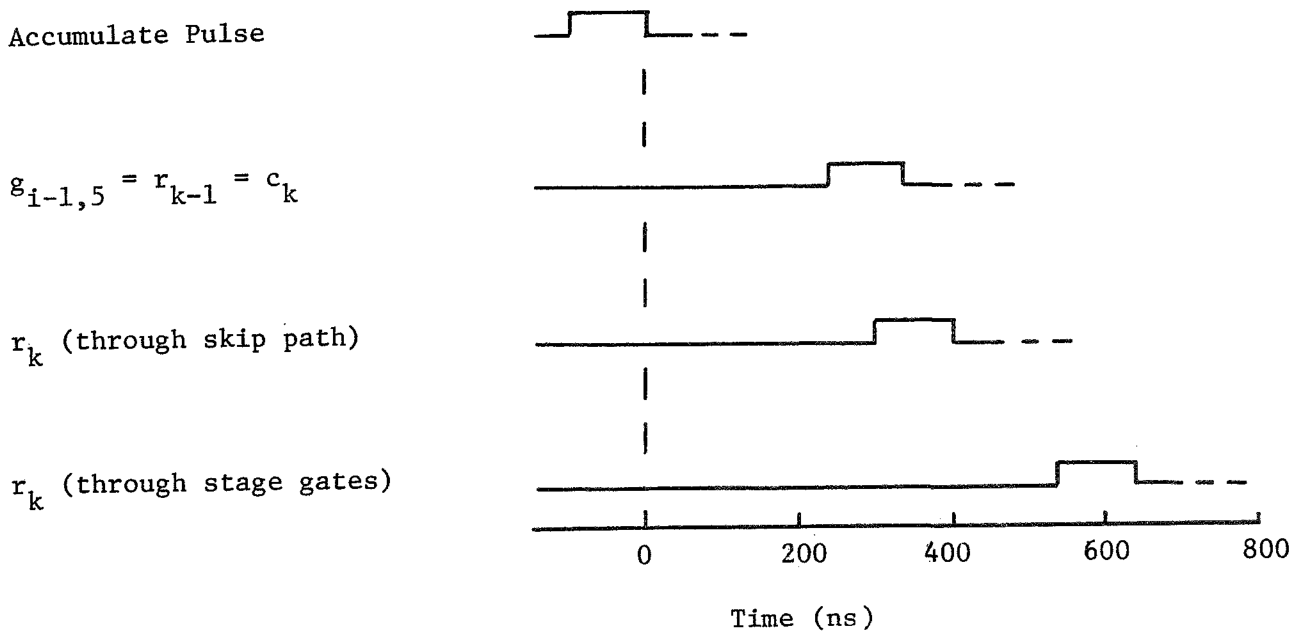


Figure 25. Carry propagation through group with $t_1 \cdot t_2 \cdot t_3 \cdot t_4 \cdot t_5$ true.

$$r_{k,2} = r_{k,1} \cdot t_{k,2} + g_{k,2}$$

$$= c_k \cdot t_{k,1} \cdot t_{k,2} + g_{k,1} \cdot t_{k,2} + g_{k,2} \qquad (47)$$

$$r_{k,3} = r_{k,2} \cdot t_{k,3} + g_{k,3}$$

$$= c_k \cdot t_{k,1} \cdot t_{k,2} \cdot t_{k,3} + g_{k,1} \cdot t_{k,2} \cdot t_{k,3}$$

$$+ g_{k,2} \cdot t_{k,3} + g_{k,3} \qquad (48)$$

$$r_{k,4} = r_{k,3} \cdot t_{k,4} + g_{k,4}$$

$$= c_k \cdot t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} + g_{k,1} \cdot t_{k,2}$$

$$t_{k,3} \cdot t_{k,4}$$

$$+ g_{k,2} \cdot t_{k,3} \cdot t_{k,4} + g_{k,3} \cdot t_{k,4} + g_{k,4} \qquad (49)$$

$$r_{k,5} = t_k \qquad (50)$$

A reduced block diagram of the five-stage group using NAND gates to produce the combinational functions in equations (46) through (50) is given in Figure 26. To show that this circuit does satisfy the required conditions, DeMorgan's theorem is applied to equations written from the diagram. The carry-generation enable level, e , is a necessary condition for $\overline{g}_{k,1}$ as explained in the section on gated carry, but will be omitted from the equations as it does not directly affect the argument.

