TECHNICAL REPORT ECOM-0206-S1

EC-RD
June 1967

LINEAR SEQUENCE GENERATORS: A NOTE ON METHODS FOR OBTAINING ANY PHASE SHIFT OF LINEAR SEQUENCES

C. E. L. Technical Report No. 8627-184

Contract No. DAAB 07-67-C0206
DA Project No. 5A6-79191 D902-40-45

Prepared by
T. G. Birdsall
C. C. Hoopes

COOLEY ELECTRONICS LABORATORY
Department of Electrical Engineering
The University of Michigan
Ann Arbor, Michigan

For

U. S. ARMY ELECTRONICS COMMAND, FORT MONMOUTH, N. J.

Technical Report ECOM-0206-S1 June 1967

ERRATA - August 1967

The following corrections are applicable to Technical Report ECOM-0206-S1, Linear Sequence Generators: A Note on Methods for Obtaining Any Phase Shift of Linear Sequences, June 1967:

Page 6

Change the second paragraph, line 5, to read, "$x_r(i)$ = contents of the r stage of the control generator (counted from right to left) . . . " r

Page 9

Change lines 9 and 10 to read, "The shaded regions in the abbreviated diagram for the MSRG (and MCRG) represent mod-2 adders placed between the corresponding stages whose two inputs are the contents of the last stage . . . "

Page 13

Change first equation under Proof, From Eq. 15, to read

$$u(i, j) = M^T(i)\,[R\,S(j)] = [A_M^i M(0)]^T R[A_S^j S(0)] \qquad (\text{mod-}2)$$

Page 16

Change first equation to

$$u(i, j) = \sum_{r=1}^{n} m_{n+1-r}(0)\, s_r(i + j) \qquad (\text{mod-}2)$$

Page 23

Change second paragraph to read, "The proof of Corollary 4 follows from the steps used in the proof of Corollary 2 and Theorem 2."

Page 25

Change equation 63 to read

$$u(i, j) = [M^T(0)R]\,S(p'j + i) \qquad (\text{mod-}2)$$

$$= s_n(p'j + p'k' + i)$$

ABSTRACT

Three techniques for advancing the phase of a linear binary sequence are described. The first employs a simple shift register generator and its equivalent modular shift register generator. The second, in a similar arrangement, employs a simple complement register generator and its equivalent modular complement register generator. The third technique described incorporates two identical Jacobi-hybrid generators.

A form of inflexible giant stepping is also discussed. By combining in the manner described a simple complement register generator and a modular shift register generator, one can advance the sequence phase in steps of an integer number of bits where the step size is a function of the generators employed. This same function can be accomplished by employing a modular complement register generator and a simple shift register generator.

Examples and applications of these techniques are included.

FOREWORD

This report was prepared by the Cooley Electronics Laboratory, The University of Michigan, Ann Arbor, on Contract No. DAAB07-67-C0206. The work was administered under the direction of the U. S. Army Electronics Command, Fort Monmouth, New Jersey. The report completes one phase of the work performed on this contract.

Mr. Lawrence A. Pick served as the technical representative for the U. S. Army Electronics Command. His assistance and guidance is gratefully acknowledged.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

1. INTRODUCTION

It is well known that if a finite length linear binary generator is used to produce a stream of binary digits by being clocked through successive states, then the sequence of digits so produced obeys a set of relations or equations among the output digits. The shortest of these relations (spanning the fewest number of digits) is known as the "characteristic sequence law." If the generator is a "maximal" generator, then the output sequence produced is the only sequence (excluding the all zero sequence) obeying its characteristic sequence law. Whereas if the sequence generator is "nonmaximal," there will be several sequences obeying the same characteristic sequence law. It is also well known that if several shifted versions of the sequence(s) that can be generated by a given generator are added (mod-2), then the resultant binary sequence also obeys the same characteristic sequence law.

In a maximal sequence generator each stage of the generator produces a shifted version of the same sequence. If one desires to generate an output sequence at some phase (time shift) other than that produced by any of the individual stages of the generator, then one need only add together, mod-2, the contents of appropriately selected stages. The determination of the appropriate stages to be summed for a particular desired time shift is the subject of this report.

This problem has been solved for several types of generators and the solution has been phrased in at least two different forms. It is the aim of this report to summarize and consolidate these results. The notation used is that of Ref. 1. This present report was stimulated by the work of Ref. 2. For simple shift-register generators (SSRG) the problem of determining the appropriate stage contents to be summed is solved by using another type of generator, the modular shift-register generator (MSRG). The basic work for this was shown in Ref. 3, and is repeated in Ref. 1. Schmidt (Ref. 2) points out that the converse is true; that is, the solution of the problem for the modular shift-register generator can be solved by using the simple shift-register generator. It is this duality pointed out by Schmidt that stimulated the thought producing the rather simple theorem exhibited in this report. This theorem is also extended to the simple complement register generator (SCRG) and modular complement-register generator (MCRG) which form a similar pair of problem solution generators. It is noted that the two Jacobi-hybrid generators, which have the same maximal characteristic equation, also form a problem solution pair.

In the following sections reference is made to "equivalent generators." For the purposes of this report two generators are equivalent if they have the same characteristic seqeunce law (Ref. 1 or 3).

2. THE PROBLEM

There are two versions of this problem which are related but which are attacked somewhat differently.

One such problem is as follows: Given a specific generator, select the appropriate register stages to be summed mod-2 in order to produce a given n tuple of digits following the point at which the basic generator contains the elementary load, E_1 (a one in the first stage and zeros in all other stages). This problem is solved and discussed in detail for the SSRG, MSRG, SCRG, and MCRG. The solution is unknown for the JHG.

The second version of the problem and the subject of this paper, is diagrammed in Fig. 1. In Fig. 1 an n stage binary generator, consisting of n stages of binary storage and associated interstage logic, produces the output binary stream y(j) from the stage designated as the last stage. The set of switches, $\{c_r\}$, selects the output from some of the register stages as the input to a mod-2 adder. The goal is to produce at the adder output the same sequence as that produced by the basic generator but advanced k digits. The shift k may be either positive or negative. A convenient notation for describing generators and combiners is to use vector and matrix notation. The contents of the register at time j is written as a column vector, Y(j). The switch positions are also written as a column vector, C , where the convention is that, if a switch is open, so that a particular stage is not active in the addition, then the coefficient is written as a zero, otherwise the switch is closed, so that

