

# Redundant Multiple-Valued Number Systems

<p>Jon T. Butler Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121 U.S.A.</p>	<p>Tsutomu Sasao Department of Computer Science and Electronics Kyushu Institute of Technology Iizuka 820, Japan</p>
---	--

## I. INTRODUCTION

In a civilization, human labor of one type is exchanged for human labor of another type. For example, when one eats at a restaurant, one exchanges money earned from one's own labor for the labor of people who cook, raise animals and vegetables, deliver goods, etc.. This exchange is made possible by a salary, which establishes a value for labor of different types. There are many ways to represent a value; however, in virtually all number systems, symbols from some finite set are combined to represent potentially large values. The decimal number system, in which there are ten symbols, has found wide use; many people believe this is due to the ten fingers of the human species. A similar situation exists in computers. Since conventional logic elements easily realize two values, computers use binary number systems. The prospect of multiple-valued logic suggests the need to understand number systems with more than two digits.

In spite of the recent use of binary systems, there is evidence that suggests the Chinese used the binary number system 5000 years ago, Newman [15, p. 2422].

An interesting discovery was made at the site of Mohenjo-Daro, an ancient city near the Indus River (Morrison and Morrison [14]). Irregularly shaped stones were found, each twice the weight of another. Archeologists believe that these were used by merchants in the Harappan culture for an equal-arm balance 4000 years ago. For some object on one side of a balance in equilibrium, the stones on the other side represent the binary value of the object's weight. Fig. 1 below shows an example of how such a system might work.

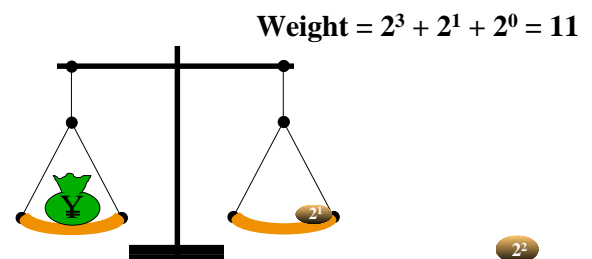


Figure 1. Binary balance scale.

# Report Documentation Page

Form Approved  
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE <b>JUL 1997</b>		2. REPORT TYPE		3. DATES COVERED	
4. TITLE AND SUBTITLE <b>Redundant Multiple-Valued Number Systems</b>				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) <b>Naval Postgraduate School, Department of Electrical and Computer Engineering, Monterey, CA, 93943</b>				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT <b>Approved for public release; distribution unlimited.</b>					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES <b>9</b>	19a. NAME OF RESPONSIBLE PERSON
a. REPORT <b>unclassified</b>	b. ABSTRACT <b>unclassified</b>	c. THIS PAGE <b>unclassified</b>			

The formation of this system is simple. First, two small stones are found, each of identical weight; a balance scale is used to determine that their weights are identical. Interestingly, the two equal-weight stones found near the Indus River, each weighed slightly less than an ounce. Then, using a balance scale again, a third stone is found equal to the weight of the two equal-weight stones. These three stones are then used to find a fourth, which has weight identical to their combined weight, etc.. Thus, the stones' weight occurs in the proportion - 1, 1, 2, 4, 8, 16, ... .

### III. COMPARISON OF NUMBER SYSTEMS

#### A. STANDARD BINARY NUMBER SYSTEM

In the standard binary number system, a number  $N$  has a value given as

$$N = a_{n-1}2^{n-1} + \dots + a_22^2 + a_12^1 + a_02^0,$$

where  $a_i \in \{0,1\}$ . Fig. 2 below shows an example of the 4-bit standard binary number system. The table shows all binary 4-tuples and their corresponding value. The histogram plots the number of representations for each numerical value. From this, it can be seen that there is exactly one 4-bit representative for each of the non-zero integers, 0, 1, 2, ... , 15.

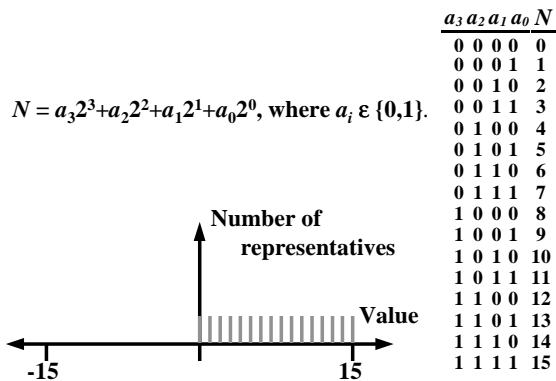


Figure 2. The 4-bit Standard Number System.

