MINIMAL AND MAXIMAL FIBONACCI REPRESENTATIONS: BOOLEAN GENERATION*

P. MONTEIRO and R. W. NEWCOMB
University of Maryland, College Park, Maryland 20742

"Then hear my message ere thou speed away." [1]

1. INTRODUCTION

Among the many important and interesting properties of Fibonacci numbers are those of yielding unique minimal and maximal representations of arbitrary nonnegative integers [2, p. 74]. Since these representations can be convenient for characterizing and calculating with integers, it is of interest to have an algorithmic process for obtaining the minimal and maximal representations. This we present here in terms of Boolean functions for which logic circuits are developed and from which hardware implementation can occur.

We take the j^{th} Fibonacci number as

(1a) $$F_j = F_{j-1} + F_{j-2}$$

(1b) $$F_0 = 0, \qquad F_1 = 1.$$

For concreteness we use the initial conditions of (1b) though as far as the algorithm to be developed is concerned others, such as $F_0 = 2$, $F_1 = 1$ for Lucas numbers, are equally satisfactory. Then it is known [3] that any nonnegative integer N can be represented as

(2a) $$N = \sum_{j=2}^{n} a_j F_j; \qquad F_n \leqslant N < F_{n+1},$$

where each a_j is a binary number, that is either zero or unity. There are many such representations possible but that called the *minimal representation,* with [4] [5]

(2b) $$a_j a_{j+1} = 0, \qquad j = 2, 3, \cdots, n-1$$

and that called the *maximal representation,* with [6]

(2c) $$a_j + a_{j+1} \geqslant 1, \qquad j = 2, 3, \cdots, n-1$$

are unique. Indeed each of these two representations in itself uniquely characterizes the Fibonacci numbers [7]. In the following we shall represent N of (2a) by the coefficients, writing for convenience

(3) $$N = a_2 a_3 \cdots a_n.$$

Note that the least significant digits are to the left. Thus, for example

$$N = 24 = F_2 + F_3 + F_6 + F_7 = 1100110$$

has for its minimal and maximal forms $N = 0010001$ and $N = 1111010$, respectively.

"Before thee mountains rise and rivers flow." [1]

2. BOOLEAN EXPRESSIONS

We first set down the rules for obtaining the minimal and maximal representations from which the desired Boolean functions can be obtained. The rules follow by iteratively applying (1a), the iterations being indexed by time t in

*This work was supported by the Air Force Office of Scientific Research under Grant AFOSR 70-1910.

discrete increments t_k, $k = 0, 1, \cdots$. Thus, we assume that a configuration for N in the form of (3) is on hand at time t_k,

$$N = a_2(t_k)a_3(t_k)\cdots a_n(t_k).$$

The configuration is "changed" at the next instant of time t_{k+1} according to the following rules.

MINIMAL FORM RULE: If at time t_k any sequence 110 occurs in N replace it by time t_{k+1} by 001; repeat for all t_k until no changes occur.

As an example to illustrate the process consider the following:

$$\begin{aligned} t_0,\ N = 61 &= 111011010 \\ t_1, \quad &= 100100110 \\ t_2, \quad &= 100100001\,. \end{aligned}$$

The maximal form rule is similar but uses a procedure which is the reverse of that for the minimal form rule.

MAXIMAL FORM RULE: If at time t_k any sequence 001 occurs in N replace it at time t_{k+1} by 110; repeat for all t_k until no changes occur.

This is illustrated by the following example:

$$\begin{aligned} t_0,\ N = 13 &= 000001 \\ t_1, \quad &= 000110 \\ t_2, \quad &= 011010\,. \end{aligned}$$

Translation of the rules into Boolean expressions can occur through the use of a truth table [8, p. 50]. Alternately we can read off from the rules the Boolean conditions. For example, from the minimal form rule we have the following: A zero in the j^{th} position becomes changed to a one (that is, complemented) if a_{j-1} and a_{j-2} are both ones. If a_j is a one it remains a one if either $a_{j+1} = a_{j-1} = 0$ or $a_{j+1} = a_{j+2} = 1$; otherwise a_j becomes zero. Using standard Boolean symbols ($\cdot$ = and, + = or, $\bar{\ }$ = complement = not) we then have the following Boolean expression:

Minimal Form (assume $a_0(t) = a_1(t) = a_{n+2}(t) = 0$,

(4a) $$a_j(t_{k+1}) = [\bar{a}_j(t_k)\cdot a_{j-1}(t_k)\cdot a_{j-2}(t_k)] + \left\{ a_j(t_k)\cdot[(\bar{a}_{j-1}(t_k)\cdot\bar{a}_{j+1}(t_k)) + (a_{j+1}(t_k)\cdot a_{j+2}(t_k))]\right\}.$$

Using similar reasoning we have

Maximal Form (assume $a_1(t) \equiv 1$, $a_{n+1}(t) = a_{n+2}(t) \equiv 0$):

(4b) $$a_j(t_{k+1}) = [a_j(t_k)\cdot a_{j-1}(t_k)] + [\bar{a}_j(t_k)\cdot\bar{a}_{j+1}(t_k)\cdot a_{j+2}(t_k)] + [\bar{a}_{j-1}(t_k)\cdot\bar{a}_j(t_k)\cdot a_{j+1}(t_k)]\,.$$

As can be readily verified, beginning at t_0 with a given number N, the maximum time to reach either the minimal or maximal form using (4) is $t_{[n/2]}$, where $[\cdot]$ denotes the integer part; this maximum time is achieved when n is odd and the initial representation has all zeros or ones except for a_n.

Equations (4) can be implemented through logic circuits, these being shown in Figures 1 and 2 for one a_j. In the figures, at a given instant t_k, the binary values of the $a_j(t_k)$ are read into a register whose cells are so labelled. These values serve as inputs to the logic circuits shown. On being processed in the logic circuitry, designed according to (4), the result is fed back to the register cells to be clocked in at the next instant t_{k+1}. The minimum time difference, $t_{k+1} - t_k$, possible is seen to be the delay time for signals to traverse the logic circuits. After a time of at most $t_{[n/2]}$ the reading of the register will have settled to the required form. Of course, to completely implement (4) the end cells of the register, which have the assumed stationary values, remain constant between any two clock pulses.

The output of Fig. 2 can be derived from the circuit of Fig. 1 by complementing the initial input and the final output, and vice-versa.

It should be mentioned that the given rules are not the only ones available. For example, we could have used the alternate minimal form rule:

At time t_k proceed from higher to lower numbered indices replacing at time t_{k+1} the first zero followed by two ones by a one followed by two zeros; repeat for all t_k until no change occurs.