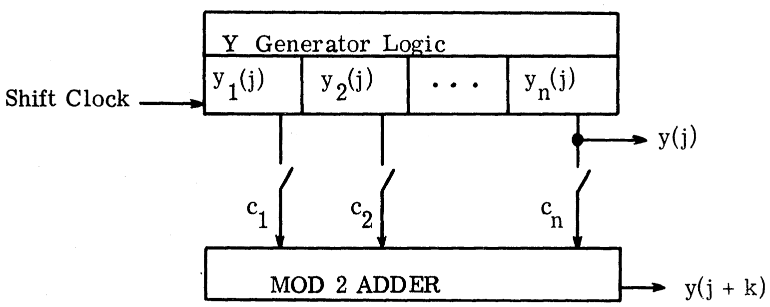


Fig. 1. Static combiner block diagram, using switches

the coefficient is written as a <u>one</u>. Thus, from the notation of Fig. 1 the content column vector Y(j) becomes

$$Y(j) = \begin{bmatrix} y_1(j) \\ y_2(j) \\ \vdots \\ y_n(j) \end{bmatrix} \tag{1}$$

where $y_r(j)$ = binary contents of stage r of the register at time j. The switch column vector C has the form

$$C = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \tag{2}$$

where $c_r = 1$ if the r-th stage of the register is fed into the adder

$= 0$ otherwise

The output sequence y(j) in Fig. 1 is taken from the last stage of the register,

$$y(j) = y_n(j)$$

The adder output sequence y(j + k) is expressed as

$$y(j + k) = \sum_{r=1}^{n} c_r\, y_r(j) \qquad \text{(mod-2)} \tag{3}$$

which in vector notation, using Eqs. 1 and 2, reduces to

$$y(j + k) = C^T\, Y(j) \quad \text{(mod-2)} \tag{4}$$

where C^T = "transpose of C" = $[c_1, c_2, \ldots, c_n]$

In practice, binary logic may be used instead of switches. The set of coefficients, $\{c_r\}$, in this case is usually considered a "control word" and a block diagram (equivalent to Fig. 1) for the static combiner <u>using binary logic</u> is diagrammed in Fig. 2. In Fig. 2 the switches of Fig. 1 have been replaced by AND gates whose outputs equal $y_r(j)$ when $c_r = 1$ or equal <u>zero</u> when $c_r = 0$.

If the control word is changed relatively frequently, the combiner will be designated

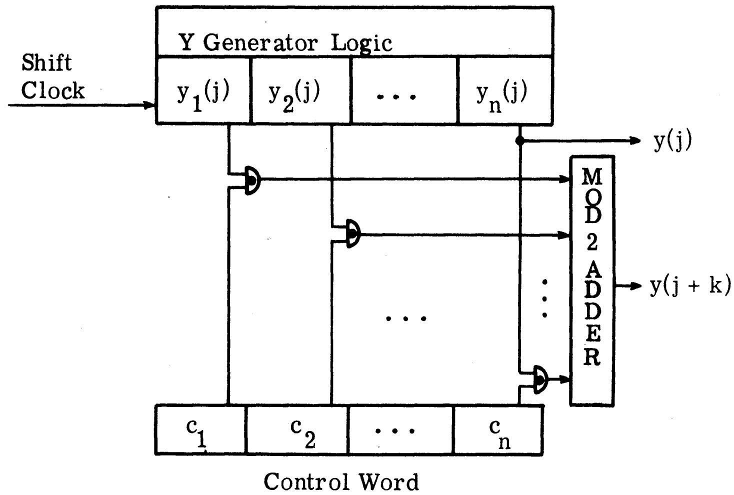


Fig. 2. Static combiner block diagram, using binary logic

a "dynamic combiner." The particular form of dynamic combiner of interest in this report is when the control word is generated by a "control" linear binary generator. For simplification in the subsequent work, it is assumed that the "control generator" is inverted with respect to the main generator, that is, the last stage of the control generator controls whether the first stage of the basic generator will be used in the combiner, and so on down to the first stage of the control generator which controls whether the last stage of the basic generator will be used in the combiner. A block diagram of this form of dynamic combiner is shown in Fig. 3. Note that in Fig. 3 the control generator X is reversed and shifts from right to left. The two

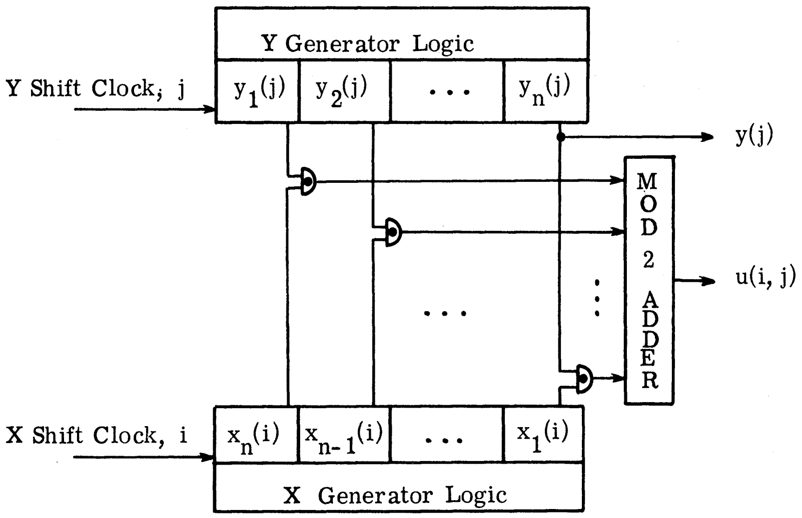


Fig. 3. Dynamic combiner block diagram

generators in Fig. 3 need not have a common clocking rate. The usual mode of operation is for the basic generator, the Y generator, to be clocked at a steady rate and the control generator, the X generator, to be clocked at a slower or irregular rate. In Fig. 3 the output of the combiner has been designated as u(i, j), a general designation which (as the theorems to be stated will prove) can be written as a shifted version of the basic output.

The vector equations or relationships for the dynamic combiner in Fig. 3 will now be derived. Again, as for the static combiner, the content of a register will be denoted by a column vector: Y(j) for the basic generator Y and X(j) for the control generator X. That is,

$$Y(j) = \begin{bmatrix} y_1(j) \\ y_2(j) \\ \vdots \\ y_n(j) \end{bmatrix}$$

where $y_r(j)$ = contents of the r stage of the basic generator at clock pulse j and

$$X(i) = \begin{bmatrix} x_1(i) \\ x_2(i) \\ \vdots \\ x_n(i) \end{bmatrix} .$$

where x(i) = contents of the r stage of the control generator (counted from right to left) at clock pulse i (Fig 3).

The generator logic or "feedback connections" of the generator can be designated by a square n x n "shift" matrix

$$A = [a_{i,j}] \tag{5}$$

where $a_{i,j}$ = 1 if stage j of the generator feeds into stage i
= 0 otherwise

At each occurrence of a shift clock pulse, the shift matrix if multiplied on the right by the current content column vector generates the new content column vector. For example,

$$Y(j+1) = A_Y \, Y(j) \quad \text{(mod-2)} \tag{6}$$

where A_Y = the n x n shift matrix for the Y generator

$Y(j)$ = contents of generator at time j

$Y(j+1)$ = contents of generator one clock pulse later

By repeated application of Eq. 6

$$Y(j) = A_Y^{\,j} \, Y(0) \quad \text{(mod-2)} \tag{7}$$

where $Y(0)$ = initial contents of the basic generator

and similarly

$$X(i) = A_X^{\,i} \, X(0) \qquad \text{(mod-2)} \tag{8}$$

where $X(0)$ = initial contents of the control generator.