#### B. NEGA-BINARY NUMBER SYSTEM

In the nega-binary number system, both positive and negative integers are represented. In this system, the number representation is the same as the binary number system with the base 2 replaced by -2. That is,

$$N = a_{n-1}(-2)^{n-1} + \dots + a_2(-2)^2 + a_1(-2)^1 + a_0(-2)^0.$$

Fig. 3 shows the 4-bit nega-binary number system. As with the standard binary number system, every number has a unique representative 4-tuple. It is interesting that there are twice as many negative integers as there are positive integers. This statement is true of general nega-binary number systems, when the number of bits is even. If the number of bits is odd, the number of positive integers is twice the number of negative integers plus 1. The disparity between the number of negative numbers and positive numbers is a significant disadvantage.

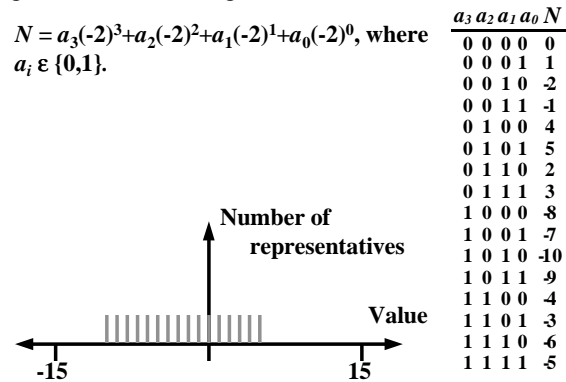


Figure 3. The 4-bit Nega-Binary Number System.

#### B. TWO'S COMPLEMENT NUMBER SYSTEM

Like the nega-binary number system, the two's complement system represents both positive and negative integers. However, in the latter, the number of positive and negative numbers are approximately the same. Here,

$$N = a_{n-1}(-2)^{n-1} + \dots + a_22^2 + a_12^1 + a_02^0.$$

Fig. 4 below shows the 4-bit two's complement

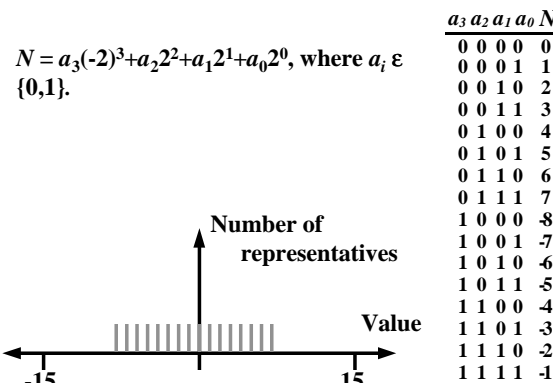


Figure 4. The 4-bit Two's Complement Number System.

number system. As with all number systems shown so far, all representations are unique. In this system, there is one more negative number than positive number.

#### D. ONE'S COMPLEMENT NUMBER SYSTEM

The one's complement number system is the first example of a redundant number system. Specifically, 0 has two representatives, usually called 0 and -0. With any number represented as

$$N = a_{n-1}(-2)^{n-1} + \dots + a_2 2^2 + a_1 2^1 + a_0 2^0 + a_{n-1}$$

0 is 00...0 and -0 is 11...1. Fig. 5 below shows the 4-bit one's complement number system. The two representatives of 0 are shown as a line of length 2 at value 0.

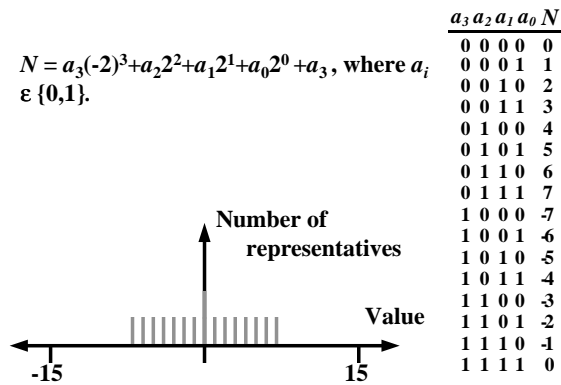


Figure 5. The 4-Bit One's Complement Number System.