This rule can be expressed in Boolean form by substituting

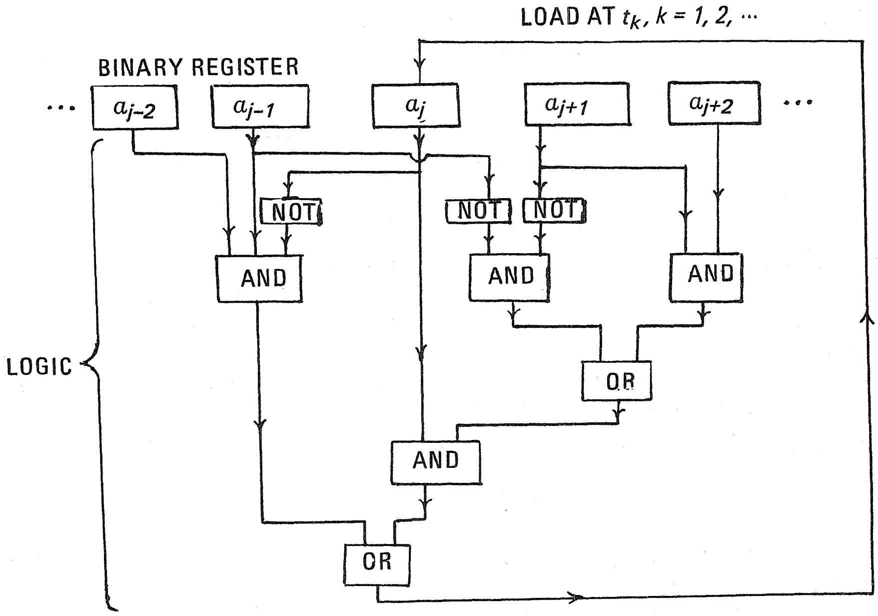

Figure 1

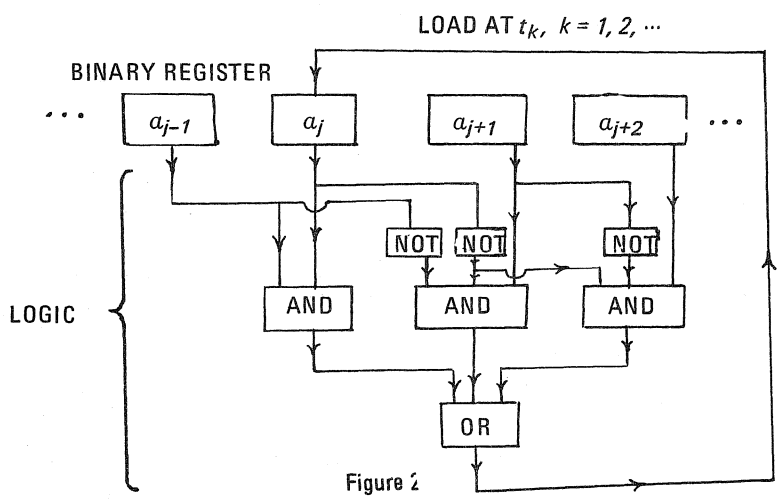

Figure 2

(5a) $$R_j = [a_{j-1} \cdot a_j \cdot \bar{a}_{j+1} \cdot C_{j+2}] + [a_j \cdot a_{j+1} \cdot \bar{a}_{j+2} \cdot C_{j+3}]$$

(5b) $$S_j = a_{j-2} \cdot a_{j-1} \cdot \bar{a}_j \cdot C_{j+1}$$

(5c) $$C_j = (\bar{a}_{j-2} + \bar{a}_{j-1} + a_j) \cdot C_{j+1}$$

(5d) $$C_{n+1} \equiv 1, \quad a_{n+1} = C_{n+2} = R_n \equiv 0$$

into

$$a_j(t_{k+1}) = [S_j(t_k) + \bar{R}_j(t_k)] \cdot a_j(t_k)$$

Equations (5) can be implemented by appropriate circuitry, as for (4), where R and S represent the reset and set inputs of an R-S flip-flop [8, p. 83] and C_j could be interpreted as a timing signal which signifies completion of changes (if any) in stage j. As before, a similar rule for the maximal form can be developed.

"When thou art weary, on the mountains stay,
And when exhausted, drink the rivers' driven spray." [1]

REFERENCES

1. Kalidasa, "The Cloud Messenger," *Shakuntala and Other Writings,* E. P. Dutton and Co., New York, 1959, p. 187.
2. V. E. Hoggatt, Jr., *Fibonacci and Lucas Numbers,* Houghton Mifflin Co., Boston, 1969.
3. J. L. Brown, Jr., "Note on Complete Sequences of Integers," *Amer. Math. Monthly,* Vol. 68, No. 6 (June-July, 1961), pp. 557–560.
4. D. E. Daykin, "Representation of Natural Numbers as Sums of Generalized Fibonacci Numbers," *Journal of the London Math. Society,* Vol. 35 (1960), pp. 143–160.
5. J. L. Brown, Jr., "Zeckendorf's Theorem and Some Applications," *The Fibonacci Quarterly,* Vol. 2, No. 3 (Oct. 1964), pp. 163–168.
6. J. L. Brown, Jr., "A New Characteristic of the Fibonacci Numbers," *The Fibonacci Quarterly,* Vol. 3, No. 1 (Feb. 1965), pp. 1–8.
7. J. Galambos, "A Constructive Uniqueness Theorem on Representing Integers," *The Fibonacci Quarterly,* Vol. 10, No. 6 (Dec. 1972), pp. 569–598.
8. T. Kohonen, *Digital Circuits and Devices,"* Prentice-Hall, Inc., Englewood Cliffs, 1972.

****

LETTER TO THE EDITOR

December 2, 1975

Dear Dr. Hoggatt:

I showed Dr. James W. Follin, Jr., of the Applied Physics Laboratory the example in D. Shanks, "Incredible Identities," *The Fibonacci Quarterly,* Vol. 12, No. 3 (Oct. 1974), pp. 271, 180. I think his generalization would be of interest.

Set $K^2 = m + n$. Then one has the identity

$$\sqrt{m} + \sqrt{2(K+\sqrt{m})} = \sqrt{K+\sqrt{n}} + \sqrt{K+m-\sqrt{n}+2\sqrt{m(K-\sqrt{n})}}\ ,$$

which can be checked by squaring twice, while performing all simplifications, including substitution and observing a perfect square.

William G. Spohn, Jr.