In order to express the combiner output in matrix notation we require the use of a simple n x n matrix, R, consisting of ones on the "cross diagonal" running from lower left to upper right:

$$R = \begin{bmatrix} 0 & 0 & . & . & . & 0 & 1 \\ 0 & 0 & . & . & . & 1 & 0 \\ \vdots & \vdots & & & & \vdots & \vdots \\ 0 & 1 & . & . & . & 0 & 0 \\ 1 & 0 & . & . & . & 0 & 0 \end{bmatrix} \tag{9}$$

The matrix R is used to invert the order of entries in a vector or matrix: left multiplication by R inverts the order of rows of a column vector or matrix, and right multiplication by R inverts the columns of a row vector or matrix. For example,

$$R \begin{bmatrix} x_1(i) \\ x_2(i) \\ \vdots \\ x_n(i) \end{bmatrix} = \begin{bmatrix} x_n(i) \\ \vdots \\ x_2(i) \\ x_1(i) \end{bmatrix}$$

By inspection of Fig. 3 the combiner output u(i, j) can be written as

$$u(i, j) = \sum_{r=1}^{n} x_{n+1-r}(i)\, y_r(j) = \sum_{r=1}^{n} x_r(i)\, y_{n+1-r}(j) \quad (\text{mod-2}) \qquad (10a)$$

or in matrix notation using Y(j), X(i), and R defined above

$$u(i, j) = [RX(i)]^T Y(j) = [X(i)]^T R\, Y(j) \quad (\text{mod-2}) \qquad (10b)$$

In the following section it will be shown that if the basic generator is an SSRG, then the control generator should be the matching or equivalent MSRG; the control generator should be loaded with the initial load of a one in only the first stage and then clocked exactly k times in order to have a combiner output which is k digits in advance of the basic output. Similarly, if the basic generator is an MSRG, then the control generator should be an equivalent SSRG similarly operated. Further, if the basic generator is an SCRG, then the control generator should be an equivalent MCRG; and if the basic generator is an MCRG, then the control generator should be an equivalent SCRG. If the basic generator is a Jacobi-hybrid generator, JHG, then the control generator should be an identical JHG, loaded with a one in the final stage, and then clocked k pulses.

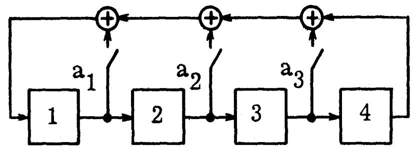
3. FORMAL THEORY

It is shown in Ref. 1 and Ref. 3 that a transient-free linear binary sequence can be generated by any one of the above mentioned five canonical generators. The relations between the shift matrices for the SSRG, MSRG, SCRG, and MCRG, are discussed in detail in Ref. 1. In Fig. 4 examples of shift matrices, A, (using Eq. 5) for 4-stage transient-free generators are given for each of the five types of canonical generators. Appropriate subscripts are appended to each shift matrix A. Also shown in Fig. 4 are block diagrams and abbreviated representations of the five canonical forms of generators corresponding to the shift matrices. In the SSRG and MSRG (see Fig. 4b), the stages of the generators are shift stages. The shaded regions in the abbreviated diagram for the MSRC (and MCRG) represent mod-2 adders placed between the corresponding stages whose two input are the contents of the last stage (if $a_r = 1$ or $b_r = 1$) and the contents of the preceding stage. This sum is fed into the following stage. In the SCRG and MCRG, the stages of the generator are complement stages (a <u>one</u> input changes the contents of the stage) which are denoted by subscript c in the stage number designation. The JHG in Fig. 4(e) is composed of shift stages and complement stages. If $c_r = 1$, then stage r is a complement stage; if $c_r = 0$, the stage r is a shift stage. The adder arrangement is fixed. Note in Fig 4(b) and (d) the feedback switches have been numbered in reverse order. This has been done for notational convenience in the corresponding shift matrices. In fact, this points out the relationship between equivalent SSRG and MSRG and between equivalent SCRG and MCRG, namely, the adder tap arrangements are simply reversed (see Ref. 1).

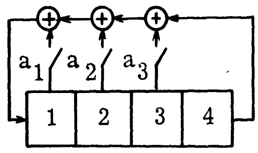
From the shift matrix standpoint the relation between a <u>simple</u> generator and its equivalent <u>modular</u> generator is that the shift matrix of one is the reflection of the shift matrix of the other about the cross diagonal. This reflection is accomplished in equation form by rotating one of the matrices by 180° around its center and then transposing. To accomplish this, the shift matrix is multiplied on both the right and the left by the R matrix and then transposed. This is the key relationship behind the solution of the combiner problem

$$A_{SS} = \begin{bmatrix} a_1 & a_2 & a_3 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

(a) SSRG



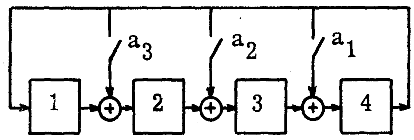
Block Diagram



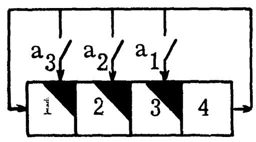
Abbreviated Representation

$$A_{MS} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & a_3 \\ 0 & 1 & 0 & a_2 \\ 0 & 0 & 1 & a_1 \end{bmatrix}$$

(b) MSRG



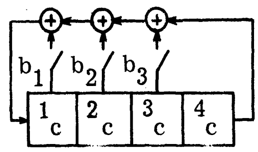
Block Diagram



Abbreviated Representation

$$A_{SC} = \begin{bmatrix} 1+b_1 & b_2 & b_3 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

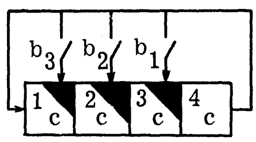
(c) SCRG



Abbreviated Representation

$$A_{MC} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & b_3 \\ 0 & 1 & 1 & b_2 \\ 0 & 0 & 1 & 1+b_1 \end{bmatrix}$$

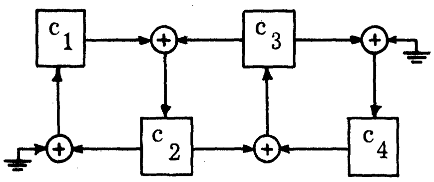
(d) MCRG



Abbreviated Representation

$$A_J = \begin{bmatrix} c_1 & 1 & 0 & 0 \\ 1 & c_2 & 1 & 0 \\ 0 & 1 & c_3 & 1 \\ 0 & 0 & 1 & c_4 \end{bmatrix}$$

(e) JHG



Block Diagram and Abbreviated Representation

Fig. 4. Examples of canonical shift matrices and generators

and is written as

$$A_{MS}^T = R\, A_{SS} R \tag{11a}$$

$$A_{MC}^T = R\, A_{SC} R \tag{11b}$$

For the JHG

$$A_J^T = A_J \tag{11c}$$

A property of the R matrix which will be used later is that R is "involuntary;" that is, when R is multiplied by itself one obtains the identity matrix:

$$R^2 = 1 \tag{12}$$

Another useful relationship among binary generators of these canonical forms will be stated here without proof. One may prove it for himself by a detailed examination of the operation of each of these generators. Consider each of the five types of canonical generator which obey the same characteristic sequence law. Consider the initial condition of each of these generators as being the elementary load, E_1, that is, a <u>one</u> in the first stage and <u>zeros</u> in all other stages. We take as the output of each of these generators the binary stream contents of the last stage. If each of these generators is clocked simultaneously, then the binary stream of last stage contents of <u>all</u> of these generators will be identical (in phase).