#### E. FIBONACCI NUMBER SYSTEM

The Fibonacci number system is the second example of a redundant number system. In such a number system, numbers have the value

$$N = a_{n-1} F_{n+1} + \dots + a_2 F_4 + a_1 F_3 + a_0 F_2,$$

where  $F_i = F_{i-1} + F_{i-2}$  and  $F_2 = F_1 = 1$ . That is, instead of powers of 2 or -2, Fibonacci numbers are used as the base. In such a number system, there are many redundant representatives. Fig. 6 shows that, for the 4-bit Fibonacci number system, there are four redundant representatives.

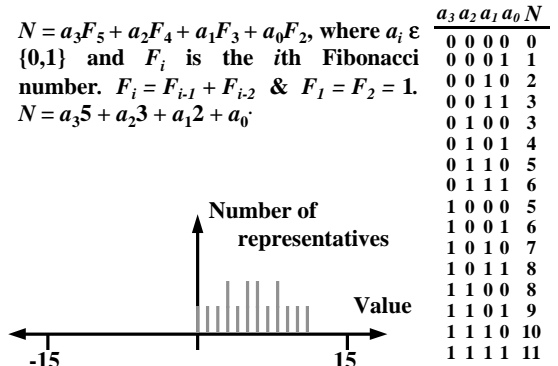


Figure 6. The 4-Bit Fibonacci Number System.

#### F. FIBONACCI NUMBER SYSTEM WITH NO ADJACENT 1'S

Zeckendorf [16] showed that if one retains only those binary patterns in the Fibonacci number system without pairs of 1's, then, each integer is uniquely represented. Fig. 7 shows this. In the table of Fig. 7, 4-tuples with at least one pair of 1's are overlaid with a line. Removing all 4-bit patterns with pairs of 1's, yields a number system in which 0, 1, ..., and 7 are uniquely represented.

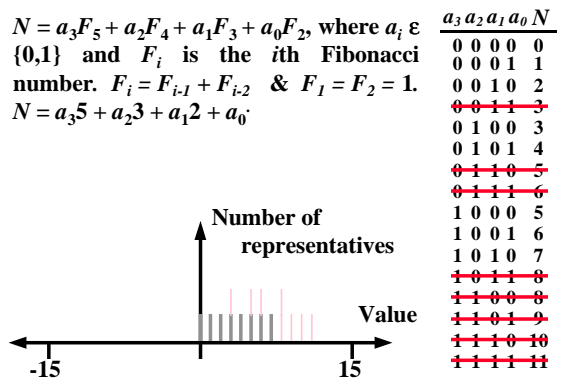


Figure 7. The 4-Bit Fibonacci Number System in Which No Pairs of Adjacent 1's Exist.

The absence of pairs of adjacent 1's makes this number system useful in systems that retrieve and store data serially. For example, Davies [6] describes a CD-ROM system in which codewords without pairs of 1's are used. Kautz [12] proposed codes that could be transmitted without an accompanying clock, requiring there be enough 0 to 1 and 1 to 0 transitions to delineate where bits begin and end.

### G. FIBONACCI NUMBER SYSTEM WITH THE MOST 1'S

Brown [1] showed that if one retains only those binary patterns in the Fibonacci number system with the most 1's, then, in the resulting system, each integer is uniquely represented. Stated differently, each representation with the most number of 1's is unique. Fig. 8 shows this for 4-bit patterns. In this case, the numbers 0, 1, ... , and 11 are uniquely represented.

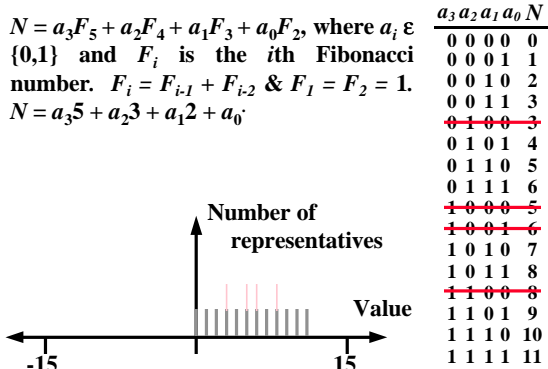


Figure 8. The 4-Bit Fibonacci Number System in Which the Most 1's Exist.