$$r_{k,1} = \overline{\overline{c_k \cdot t_{k,1}} \cdot \overline{g}_{k,1}}$$

$$= c_k \cdot t_{k,1} + g_{k,1}$$

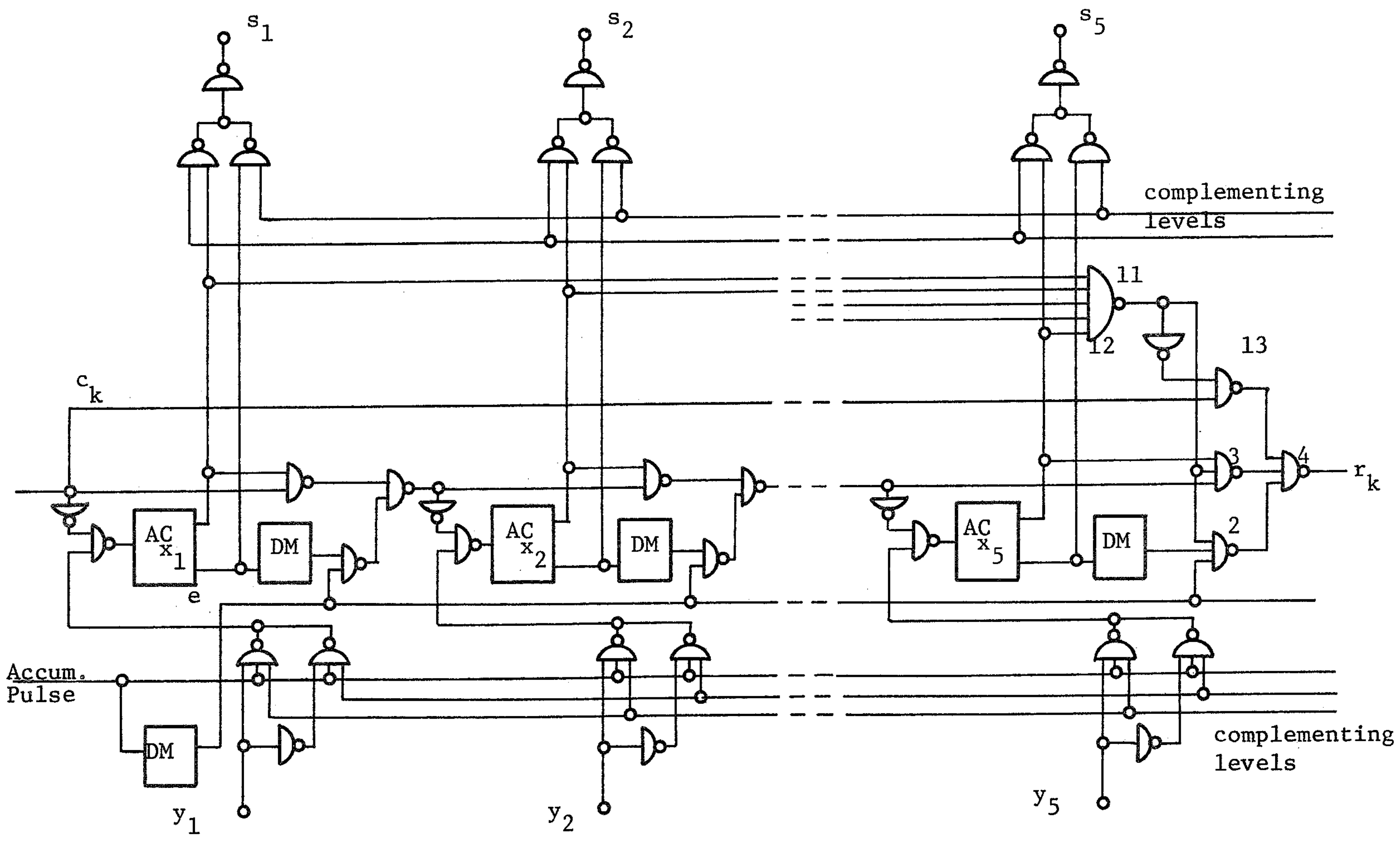


Figure 26. Carry skip group k.

$$
\begin{aligned}
r_{k,2} &= \overline{\overline{r_{k,1}\cdot t_{k,2}}\cdot\overline{g_{k,2}}} \\
&= r_{k,1}\cdot t_{k,2}+g_{k,2}
\end{aligned}
$$

$$
\begin{aligned}
r_{k,3} &= \overline{\overline{r_{k,2}\cdot t_{k,3}}\cdot\overline{g_{k,3}}} \\
&= r_{k,2}\cdot t_{k,3}+g_{k,3}
\end{aligned}
$$

$$
\begin{aligned}
r_{k,4} &= \overline{\overline{r_{k,3}\cdot t_{k,4}}\cdot\overline{g_{k,4}}} \\
&= r_{k,3}\cdot t_{k,4}+g_{k,4} \\
&= c_k\cdot t_{k,1}\cdot t_{k,2}\cdot t_{k,3}\cdot t_{k,4}+g_{k,1}\cdot t_{k,2} \\
&\quad \cdot t_{k,3}\cdot t_{k,4}+g_{k,2}\cdot t_{k,3}\cdot t_{k,4}+g_{k,3} \\
&\quad \cdot t_{k,4}+g_{k,4}
\end{aligned}
$$

It is seen that these equations are in agreement with equations (46) through (49).