<u>Theorem 1</u>

Let the column vector M(j) be the contents of an MSRG after j shifts, and let S(i) be the contents of the equivalent SSRG (has the same characteristic sequence law) after i shifts. Or, let M(j) be the contents of an MCRG after j shifts, and let S(i) be the contents of the equivalent SCRG after i shifts. All registers are assumed to have the same initial condition of the elementary load, E_1. Let A_M and A_S be the corresponding shift matrices. Let

$$u(i, j) = \sum_{r=1}^{n} m_r(i)\, s_{n+1-r}(j) \qquad \text{(mod-2)} \tag{13}$$

Then

$$u(i,\ j) = s_n(i + j) = m_n(i + j) \tag{14}$$

the common value of the last stage contents of any of the four canonical generators after $i + j$ shifts.

Proof

The vector form of Eq. 13 (see Eq. 10) is

$$u(i,\ j) = M^T(i)\ [RS(j)] \qquad \text{(mod-2)} \tag{15}$$

The contents of the M register, i shifts after the initial contents E_1 is, from Eq. 7

$$M^T(i) = [A^i_M E_1]^T = E_1^T\ [A_M^T]^i \qquad \text{(mod-2)} \tag{16}$$

The basic relation between modular and simple generators (independent of shift or complement) is Eq. 11 repeated below.

$$A_M^T = R\ A_S\ R \tag{11}$$

Because $R^2 = I$

$$[RA_S R]^i = RA_S R\ R\ A_S R \cdot \cdot \cdot\ RA_S R = RA_S^i\ R \tag{17}$$

Equation 15 becomes

$$u(i,\ j) = E_1^T\ R\ A_S^i\ R\ RS(j) \qquad \text{(mod-2)} \tag{18}$$

$$u(i,\ j) = E_1^T\ R\ A_S^i\ S(j) \qquad \text{(mod-2)} \tag{19}$$

$$u(i,\ j) = E_1^T\ R\ S(i + j) \qquad \text{(mod-2)} \tag{20}$$

Because $E_1^T\ R = (0,\ 0, \ldots, 0, 1)$ left multiplication by it simply selects the last entry in $S(i + j)$, that is

$$u(i,\ j) = s_n(i + j) \tag{21}$$

By re-indexing, Eq. 13 can be written

$$u(i,\ j) = \sum_{r=1}^{n} s_r(j)\ m_{n+1-r}(i) \qquad \text{(mod-2)}$$

then rewriting Eq. 11 as

$$A_S{}^T = R\,A_M R$$

and following the same procedure above one obtains

$$u(i,\ j) = m_n(i + j)$$

End of Proof

Corollary 1

Let the initial conditions of the generators in Theorem 1 be arbitrary. Then under the conditions stated in Theorem 1

$$u(i,\ j) = [M^T(0)\,R]\ S(i + j) \quad \text{(mod-2)} \qquad (22a)$$

or

$$u(i,\ j) = [S^T(0)\ R]\ M(i + j) \quad \text{(mod-2)} \qquad (22b)$$

where

M(0) = initial content column vector of the modular generator

S(0) = initial content column vector of the simple generator

Proof

From Eq. 15

$$u(i,\ j) = M^T(i)\,[R\,S(j)] = [A_M{}^i(0)]^T\,R[A_S^j\ S(0)] \qquad \text{(mod-2)}$$

$$= M(0)^T\,(A_M{}^T)^i\,R\,A_S{}^j\,S(0) \qquad \text{(mod-2)}$$

Substituting Eqs. 11 and 17

$$u(i,\ j) = M(0)^T\,R\,A_S{}^i\,R\,R A_S{}^j\,S(0)$$

$$u(i,\ j) = [M(0)^T\,R]\ S(i + j) \qquad \text{(mod-2)}$$

Equation 22(b) is proved in a similar manner.

End of Proof

Corollary 2

Let the initial conditions of the generators in Theorem 1 be arbitrary. For maximal generators

$$u(i,\ j) = s_n(i + j + k) \tag{23a}$$

or

$$u(i,\ j) = m_n(i + j + k') \tag{23b}$$

Proof

From Corollary 1, Eq. 22(a)

$$u(i,\ j) = [M^T(0)\ R]\ \ S(i + j) \qquad (\text{mod-}2) \tag{22a}$$

For a maximal generator the initial content vector $M(0)$ can be expressed in terms of the elementary load E_1. That is, there exists a k such that

$$M(0) = A_M^k\ E_1 \qquad (\text{mod-}2) \tag{24}$$

Thus

$$\begin{aligned} u(i,\ j) &= E_1^T (A_M^T)^k\ R\ S(i + j) \\ &= E_1^T\ R\ A_S^k\ R\ RS(i + j) \\ &= E_1^T\ R\ S(i + j + k) \\ &= s_n(i + j + k) \end{aligned}$$

Equation 23(b) is proved in a similar manner by beginning with Eq. 22(b) and by finding a k' such that

$$S(0) = A_S^{k'}\ E_1 \qquad (\text{mod-}2) \tag{25}$$

End of Proof

An application of Theorem 1 to the dynamic combiner of Fig. 3 is illustrated in Fig. 5. In Fig. 5 an SSRG and its equivalent MSRG are employed in the dynamic combiner. According to Theorem 1 the adder output u(i, j) can be expressed in terms of either the SSRG output $s_n(j)$ or the MSRG output $m_n(j)$. As mentioned in Section 2 if both the SSRG and MSRG are initially loaded with the elementary load E_1, the output sequence from the last stages of the two equivalent generators are identical. If one desires to obtain a sequence that is advanced k bits ahead of the SSRG output sequence $s_n(j)$, then the following procedure is followed. Both generators are loaded with the elementary E_1. The shift input (clock line) of the MSRG only is pulsed k times. Then this pulse line is opened and the SSRG shift input is pulsed. The adder output sequence is then advanced k bits ahead of the SSRG output sequence. At any time the adder output sequence can be advanced an additional k' bits ahead of the SSRG output sequence by holding the contents of the SSRG and advancing the MSRG k' states. The contents of the MSRG is maintained and the SSRG again shifted.

Figure 5 has been drawn to emphasize the "symmetry" or mirror imagery between the stages and adders of the two generators employed. Input streams of one correspond to output streams of the other, etc.

Of course, according to Theorem 1, the function of the MSRG as the control generator in Fig. 5 and the SSRG as the basic generator can be interchanged with the same result. Also the SSRG and MSRG can be replaced with an SCRG and MCRG respectively.