### H. NEGA-FIBONACCI NUMBER SYSTEM

Another extension of the Fibonacci number system represents negative integers as well as positive integers. Let  $F_i = (-1)^{i+1}F_i$ , and let

$$N = a_{n-1}F_{-(n+1)} + \dots + a_2 F_{-4} + a_1 F_{-3} + a_0 F_{-2},$$

where  $a_i \in \{0,1\}$  and  $F_i$  is a Fibonacci number. Negative values of  $N$  occur when the sum of negative Fibonacci numbers exceeds the sum of positive numbers in the representation. The case of  $n = 4$  is shown in Fig. 9. Note that this number system has many redundant representatives.

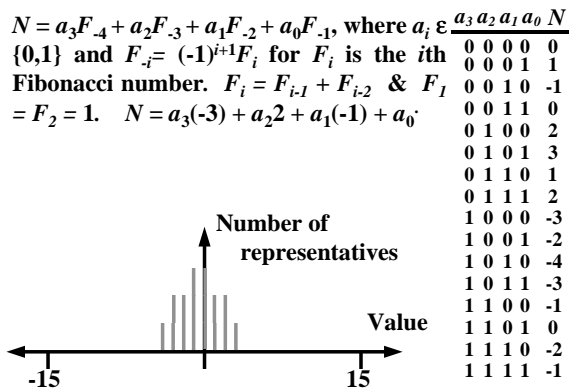


Figure 9. The 4-Bit Nega-Fibonacci Number System.

### I. NEGA-FIBONACCI NUMBER SYSTEM WITHOUT ADJACENT 1'S

It has been shown Bunder [3] that, in the nega-Fibonacci number system, if the tuples with adjacent 1's are removed, then the remaining tuples represent distinct values. Fig. 10 below shows that, for the 4-bit system, there are 8 remaining tuples, representing the values -4 through 3.

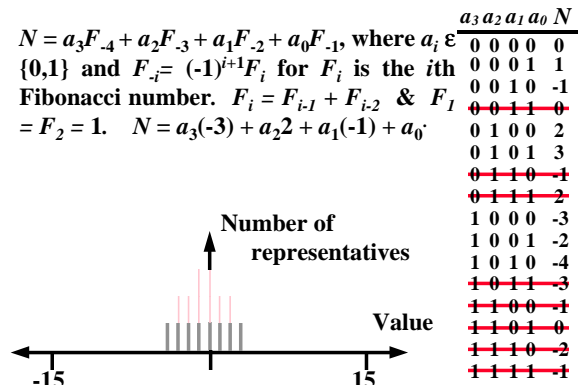


Figure 10. The 4-Bit Nega-Fibonacci Number System.

### J. TRIBONACCI NUMBER SYSTEM

A natural extension of the Fibonacci number system is the tribonacci number system, Capocelli et al [5] and Fraenkel [8]. In this system, each number has the form

$$N = a_{n-1}F_{n+1} + \dots + a_2 F_4 + a_1 F_3 + a_0 F_2,$$

where  $F_i = F_{i-1} + F_{i-2} + F_{i-3}$  and  $F_4 = 2^2$ ,  $F_3 = 2^1$ , and  $F_2 = 2^0$ . Fig. 11 shows that, for the 4-bit tribonacci number system, there is one redundant representative.

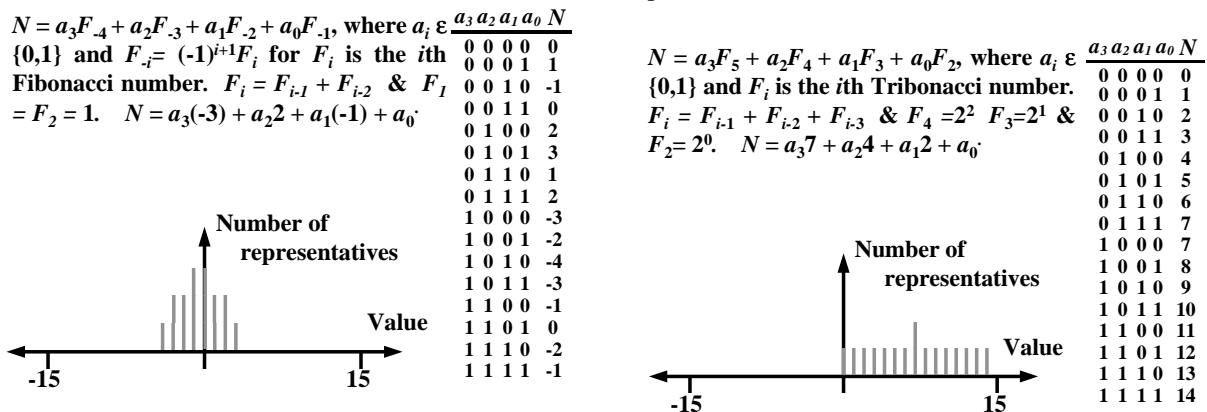


Figure 11. The 4-Bit Tribonacci Number System.