The carry produced by the fifth stage of the group is

$$
r_{k,5} = r_k = \overline{\overline{g_{k,5}\cdot\overline{t_{k,1}\cdot t_{k,2}\cdot t_{k,3}\cdot t_{k,4}\cdot t_{k,5}}}\cdot\overline{r_{k,4}\cdot t_{k,5}\cdot\overline{(t_{k,1}\cdot t_{k,2}\cdot t_{k,3}\cdot t_{k,4}\cdot t_{k,5})}}\cdot\overline{c_k\cdot\overline{\overline{(t_{k,1}\cdot t_{k,2}\cdot t_{k,3}\cdot t_{k,4}\cdot t_{k,5})}}}}
$$

$$\begin{aligned}
&= g_{k,5} \cdot \overline{(t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})} \\
&\quad + r_{k,4} \cdot t_{k,5} \overline{(t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})} \\
&\quad + c_k \, (t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})
\end{aligned}$$

It must be shown that

$$\begin{aligned}
& r_{k,4} \cdot t_{k,5} \cdot \overline{(t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})} \\
&= \overline{(t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})} \cdot (g_{k,4} \cdot t_5 \\
&\quad + g_{k,3} \cdot t_{k,4} \cdot t_{k,5} + g_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5} \\
&\quad + g_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})
\end{aligned}$$

We see that

$$\begin{aligned}
t_{k,5} \cdot r_{k,4} = c \cdot {} & (t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot r_{k,4} \cdot t_{k,5}) \\
& + g_{k,4} \cdot t_{k,5} + g_{k,3} \cdot t_{k,4} \cdot t_{k,5} \\
& + g_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5} \\
& + g_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5}
\end{aligned}$$

Then

$$\overline{(t_{k,1}\cdot t_{k,2}\cdot t_{k,3}\cdot t_{k,4}\cdot t_{k,5})}\cdot t_{k,5}\cdot r_{k,4}$$

$$= c\cdot \overline{(t_{k,1}\cdot t_{k,2}\cdot t_{k,3}\cdot t_{k,4}\cdot t_{k,5})} \to 0$$

$$\cdot\, (t_{k,1}\cdot t_{k,2}\cdot t_{k,3}\cdot t_{k,4}\cdot t_{k,5})$$

$$+\, \overline{(t_{k,1}\cdot t_{k,2}\cdot t_{k,3}\cdot t_{k,4}\cdot t_{k,5})}\cdot (g_{k,4}\cdot t_{k,5}$$

$$+\, g_{k,3}\cdot t_{k,4}\cdot t_{k,5} + g_{k,2}\cdot t_{k,3}\cdot t_{k,4}\cdot t_{k,5}$$

$$+\, g_{k,1}\cdot t_{k,2}\cdot t_{k,3}\cdot t_{k,4}\cdot t_{k,5}) \qquad ,$$

which is the required result.

It is necessary that the proper gating levels be at gates 2, 3, and 13 of Figure 26 before a carry pulse reaches them. Figure 27 shows that the blocking level will be present 50 ns before $g_{k,5}$ reaches gate 2 and 90 ns before $r_{k,4}$ reaches gate 3. The gating level will be present 80 ns before c_k reaches gate 13.

The 30-stage accumulator is made up of six groups of five stages with $c_k = r_{k-1}$, except that $c_1 = r_6$, the end-around carry. This is shown in Figure 28.

The accumulator using carry skips yields a speed advantage of 6. 50 6. 50/1. 24 μs or 5. 21/1 over the accumulator previously described.

The components that must be added to the gated-carry circuit to facilitate carry skips are one 6-input NAND gate, one 2-input NAND gate, and one inverter for each group. In every fifth stage, two 2-input NAND gates must be replaced with 3-input NAND gates. The cost figure for the 30-stage accumulator with carry skips is then 816. This is an increase of 29. 5 percent over the cost of the simple iterative accumulator.