In general, the "shift-and-add" property of maximal sequence does not apply to nonmaximal sequence, that is, if y(j) is a maximal sequence, then

$$y(j) \oplus y(j + r) = y(j + s) \qquad \text{for all } r \neq 0 \qquad \text{mod-}L \qquad (26)$$

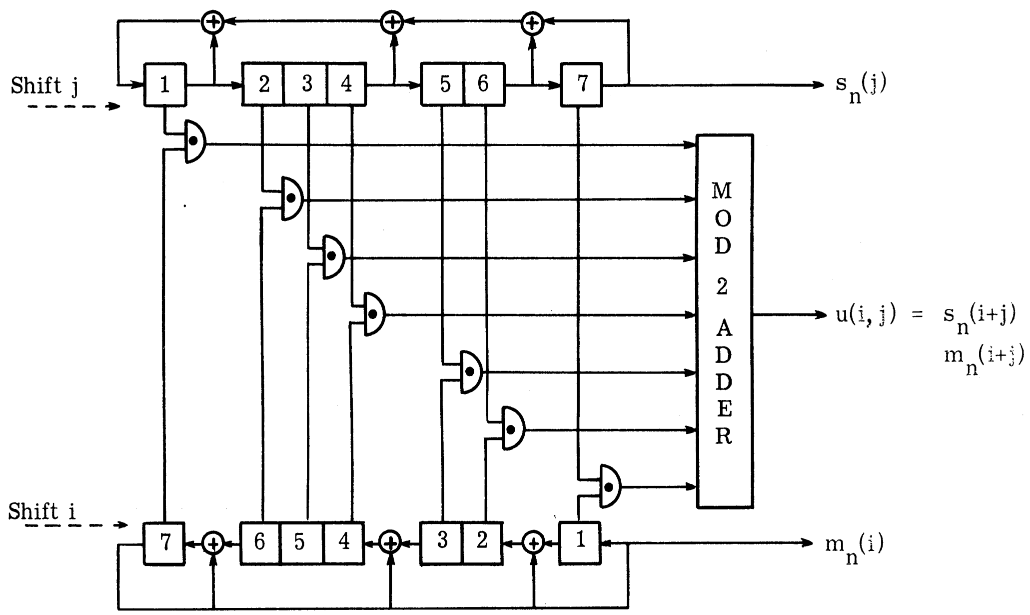


Fig. 5. Example of dynamic combiner block diagram

However, if $y(j)$ is a nonmaximal sequence then Eq. 26 does not in general apply for all integer shifts $r \neq 0 \pmod{\ell}$ but does apply for a set of specified shifts. Since Theorem 1 is not restricted in any way to only maximal generators, the dynamic combiner in Fig. 5 can be used to generate nonmaximal sequences advanced in time relative to the output nonmaximal sequence from one or the other of the generators employed. Note that Theorem 1 requires loading the generators with the elementary load E_1 and the advance at the adder output is relative to the sequence from the last stage of the basic generator. We now consider the significance of Corollary 1. When the initial content vectors $M(0)$ and $S(0)$ are not equal to the elementary load E_1 the adder output $u(i, j)$ is expressed by Eq. 22. From Eq. 22(a)

$$u(i, j) = \sum_{r=1}^{n} m_{m+1-r}(0)\, s_r(i + j) \qquad \text{(mod-2)}$$

or

$$u(i, j) = m_n(0)\, s_1(i + j) \oplus \cdots \oplus m_1(0)\, s_n(i + j) \qquad (27)$$

From Eq. 27 for no shifts of the "modular" control generator $(i = 0)$

$$u(0, j) = m_n(0)\, s_1(j) \oplus \cdots \oplus m_1(0)\, s_n(j) \qquad (28)$$

Assume that the generators in the dynamic combiner are maximal generators, then by the shift and add property of maximal sequences Eq. 28 is the same maximal sequence. Denote this sequence $s_u(j)$, then

$$u(0, j) = s_u(j)$$

and

$$u(i, j) = s_u(i + j)$$

Thus one is able to advance the phase of the adder output sequence by advancing the control generator. In the case of maximal generators the sequence $s_u(j)$ can be related to the sequence from the last stage of the "simple" basic generator. That is, by Corollary 2,

$$u(i, j) = s_u(i + j) = s_n(i + j + k) \qquad (29)$$

where k is specified by Eq. 24.

In the same manner, similar results can be obtained for a "modular" basic generator and a "simple" control generator.

In the case of nonmaximal generators the adder output cannot always be related to the output sequence of the basic generator; in fact the adder output sequence may be an entirely different sequence from that produced at the last stage of the basic generator. For the case of the "simple" basic generator and "modular" control generator, if a k exists such that

$$M(0) = A_M^k \; E_1 \qquad \text{(mod-2)} \tag{30}$$

then the adder output sequence u(i, j) can be expressed as in Eq. 29. If, however, no k exists which satisfies Eq. 30, then the adder output sequence u(i, j) is a different sequence than produced at the basic generator last stage. Denote this sequence d_u, then from Eq. 28

$$u(0, j) = m_n(0)\, s_1(j) \oplus \ldots \oplus m_1(0)\, s_n(j)$$

$$= d_u(j)$$

and

$$u(i, j) = d_u(i + j)$$

Thus again for nonmaximal generators one is able to advance the phase of the adder output sequence by advancing the control generator. However, the adder sequence may or may not be a shifted version of the basic generator's output sequence.

The remainder of this section outlines the application of Theorem 1 to a similar operation for Jacobi-hybrid generators. It hinges on the mathematical fact that two Jacobi-hybrid generators which have the same characteristic sequence obey Eq. 11. Physically these two equivalent generators differ only in the numbering of the stages, right to left versus left to right. Thus they are not really different in the equipment sense as compared to modular and simple generators which may be distinctly different in equipment realization.

The JHG shift matrix consists of entries on the main diagonal, all ones on the first sub and super main diagonals, and zeros elsewhere [see Fig. 4(e)]. If A_H denotes such a matrix, then it is obviously self transpose or symmetric, that is

$$A_H^T = A_H \tag{31}$$

Each such A_H is equivalent to (has the same characteristic equation as) the shift matrix of another JHG which is the 180^o rotation of A_H. That is, if A_H is the shift matrix of a JHG and

$$A_G = R\, A_H\, R \tag{32}$$

then A_G is the shift matrix of a JHG. Consider these two generators, each with initial contents E_1. Let

$$u(i,\ j) = \sum_{r=1}^{n} g_r(i) \cdot h_{n+1-r}(j) \qquad \text{(mod-2)} \tag{33}$$

Since

$$A_G{}^T = R\ A_H\ R = A_G \tag{34}$$

it follows from the proof of Theorem 1 that

$$u(i,\ j) = h_n(i + j) = g_n(i + j) \tag{35}$$

There is no "direction of running" in a JHG as there is in the other canonical forms of generator. The two shift matrices, A_H and A_G, correspond to writing the same law from right to left and left to right. The only essential feature is the correspondence between the direction in which the stages are numbered for the law, and in which they are numbered for the contents vector. This may be most clearly seen in the following matrices and diagrams illustrating Theorem 1 for a JHG. Let

$$A_G = \begin{bmatrix} a_1 & 1 & 0 & 0 \\ 1 & a_2 & 1 & 0 \\ 0 & 1 & a_3 & 1 \\ 0 & 0 & 1 & a_4 \end{bmatrix}$$

Then according to Eq. 34

$$A_H = R\ A_G R = \begin{bmatrix} a_4 & 1 & 0 & 0 \\ 1 & a_3 & 1 & 0 \\ 0 & 1 & a_2 & 1 \\ 0 & 1 & 1 & a_1 \end{bmatrix}$$

It is readily apparent that this combination can be considered as two G registers multiplied stage by stage and summed, where the two initial contents are E_1, a single <u>one</u> in the first stage, and E_n, a single <u>one</u> in the last stage (see Fig. 6).