### K. RESTRICTED TRIBONACCI NUMBER SYSTEM

Fraenkel [8] has shown that if there are no groups of three or more consecutive 1's, then the representations are unique. Fig. 12 shows that, for the 4-bit tribonacci number system, eliminating 4-tuples with three or more consecutive 1's yields a number system in which 0, 1, ..., 12 have unique representations.

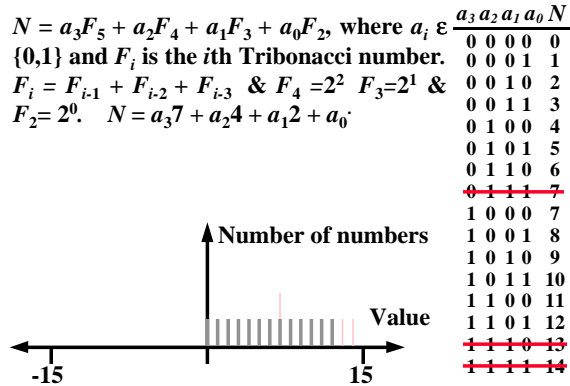


Figure 12. The 4-Bit Tribonacci Number System in Which No Groups of Three or More Consecutive 1's Occurs.

This result can be made more general. Indeed, Fraenkel [8] has shown that in number systems where  $F_i$  is given as

$$F_i = F_{i-1} + F_{i-2} + \dots + F_{i-m},$$

for  $i > m+1$  and  $F_i = 2^{i-2}$ , for  $1 < i \leq m+1$ , then if there are no more than  $m-1$  consecutive 1's, the representations are unique.

### L. FIBONACCI-LIKE NUMBER SYSTEM

Another natural extension of the Fibonacci number system is a Fibonacci-like number system Klein [13], where each number has the form

$$N = a_{n-1}F_{n+1} + \dots + a_2F_4 + a_1F_3 + a_0F_2,$$

where  $F_i = F_{i-1} + F_{i-2}$  for  $i > m+1$ , and  $F_i = i-1$ , for  $1 < i \leq m+1$ , where  $m \geq 2$ . Fig. 13 shows a 4-bit Fibonacci-like number system. This is redundant; five duplicate representations exist.

### L. FIBONACCI-LIKE NUMBER SYSTEM IN WHICH EACH PAIR OF 1'S IS SEPARATED BY TWO OR MORE 0'S.

From Theorem 1 of Fraenkel [8], we can conclude that if every pair of 1's is separated

$N = a_3F_5 + a_2F_4 + a_1F_3 + a_0F_2$ , where  $a_i \in \{0,1\}$  and  $F_i$  is the  $i$ th Fibonacci-like number.  $F_i = F_{i-1} + F_{i-3}$  &  $F_4 = 3$ ,  $F_3 = 2$ , &  $F_2 = 1$ .  $N = a_34 + a_23 + a_12 + a_0$ .

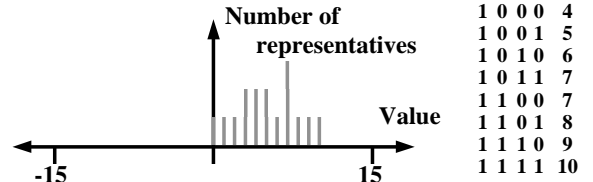


Figure 13. The 4-Bit Fibonacci-Like Number System.

by at least  $m-1$  0's, then the tuples are unique. Fig. 14 shows that, if we do this in the 4-bit system, then the numbers 0, 1, ..., 5 are represented uniquely.

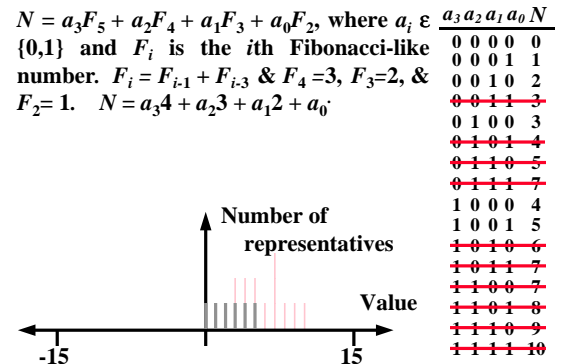


Figure 14. The 4-Bit Fibonacci-Like Number in Which Each Pair of 1's is Separated by Two or More 0's.