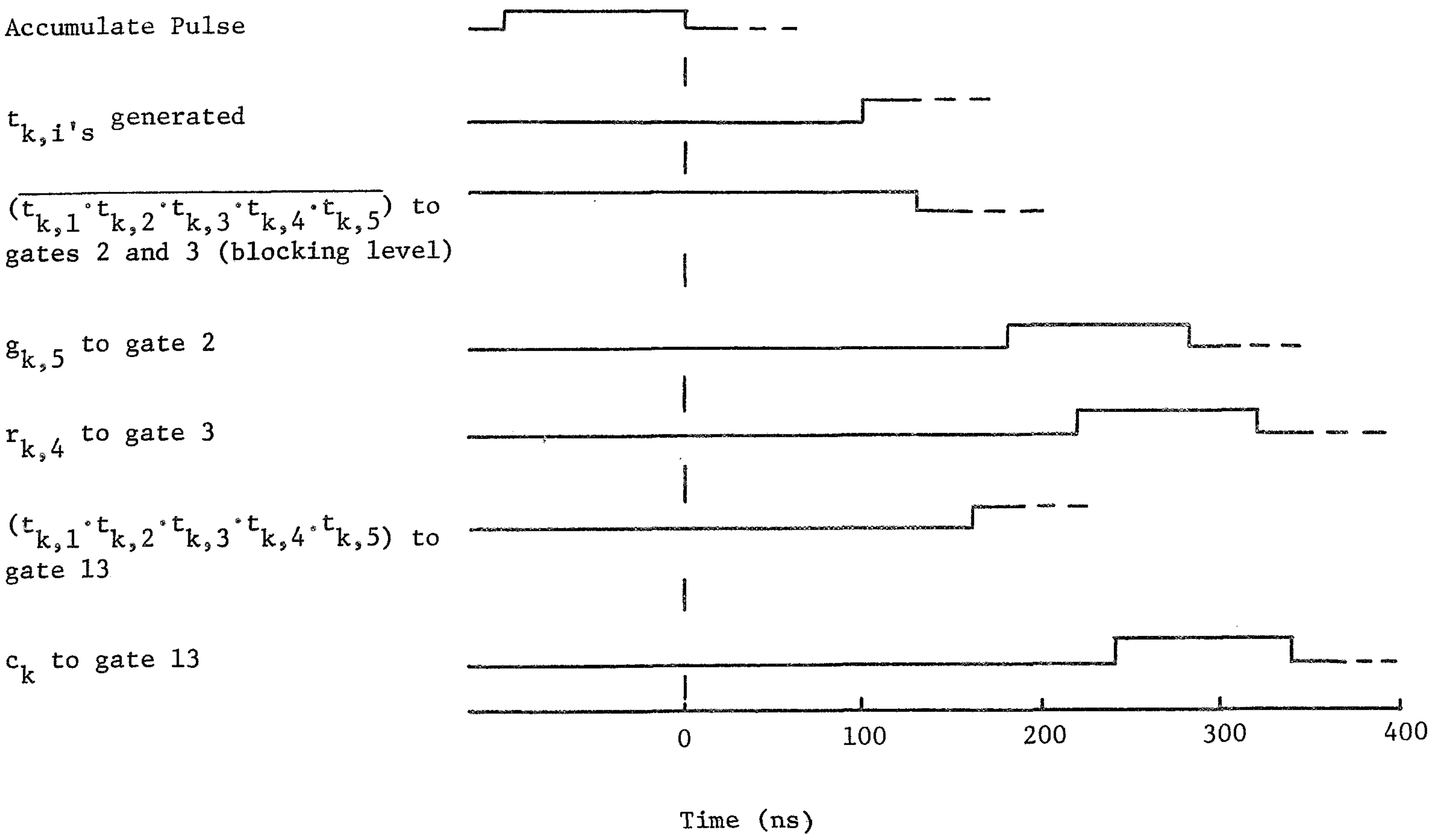

Accumulate Pulse
$t_{k,i}$'s generated
$\overline{(t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})}$ to gates 2 and 3 (blocking level)
$g_{k,5}$ to gate 2
$r_{k,4}$ to gate 3
$(t_{k,1} \cdot t_{k,2} \cdot t_{k,3} \cdot t_{k,4} \cdot t_{k,5})$ to gate 13
c_k to gate 13
0
100
200
300
400
Time (ns)

Figure 27. Gating level timing diagram.

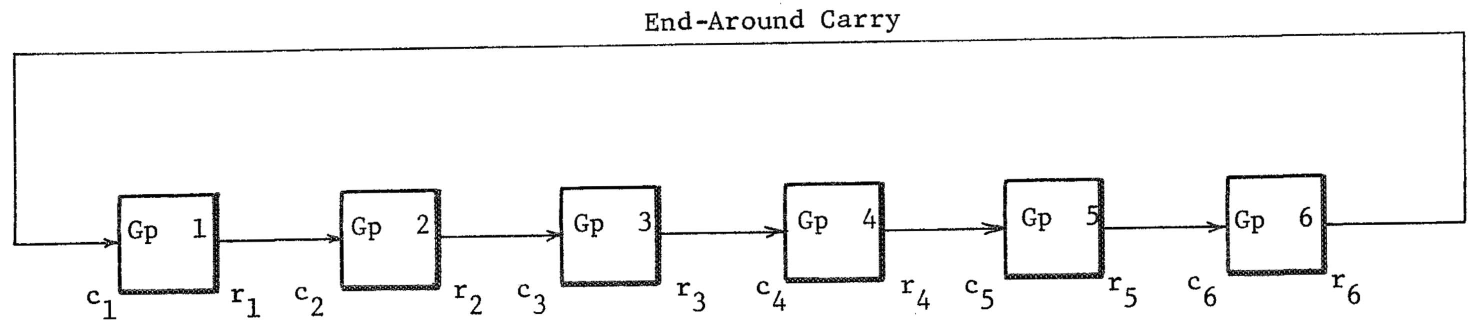


Figure 28. Six-group representation.

DISCUSSION AND CONCLUSIONS

The relative cost figures and accumulation times for the simple iterative accumulator and accumulators using fast carry techniques are presented in Table 4. It is seen that all three realizations investigated offer a considerable decrease in accumulation time without too great a difference in cost figure.

TABLE 4. ACCUMULATOR COST AND TIME REQUIREMENTS

Accumulator Design	Cost Figure	Max Accum Time (μs)	Avg Accum Time (μs)
Simple iterative accumulator	630	6. 50	6. 50
Accumulator with gated carry	756	2. 14	2. 14
Accumulator with carry-completion detection	818	2. 26	0. 88
Accumulator with carry skips	816	1. 24	1. 24

The comparison may be formalized by the use of a criterion similar to that used by Burla [5] for parallel adders. The criterion is

$$Q = t/t_0 \cdot F/F_0 \cdot 100 \qquad (51)$$

where

t = accumulation time of the circuit under consideration

t_0 = accumulation time of the iterative accumulator

F = cost figure of the network under consideration

F_0 = cost figure of the iterative accumulator.