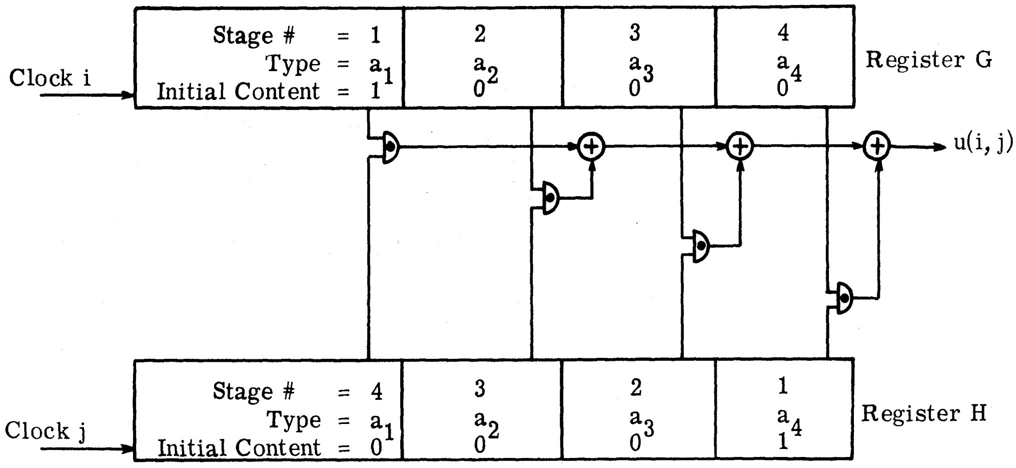


Fig. 6. Block diagram for $u(i,j) = \sum_{r=1}^{4} g_r(i)\, h_{5-r}(j)$; $G(0) = H(0) = E_1$

Formally, let the two column vectors denoting the contents be

$$G(i) = A_G^{\,i}\, E_1 \qquad (\text{mod-}2)$$

$$G'(j) = A_G^{\,j}\, E_n \qquad (\text{mod-}2) \tag{36}$$

Then

$$u(i,\ j) = \sum_{r=1}^{n} g_r(i)\, g'_r(j) \qquad (\text{mod-}2) \tag{37}$$

and therefore

$$u(i,\ j) = g_n(i + j) = g'_1(1 + j) \tag{38}$$

4. GIANT STEPPING

In Section 3 it was shown that the phase of a sequence can be advanced by advancing the contents of the control generator. In order to advance k bits the control generator must be stepped k bits. In this section two theorems are stated which permit advancing the phase some integer number of bits m for each shift of the control generator. That is the adder output sequence

$$u(i, j) = s_n(j + m\, i) \tag{39}$$

where m is a function of the type of control generator and its sequence law. The term "giant stepping" has been applied to such a process because the phase can be advanced by large amounts (for large m). However, since the step size, m , in Eq. 39 is related to the generators employed, it cannot be arbitrarily specified and so Eq. 39 is a form of "inflexible" giant stepping.

Theorem 2a

Let the column vector M(i) be the contents of a maximal control MSRG after i shifts, and let S(j) be the contents at time j of a maximal basic SCRG whose feedback adder arrangement is identical to the SSRG equivalent to the control MSRG. Both registers are assumed to have the same initial contents of the elementary load, E_1. Let A_{MS} and A_{SC} be the corresponding shift matrices. Let

$$u(i,\ j)\ =\ \sum_{r=1}^{n} m_r(i)\, s_{n+1-r}(j) \quad \text{(mod-2)} \tag{40}$$

Then

$$u(i,\ j) = s_n(j + mi) \tag{41}$$

the output sequence from the last stage contents of the basic generator.

Proof

As in Theorem 1, Eq. 40 can be written

$$u(i, j) = M(i)^T R S(j) = E_1^T A_{MS}^{T\ i} R A_{SC}^j E_1$$

By hypothesis the basic SCRG has been selected such that

$$A_{SC} + I = A_{SS} \quad (\text{mod-2}) \tag{42}$$

where

A_{SC} = the shift matrix of the basic SCRG

I = identity matrix

A_{SS} = the shift matrix of the SSRG equivalent to the control MSRG

It is shown in Ref. 1 or 3 that for maximal generators there exists an integer m such that

$$A_{SC}^m = A_{SC} + I \qquad (\text{mod-2}) \tag{43}$$

From Eqs. 11a, 42 and 43

$$A^T_{MS} = R A_{SS} R = R A_{SC}^m R \tag{44}$$

Thus

$$\begin{aligned} u(i, j) &= E_1^T R A_{SC}^{mi} R R A_{SC}^j E_1 && (\text{mod-2}) \\ &= [E_1^T R]\ S(j + mi) && (\text{mod-2}) \\ &= s_n(j + mi) \end{aligned}$$

<u>End of Proof</u>

<u>Theorem 2b</u>

Interchanging the role of the MSRG and SCRG in Theorem 2a, that is, letting the MSRG be the basic generator and letting the SCRG be the control generator, one obtains

$$u(i, j) = \sum_{r=1}^{n} s_r(j)\ m_{n+1-r}(i) = m_n(m'j + i) \tag{45}$$

<u>Proof</u>

From Eq. 45

$$u(i, j) = S(j)^T R M(i) = E_1^T A_{SC}^{T\ j} R A_{MS}^i E_1 \tag{46}$$

From Eqs. 42, 43

$$A_{SC}^m = A_{SC} + I = A_{SS} \tag{47}$$

Let m' equal the smallest integer such that the product

$$mm' = 1 \pmod{L} \tag{48}$$

where

$L = 2^n - 1$ = length of the maximal sequences

Then from Eqs. 47 and 48 and the fact that A_{SC} and A_{SS} correspond to maximal generators with period L

$$(A_{SC}{}^{m})^{m'} = A_{SC} = A_{SS}{}^{m'} \pmod{2} \tag{49}$$

But

$$A_{SS} = R\, A_{MS}^{T}\, R$$

so

$$A_{SS}{}^{m'} = (R\, A_{MS}^{m'}\, R)^{T}$$

or

$$A_{SC}^{T} = R\, A_{MS}^{m'}\, R \tag{50}$$

It follows from Eq. 50 that Eq. 46 can be reduced to

$$u(i,\ j) = E_1{}^{T} R\, A_{MS}{}^{m'j+i}\, E_1$$

$$= m_n(m'j + i)$$

<u>End of Proof</u>

We single out the significance of Eq. 48 in the proof of Theorem 2b by the following:

<u>Corollary 3</u>

The step sizes m and m' in Theorem 2 are related by

$$mm' = 1 \pmod{L} \tag{48}$$

Corollary 4

For the conditions of Theorem 2 let the initial loads be arbitrary. Designate them M(0) and S(0), then

$$u(i, j) = [M^T(0)R]\, S(j + mi) \quad \text{(mod-2)} \tag{51}$$

$$= s_n(j + mk + mi)$$

where

$$M(0) = A_{MS}^{k} E_1$$

or

$$u(i, j) = [S^T(0)R]\, M(m'j + i) \quad \text{(mod-2)}$$

$$= m_n(m'j + m'k' + i) \tag{52}$$

where

$$S(0) = A_{SC}^{k'} E_1$$

The proof of Corollary 4 follows from the steps used in the proof of Corollary and Theorem 2.

Theorem 3a

Let the column vector S(i) be the contents of a maximal control SSRG after i shifts, and let M(j) be the contents at time j of a maximal basic MCRG whose feedback adder arrangement is identical to the MSRG equivalent to the control SSRG. Both registers are assumed to have the same initial contents of the elementary load, E_1. Let A_{MC} and A_{SS} be the corresponding shift matrices. Let

$$u(i, j) = \sum_{r=1}^{n} s_r(i)\, m_{n+1-r}(j) \quad \text{(mod-2)} \tag{53}$$

Then

$$u(i, j) = m_n(j + pi) \tag{54}$$

the output sequence from the last stage contents of the basic generator.

Proof

Equation 53 can be written

$$u(i,\ j) = S(i)^T R\, M(j) = E_1^T A_{SS}^{T\,i} R\, A_{MC}^{\ j} E_1 \tag{55}$$

By hypothesis the basic MCRG has been selected such that

$$A_{MC} + I = A_{MS}$$

where

A_{MC} = the shift matrix of the basic MCRG

A_{MS} = the shift matrix of the MSRG equivalent to the control SSRG

As in Eq. 43 there exists an integer p such that

$$A_{MC}^{\ p} = A_{MC} + I \tag{56}$$

which leads to the relation

$$A_{SS}^T = R\, A_{MS} R = R\, A_{MC}^{\ p} R \tag{57}$$

Equation 54 then follows from Eq. 57 and Eq. 55.

End of Proof

Theorem 3b

Interchanging the role of the SSRG and MCRG in Theorem 3a, that is, letting the SSRG be the basic generator and letting the MCRG be the control generator, one obtains

$$u(i,\ j) = \sum_{r=1}^{n} m_r(j)\, s_{n+1-r}(i) = s_n(p'j + i) \tag{58}$$

Proof

As in the proof of Theorem 2b the key relationship is Eq. 56 for which a smallest integer p' exists such that

$$pp' = 1 \qquad \text{(mod-L)} \qquad (59)$$

Using Eq. 56 this leads to the relationships

$$(A_{MC}{}^{p})^{p'} = A_{MC} = A_{MS}{}^{p'} \qquad \text{(mod-2)} \qquad (60)$$

and

$$A_{MC}^{T} = R\, A_{SS}{}^{p'} R \qquad \text{(mod-2)} \qquad (61)$$

Equation 58 then follows directly from application of Eq. 61.

<u>Corollary 5</u>

The step sizes p and p' in Theorem 3 are related by

$$pp' = 1 \qquad \text{(mod-L)} \qquad (59)$$

<u>Corollary 6</u>

For the conditions of Theorem 3 let the initial loads be arbitrary. Designate them M(0) and S(0), then

$$u(i,\ j) = [S^{T}(0)\, R]\ M(j + pi) \qquad \text{(mod-2)} \qquad (62)$$

$$= m_n(j + pk + pi)$$

where

$$S(0) = A_{SS}^{k}\ E_1$$

or

$$u(i,\ j) = [M^{T}(0) R]\ S(p'j + i) \qquad \text{(mod-2)} \qquad (63)$$

$$= s_n(1'j + p'k' + i)$$

where

$$M(0) = A_{MC}^{k'}\ E_1$$

Corollaries 5 and 6 are similar to Corollaries 3 and 4 and they can be proved in a similar manner.

Note that in both Theorem 2 and Theorem 3 the generators are maximal generators. This assumption is, in general, required to make the statements corresponding to Eqs. 43 and 56. Also in the case of Theorem 2 the SCRG is selected such that its adder tap configuration is identical to the SSRG which would be used with the MSRG to satisfy Theorem 1. These conditions cannot always be simultaneously satisfied for every choice of MSRG. Only if the period of the maximal sequence $L = 2^n - 1$ is a prime number, can one find an SCRG satisfying both requirements of Theorem 2 for every maximal MSRG of length n. Similar comments apply to the MCRG in Theorem 3 for each maximal SSRG selected.

Theorems 2 and 3 prove the existence of an inflexible giant stepping capability; however, the numerical value of step sizes m and m' or p and p' is not readily obtained. Two basic methods are available for obtaining the step sizes: (1) from the B_A matrix, or (2) from the interstage shift of an SCRG.

Method 1. B_A Matrix

Consider m and m' in Theorem 2. According to Eq. 43 m is the row of the B_A matrix for the characteristic sequence law of the shift matrix A_{SC} which contains ones in only the n-th and (n - 1)-th columns and zeros elsewhere (see Ref. 1 or 3). Further the MSRG generator corresponding to the law of the SCRG can be used as a B_A matrix computer to generate the rows of the required B_A matrix (see Ref. 1).

After obtaining m in the above manner, Corollary 3 can be used to determine m'. Alternatively the B_A matrix corresponding to the sequence law of the MSRG can be used. It can be shown that m' is the solution of

$$A_{MS}^{m'} = A_{MS} + I$$

where

$$A_{MS} = \text{shift matrix of the MSRG}$$

Thus the same procedure can be used to find m' as used above for finding m. Of course, in general, the B_A matrices for the two cases are different. The term, m, is found from the B_A matrix corresponding to the law of the MSRG.

The same procedure is followed for finding p and p' in Theorem 3 with the appropriate interchange of generators and laws.

Method 2. Interstage Shift of SCRG

It is shown in Ref. 1 that the shift k between adjacent stages of a maximal SCRG is the solution of

$$A_{SC}^{k} = A_{SC} + I \qquad \text{(mod-2)}$$

This is identical to the form of the equations for determining m and m'. This interstage shift has been tabulated in Ref. 1 for all maximal SCRG of length $2 \leq n \leq 12$.

Alternatively the appropriate SCRG's can be connected and the interstage shift measured. Or, given one of the steps, the other can be calculated as described in the corollary.