The results of applying this criterion are shown in Table 5. Using this criterion, the circuit for carry-completion detection appears best. However, the control circuitry for it is more complicated than for the others considered. The average accumulation time was calculated using carry propagation lengths resulting from the accumulation of numbers with random bit distribution. If this distribution is not random, the accumulation time for the carry-completion detection circuit could be greater than that for the carry-skip circuit.

TABLE 5. COMPARISON OF ACCUMULATOR DESIGNS

Accumulator Design	t (μs)	$t/t_0 \times 100$	F	F/F_0	Q
Simple iterative accumulator	6. 50	100	630	1	100
Accumulator with gated carry	2. 14	33	756	1. 2	40
Accumulator with carry-completion detection	(Avg) 0. 880	14	818	1. 3	18
Accumulator with carry skips	1. 24	19	816	1. 3	25

In some applications the gated-carry circuit might be desirable. If the accumulation time for it is acceptable, it has the advantage of having fewer components.

The gated-carry circuit offers the fastest accumulation time for fixed-time operation. In general, the carry-completion detection circuit would be recommended for application when a variable accumulation rate is acceptable; the carry-skip circuit would be recommended when a fixed accumulation rate is desired.

APPENDIX

DETERMINATION OF OPTIMAL AND ECONOMICAL SKIP DISTRIBUTIONS [12]

A discussion of Majerski's methods for determining optimal and economical skip distributions in an adder with a maximum possible number of stages for a given maximum allowable propagation time will require the introduction of several definitions and notational conventions. One-layer skip distribution occurs when adder stages are included in carry skips, and every stage is included at most in one skip.

Two-layer skip distribution occurs when adder stages are included in two adder skips, and every stage is included at most in two skips. If two skips include common stages, all stages contained in the shorter skip are included in the longer skip. The longer skip is called first-layer skip, and the shorter one is called second-layer skip. If none of the stages in a skip are contained in another skip, the skip is treated as a first-layer skip.

An adder with one-layer skip distribution is divided into groups, each of which includes all stages in a skip circuit or a single stage not contained in a skip.

An adder with two-layer skip distribution is divided into sections, which include all stages composing one first-layer skip or a single stage not included in skips. A section is divided into groups. A group includes all stages in a second-layer skip or a single stage in a second-layer skip. Sections and groups are numbered successively, starting with the least significant bits.

The notation used is:

n = Number of adder stages

m = Number of groups in an adder with one-layer skip distribution

k_i, $(i = 1, 2, \ldots, m)$ = Number of stages of the i^{th} group of an adder with one-layer skip distribution

p = Number of skips in an adder with one-layer skip distribution

M = Number of sections in an adder with two-layer skip distribution

K_i, $(i = 1, 2, \ldots, M)$ = Number of stages in the i^{th} section of an adder with two-layer skip distribution

P = Number of first-layer skips in an adder with two-layer skip distribution

m_i, $(i = 1, 2, \ldots, M)$ = Number of groups in the i^{th} section of an adder

k_{ij}, $(i = 1, 2, \ldots, M); j = 1, 2, \ldots, m_i)$ = Number of stages in the j^{th} group of the i^{th} section of an adder

p_i, $(i = 1, 2, \ldots, M)$ = Number of second-layer skips in the i^{th} section of an adder

d = Maximum propagation time of a NOR gate

T = Maximum carry propagation time (mcpt); the greatest carry propagation time for all possible combinations of adder input variables.

The optimal skip distribution in an n-stage adder is defined as occurring when, for a given n, T is minimal and the number of skips is minimal. The economical skip distribution is defined as occurring when, for a given n and T, the number of skips is minimal.

The mean carry propagation time (mcpt) in the NOR gate adder with one-layer skip distribution is given as:

T = maximum (T_1, T_2), where

$$T_1 = \underset{\alpha,\beta}{\text{maximum}} \left[(k_\alpha - 2) + (\beta - \alpha - 1) + k_\beta \right] \qquad \text{(A-1)}$$

$$T_2 = (m - 1) + \underset{\alpha}{\text{maximum}}\ (k_\alpha - 1) \qquad \text{(A-2)}$$

and where the α, β indexes of groups satisfy the conditions

$$1 \le \alpha \le m,\ \alpha < \beta < m + \alpha \text{ and } k_{m+i} + k_i \qquad \text{(A-3)}$$

The terms $(k_\alpha - 2)$, k_β, and $\underset{\alpha}{\text{maximum}}\ (k_\alpha - 1)$ denote maximum times of carry propagation through positions of the groups in which the carry propagation begins and ends. The -2 in the term $(k_\alpha - 2)$ is due to the carry propagation time beginning with the change of the carry control signal rather than at the change of states of the adder inputs.

For the optimal skip distribution in an n(T)-position adder, the maximum number of adder positions is the function of mcpt, $T \ge 5$, given by the formulas

$$\nu_T = \left(\frac{T + 7}{4}\right)^2 \qquad \text{(A-4)}$$

$$n^{(T)} = 2E\,\frac{(\nu_T)}{2} = 2E\left[\frac{(T + 7)^2}{32}\right] \qquad \text{(A-5)}$$

where ν_T is defined by equation (A-4).

For odd values of T for which T/8 does not produce a remainder equal 3, the optimal skip distribution is the one in which the numbers of stages in groups are k and 1 alternately. For this case, the following formulas apply:

$$k = 2E\left[\frac{1}{2}\left(K_T' + 1 + \sqrt{\nu_T - n^{(T)}}\right)\right] - 1 \quad , \qquad \text{(A-6)}$$

$$K_T' = \frac{T + 3}{4} \quad , \qquad \text{(A-7)}$$

$$m = T - 2k + 5 \text{ (m . . . even number)} \qquad \text{(A-8)}$$

For a 36-stage adder, the values given by these formulas are a minimal value of $T = 17$ and a skip distribution of stages as 5, 1, 5, 1, 5, 1, 5, 1, 5, 1, 5, 1.

For the values of T such that $T/8$ gives a remainder of 3, the optimal skip distribution is the one with the numbers of stages in successive fours of groups equaling $K-2$, 1, k, 1. In this case, the following formulas are used:

$$k = 2E\left[\frac{1}{2}\left(K_T'' + 1 + \sqrt{\nu_T - n^{(T)}}\right)\right] - 1 \quad , \qquad \text{(A-9)}$$

$$K_T'' = \frac{T + 7}{4} \quad , \qquad \text{(A-10)}$$

$$m = T - 2K + 7 \text{ (m . . . number divisible by 4) .} \qquad \text{(A-11)}$$

To determine an economical skip distribution, the distributions for two adders are computed and the one with the smaller number of skips is chosen.

The first adder, with k and 1 stages in groups alternately, is determined by formulas (A-4), (A-6), (A-7), and (A-8), with n replacing $n^{(T)}$ in (A-6). An adder with $(m/2)$ $(k + 1)$ stages is obtained. If $(m/2)$ $(k + 1)$ is greater than n, $(m/2)$ $(k + 1) - n$ stages are canceled. This should be done in pairs in groups and preferably in such a way as to cancel one skip.

For the adder with $k-2$, 1, k, 1 positions in successive fours of groups, formulas (A-4), (A-5), (A-9), (A-10), and (A-11), with n replacing $n^{(T)}$ in (A-10), apply. The number of stages obtained is

$$m' (k - 1) + m'' (k + 1) \quad ,$$

where

$$m' = E\left(\frac{m + 2}{4}\right),\ m'' = E\ (m/4) \tag{A-12}$$

The adder with the smaller number of skips is then the one with economical skip distribution.

The mcpt for an adder with two-layer skip distribution is given by

$$T = \text{maximum}\ (T_1,\ T_2,\ T_3,\ T_4) \quad ,$$

where

$$T_1 = \underset{\phi,\psi,\alpha,\beta}{\text{maximum}} \left\{ \left[(k_{\phi\alpha} - 2)\ (m_\phi - \alpha)\right] + (\psi - \phi - 1) + \left[(\beta - 1) + k_{\psi\beta}\right] \right\} \tag{A-13}$$

$$1 \leq \phi \leq M,\ \phi < \psi < M + \phi,\ 1 \leq \alpha \leq m_\phi,\ 1 \leq \beta \leq m_\psi,\ \text{and}\ k_{M+i,j} = k_{ij} \tag{A-14}$$

$$T_2 = \underset{\phi,\alpha,\beta}{\text{maximum}} \left[(k_{\phi\alpha} - 2) + (\beta - \alpha - 1)\ k_{\phi\beta}\right] \tag{A-15}$$

$$1 \leq \phi \leq M,\ 1 \leq \alpha < \beta \leq m_\phi \tag{A-16}$$

$$T_3 = (M - 1) + \underset{\phi,\ \alpha}{\text{maximum}} \left[(m_\phi - 1) + (k_{\phi\alpha} - 1)\right] \tag{A-17}$$

$$1 \leq \phi \leq M, \ 1 \leq \alpha \leq m_{\phi} \tag{A-18}$$

$$T_4 = (M - 1) + \underset{\phi, \alpha, \beta}{\text{maximum}} \left\{ \left[(\alpha - 1) + k_{\phi\alpha} \right] + \left[(k_{\phi\beta} - 2) + (m_{\phi} - \beta) \right] \right\} \tag{A-19}$$

$$1 \leq \phi \leq M, \ 1 \leq \alpha \leq \beta \leq m_{\phi} \quad . \tag{A-20}$$

The methods presented here apply exclusively to the NOR gate adders with an even number of stages, in which T always takes odd values.

In an adder composed of an even number of M sections, having K and 1 stages alternately, with identical skip distributions in multistage sections, T may be expressed as:

$$T = t + M - 3 \quad (t \text{ an even number}) \ , \tag{A-21}$$

where t is a function of the number of stages and of the skip distribution in a section. Formula (A-21) defines t .