An example of the inflexible giant stepping procedure is shown in Fig. 7. In Fig. 7 we have replaced the SSRG of Fig. 5 with an SCRG having the same feedback arrangement as the SSRG. The MSRG is unchanged. Using Method 2 above for determining the step

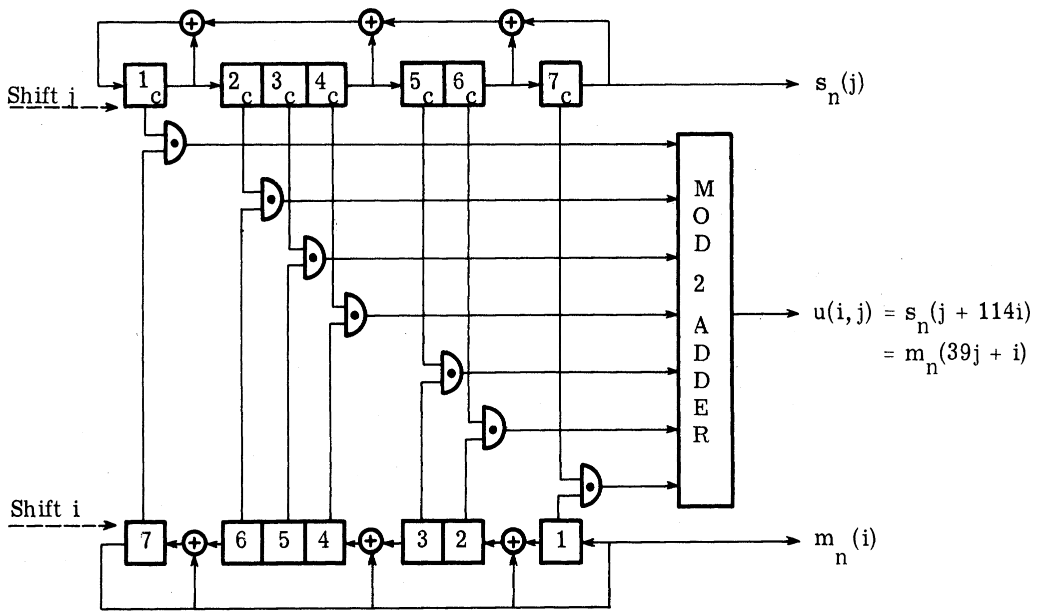


Fig. 7. Example of dynamic combiner block diagram for giant stepping

sizes m and m', one finds that m = 114, m' = 39 (see Ref. 1, Table 1). As a check

$$mm' = 4446 = 1 \quad (\text{mod-}127)$$

Note that in Fig. 7, as in Fig. 5, the "mirror imagery" exists between the generators. However, for the generators in Fig. 7 the characteristic sequence laws of the generators are in general different, whereas, in Fig. 5 the characteristic sequence laws are identical.

5. SUMMARY

We have shown, through Theorem 1, that a "modular" type generator and an equivalent "simple" type generator can be employed in a configuration so that a sequence can be generated which has an arbitrary phase advance relative to either of the generators output. The desired advance is obtained simply by pulsing either one of the generators. Further, it has been shown that two identical JHG's can be employed to perform the same function.

These techniques are not restricted to maximal generators, provided the proper initial loading of generators is used. For some initial loads for nonmaximal generators it was shown that the sequence from the adder may be a different sequence from that produced at the generator's output.

A form of "inflexible giant stepping" for maximal generators was discussed. It was shown through Theorems 2 and 3 that a "compliment register" generator and a "shift register" generator, one being a "simple" type, the other a "modular" type, can be employed to perform the giant stepping. Two methods for obtaining the step sizes were discussed. Some restrictions are imposed by Theorems 2 and 3, so that for all shift registers the required complement register generator may not be suitable. However, this is not a serious limitation in that one can begin with a different shift register generator.

REFERENCES

1. C. C. Hoopes and R. N. Randall, Study of Linear Sequence Generators, Cooley Electronics Laboratory Technical Report No. 165, The University of Michigan, Ann Arbor, Michigan, June 1966.

2. H. J. Schmidt, Methods for Obtaining any Phase Shift of Maximal Linear Sequences, AFAL-TR-66-313, Air Force Avionics Laboratory, Research and Technology Division, Wright-Patterson AFB, Ohio, November 1966.

3. T. G. Birdsall and M. P. Ristenbatt, Introduction to Linear Shift Register Generated Sequences, Cooley Electronics Laboratory Technical Report No. 90, The University of Michigan, Ann Arbor, Michigan, 1958.

DISTRIBUTION LIST

No. of Copies	
20	Director, National Security Agency ATTN: TDL Ft. George G. Meade, Md. 20755
1	Office of Assistant Secretary of Defense (Research and Engineering) ATTN: Technical Library, Rm 3E1065 Washington, D. C. 20301
2	Chief of Research and Development Department of the Army Washington, D. C. 20315
2	Commanding General U. S. Army Materiel Command ATTN: R&D Directorate Washington, D. C. 20315
1	Commanding Officer 52d USASASOC Fort Huachuca, Arizona 85613
1	Commanding Officer U. S. Army Sec Agcy Combat Dev Actv Arlington Hall Station Arlington, Virginia 22212
2	U. S. Army Security Agency ATTN: OACofS, Dev (IARD) Arlington Hall Station Arlington, Virginia 22212
1	Chief, Mountain View Office Electronic Warfare Lab., USAECOM P. O. Box 205 Mountain View, California 94040
1	USAECOM Liaison Officer Rome Air Development Center ATTN: EMPL Griffiss Air Force Base New York 13440
5	Commanding General U. S. Army Electronics Command Fort Monmouth, New Jersey 07703 ATTN: AMSEL-EW AMSEL-RD-MAT AMSEL-RD-MAF (record copy) AMSEL-WL-C AMSEL-WL-E
1	Remote Area Conflict Info Center Battelle Memorial Institute 505 King Avenue Columbus, Ohio 43201
5	Director National Security Agency ATTN: D-42 (Mr. M. H. Klein) Fort George G. Meade, Maryland 20755
1	Sylvania Electronic Systems - West P. O. Box 205 ATTN: E. Wong Mountain View, California 94040
1	Dr. T. W. Butler, Director Cooley Electronics Laboratory The University of Michigan Ann Arbor, Michigan 48107
25	Cooley Electronics Laboratory The University of Michigan Ann Arbor, Michigan 48107

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Cooley Electronics Laboratory The University of Michigan Ann Arbor, Michigan	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

LINEAR SEQUENCE GENERATORS: A NOTE ON METHODS FOR OBTAINING ANY PHASE SHIFT OF LINEAR SEQUENCES

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Special Technical Report - June 1967

5. AUTHOR(S) *(Last name, first name, initial)*

Birdsall, T. G.
Hoopes, C. C.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1967	40	3

8a. CONTRACT OR GRANT NO. DAAB 07-67-C0206 b. PROJECT NO. 5A6-79191 D902-40-45 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) C. E. L. Technical Report No. 8627-184 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. AVAILABILITY/LIMITATION NOTICES

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY U. S. Army Electronics Command Fort Monmouth, New Jersey AMSEL-WL-C

13. ABSTRACT Three techniques for advancing the phase of a linear binary sequence are described. The first employs a simple shift register generator and its equivalent modular shift register generator. The second in a similar arrangement employs a simple-complement register generator and its equivalent modular complement register. The third technique described incorporates two identical Jacobi-hybrid generators.

A form of inflexible giant stepping is also discussed. By combining in the manner described a simple complement register generator and a modular shift register generator one can advance the sequence phase in steps of an integer number of bits where the step size is a function of the generators employed. This same function can be accomplished by employing a modular complement register generator and a simple shift register generator.

Examples and applications of these techniques are included.

UNCLASSIFIED
Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Pseudo-Random Sequences Feedback Generators Generators Equivalent Generators Linear Sequences Matrix Algebra Polynomials Mathematic Analysis						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through ________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

UNCLASSIFIED
Security Classification