Optimal skip distribution in a section occurs when, for a given t , the number $K^{(t)}$ positions in a section is a maximum and, with maximum $K^{(t)}$, the number of skips in the section is a minimum. The parameters for such a distribution are given by the following formulas:

$$m^{(t)} = 2E\left(\frac{t}{6}\right) + 1 \qquad \text{if } t \geq 16 \tag{A-22}$$

$$m^{(t)} = 2E\left(\frac{t + 4}{6}\right) + 1 \qquad \text{if } t < 16 \tag{A-23}$$

$$k_j^{(t)} = \frac{1}{2}\left[1 + (-1)^j\right] + \frac{1}{2}\left[1 - (-1)^j\right] \times$$

$$\text{minimum}\left[2E\left(\frac{t+4}{4}\right) - j \text{ , } 2E\left(\frac{t - 2m^{(t)} + 4}{4}\right) + j\right]$$

$$\text{for } j = 1, 2, \ldots, m^{(t)} \quad , \tag{A-24}$$

$$K^{(t)} = 2E\left[\frac{(2t - 3m^{(t)})\, m^{(t)} + 2t + 6m^{(t)}}{16}\right] + 1 \tag{A-25}$$

The parameters $m^{(t)}$, $k_j^{(t)}$, $K^{(t)}$ take exclusively odd values.

Majerski found for $T = 13$, $t = 8$, $t_i = 8, 1, 8, 1, 8, 1, 8, 1$, $n^{(T)} = 32$, and for $T = 15$, $t = 10$, $t_i = 12, 1, 8, 1, 12, 1, 8, 1$, $n^{(T)} = 34$.

The optimal skip distribution in an $n^{(T)}$ - position adder with the maximum number of positions for a given $T \geq 9$ may be determined as follows:

$$\frac{2T - 6}{3} \leq t \leq \frac{2T + 9}{3} \tag{A-26}$$

$$M = T - t + 3 \tag{A-27}$$

For $t = 6$ and $t = 8$, 12, 16, . . . , the adder is made up of sections of $K^{(t)}$ and 1 stages alternately. The number of adder stages is

$$n' = \frac{1}{2} M\left(K^{(t)} + 1\right) \tag{A-28}$$

For $t = 10, 14, 18, \ldots$, successive adder sections are taken according to the parameters given by the terms of sequences

$$t+2,\ 1,\ t-2,\ 1,\ t+2,\ 1,\ t-2,\ 1,\ \ldots\ ,\ t+2,\ 1,\ t-2,\ 1 \tag{A-29}$$

if M is a number divisible by 4 and

$$t+2,\ 1,\ t-2,\ 1,\ t+2,\ 1,\ t-2,\ 1,\ \ldots\ ,\ t+2,\ 1,\ t-2,\ t,\ 1 \tag{A-30}$$

if M is a number not divisible by 4.

The 1 terms in formulas (A-29) and (A-30) correspond to one-stage sections. A cyclic change of adder stages is permitted. The number of adder stages is then

$$n'' = E\left(\frac{M}{4}\right)\left(K^{(t+2)} + K^{(t-2)} + 2\right) + \frac{1}{2}\left[1 - (-1)^{M/2}\right]\left(K^{(t)} + 1\right). \tag{A-31}$$

From among the skip distributions for all t obtained from formula (A-26), the one with maximum $n^{(T)}$ stages and for $n^{(T)}$ with minimum number of skips is chosen.

For an economical skip distribution for an n-stage adder with $t \geq 9$, the skip distribution is determined for two adders with

$$t = 2E\ (T/3),\ 2E\ (T/3) + 2,\ 2E\ (T/3) + 4,\ \ldots$$

In the first adder, accept $K^{(t)}$ and 1-stage sections alternately. The number of adder sections and the number of adder stages are given in formulas (A-27) and (A-28).

In the second adder (only if $t \geq 8$), accept the same number of adder sections, but for successive sections use instead of t the successive terms of the sequences

$$t+2,\ 1,\ t-2,\ 1,\ \ldots\ ,\ t+2,\ 1,\ t-2,\ 1$$

if M is a number divisible by 4, (A-32)

or

$$t+2,\ 1,\ t-2,\ 1,\ t+2,\ 1,\ t-2,\ 1,\ \ldots\ ,\ t+2,\ 1,\ t-2,\ 1,\ t,\ 1 \qquad \text{(A-33)}$$

if M is a number not divisible by 4 and $t = 10, 14, 18, \ldots$

or

$$2,\ 1,\ t+2,\ 1,\ t-2,\ 1,\ t+2,\ 1,\ \ldots\ ,\ t-2,\ 1,\ t+2,\ 1,\ t,\ 1$$

if M is a number not divisible by 4 and $t = 8, 12, 16 \ldots$.

Such skip distributions are determined for at least three successive integers t and for further t until both numbers n' and n'' are smaller than n . Then for all skip distributions in the adders for which $n' > n$ or $n'' > n$, cancel $n' - n$ or $n'' - n$ stages to cancel the greatest number of skips. From among the obtained skip distributions of n-position adders, the one with the minimum number of skips is chosen.

REFERENCES

1. Carroll, C. C.; et al: On the Realization of a Generalized Second-Order Digital Compensator, Technical Report Nr 9, NAS8-11274, Auburn Research Foundation, Auburn, Ala., September 1967.

2. Carroll, C. C.; and Nagle, H. T.: A Special-Purpose Computer Realization of a Third-Order Digital Filter for the PIGA Control Loop. Technical Report Nr 9, NAS8-20163, Auburn Research Foundation, Auburn, Ala., May 1968.

3. Sklansky, J.: An Evaluation of Several Two-Summand Binary Adders, IRE Transactions on Electronic Computers, vol. EC-9, June 1960, pp. 213-226.

4. MacSorely, O. L.: High-Speed Arithmetic in Binary Computers. Proceedings of IRE, vol. 49, January 1961, pp. 67-91.

5. Lehman, M.; and Burla, N.: Skip Techniques for High-Speed Carry-Propagation in Binary Arithmetic Units. IRE Transactions on Electronic Computers, vol. EC-10, December 1961, pp. 691-698.

6. Gilchrist, B; Pomerene, J. H.; and Wong, S. Y.: Fast-Carry Logic for Digital Computers. IRE Transactions on Electronic Computers, vol. EC-4, December 1955, pp. 133-136.

7. Chu, Yoahan: Digital Computer Design Fundamentals. McGraw-Hill, Inc. (New York), 1962.

8. Burks, A. W.; Goldstine, H. H.; and Neumann, J. Von: Preliminary Discussion of the Logical Design of an Electronic Computing Instrument. The Institute for Advanced Study (Princeton, N. J.), 1947.

9. Reitweisner, G. W.: The Determination of Carry Propagation Length for Binary Addition. IRE Transactions on Electronic Computers, vol. EC-9, March 1960, pp. 35-38.

10. Hendrickson, H. C.: Fast High-Accuracy Binary Parallel Addition. IRE Transactions on Electronic Computers, vol. EC-9, December 1960, pp. 465-469.

REFERENCES (Concluded)

11. Majerski, S.; and Wiweger, M.: NOR-Gate Binary Adder with Carry Completion Detection. IEEE Transactions on Electronic Computers, vol. EC-16, February 1967, pp. 90-92.

12. Majerski, S.: On Determination of Optimal Distributions of Carry Skips in Adders. IEEE Transactions on Electronic Computers, vol. EC-16, no. 1, February 1967, pp. 90-92.

13. Instruction Manual, μ-Pac Integrated Circuit Modules. Honeywell Computer Control Division, September 1966.

BIBLIOGRAPHY

Bartee, T. C.; and Chapman, D. J.: Design of An Accumulator for a General Purpose Computer. IEEE Transactions on Electronic Computers, vol. EC-14, no. 4, August 1965, pp. 570-575.

McCluskey, E. J.: Introduction to the Theory of Switching Circuits. McGraw-Hill Book Co. (New York), 1965.

Miller, R. E.: Switching Theory (Combinatorial Circuits). John Wiley and Sons (New York), vol. I, 1965.

Morgan, C. P,; and Jarvis, D. B.: Transistor Logic Using Current Switching and Routing Techniques and Its Application to a Fast-Carry Propagation Adder. Proceedings IEE (London), vol. 106B, September 1959, pp. 477-478.

Prather, Ronald E.: Introduction to Switching Theory: A Mathematical Approach. Allyn and Bacon, Inc. (Boston), 1967.

Specialist discussion meeting on new digital computer techniques — Special Aspects of Logical Design. Proceedings IEE (London), vol. 106B, September 1959, pp. 462-469.

Weinberger, A.; and Smith, J. L.: A Logic for High-Speed Addition. Natl. Bur. Standards (Washington, D. C.), Circular 591, Sec. 1; February 1958.

TM X-53983

APPROVAL

FAST CARRY ACCUMULATOR DESIGN

By William C. Mastin

The information in this report has been reviewed for security classification. Review of any information concerning Department of Defense or Atomic Energy Commission programs has been made by the MSFC Security Classification Officer. This report, in its entirety, has been determined to be unclassified.

This document has also been reviewed and approved for technical accuracy.

Bobby F. Walls
BOBBY F. WALLS
Chief, Optical Sensors Section

Carl H. Mandel
C. H. MANDEL
Chief, Guidance and Control Division

P H Broussard Jr
P. H. BROUSSARD, JR.
Chief, Sensors Branch

F. B. Moore
F. B. MOORE
Director, Astrionics Laboratory

DISTRIBUTION

INTERNAL

DIR
DEP-T
AD-S
PM-PR-M

S&E-DIR

S&E-CSE-DIR
Dr. Haeussermann

S&E-ASTR-DIR
Mr. Moore

S&E-ASTR-A
Mr. Hosenthien
Miss Flowers

S&E-ASTR-G
Mr. Mandel
Dr. Doane
Mr. Wood
Mr. Broussard
Mr. Walls
Mr. Jones
Mr. Doran
Mr. Mastin (3)
Mr. Carter
Mr. Turner
Mr. Richards

S&E-ASTR-S
Mr. Wojtalik

S&E-ASTR-C
Mr. Swearingen
Mr. White

S&E-ASTR-ZX

A&TS-MS-IL (8)

A&TS-MS-IP (2)

A&TS-MS-H

A&TS-PAT
Mr. Wofford

A&TS-TU (15)
Mr. Winslow

EXTERNAL

Dr. C. C. Carrol (2)
Dept. of Electrical Engineering
Auburn University
Auburn, Alabama

Scientific and Technical Information
Facility (2)
P. O. Box 33
College Park, Maryland 20740
Attn: NASA Representative (S-AK/RKT)