### M. m-NACCI NUMBER SYSTEM

Another natural extension of the Fibonacci number system is a multiple-valued system in which a number is represented by

$$N = a_{n-1}F_{n+1} + \dots + a_2F_4 + a_1F_3 + a_0F_2,$$

where the digits,  $F_i$ , are 0, 1, ..., and  $m-1$ , and the basis is given as

$$F_i = m F_{i-1} - F_{i-2}$$

for  $i > 3$ , and  $F_3 = m$ , and  $F_2 = 1$ , where  $m \geq 3$ . Fig. 15 shows a 3-digit system, where  $m = 3$ . Here, there are two redundant representatives.

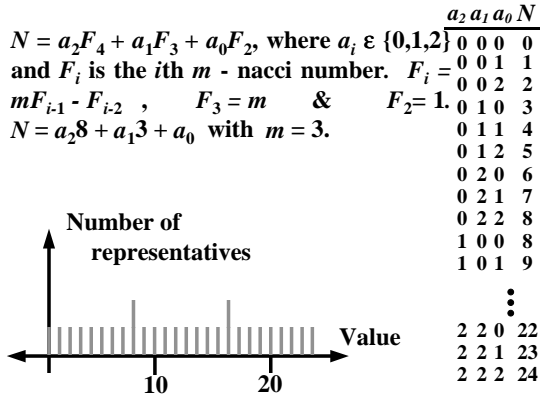


Figure 15. The 3-Digit  $m$ -nacci- Number System, Where  $m = 3$ .

**N.  $m$ -NACCI NUMBER SYSTEM WHERE EACH PAIR OF  $m$ -1'S IS SEPARATED BY AT LEAST ONE  $i$  SUCH THAT  $i \in \{0,1,\dots,m-3\}$**

Klein [13] has shown that in the  $m$ -nacci number system, if each  $m-1$  is separated by at least one  $i$  such that  $i \in \{0,1,\dots,m-3\}$ , then each number has a unique representation. Fig. 16 shows that if we apply this restriction to the number system shown in Fig. 15, then six of the 27 tuples are redundant, leaving 21 numbers, each with a unique representation.

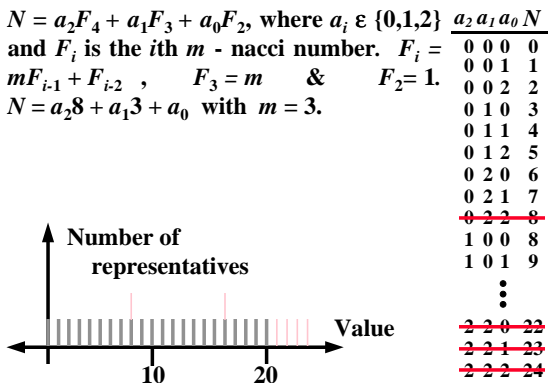


Figure 16. The 3-Bit  $m$ -nacci- Number System, Where  $m = 3$  and Every Pair of 2's is Separated by At Least One 0.

**IV. MEASURING REDUNDANCY**

In Butler and Sasao [4], we ask and answer the question

*To what extent does redundancy exist in redundant number systems?*

The question is important when we restrict the tuples allowed in the system. For example, we are interested in the extent to which the restriction - *no adjacent 1's* - reduces the tuples available to represent numbers in the Fibonacci number system. We measure redundancy in two ways 1) the number of tuples available to represent values and 2) the percentage of various digits over all representations. For example, we expect that the restriction - *no adjacent 1's* - to reduce the number of 1's in comparison to the number of 0's. We are interested in the extent to which this happens.

Our first result shows the distribution of non-redundant representations in the Fibonacci, tribonacci, quadranacci, etc. number systems. Specifically, Fig. 17 shows the number of representations versus the number of 1's in 16-bit representatives. That is, we consider 16-bit tuples whose value is given by

$$N = a_{15} F_{17} + \dots + a_2 F_4 + a_1 F_3 + a_0 F_2, \quad (1)$$

where  $F_i = F_{i-1} + F_{i-2} + \dots + F_{i-m}$ , for  $i > m+1$  and  $F_i = 2^{i-2}$ , for  $1 < i \leq m+1$ . For  $m = 2, 3$ , and 4, this represents the Fibonacci, tribonacci, and quadranacci number systems, respectively. It is

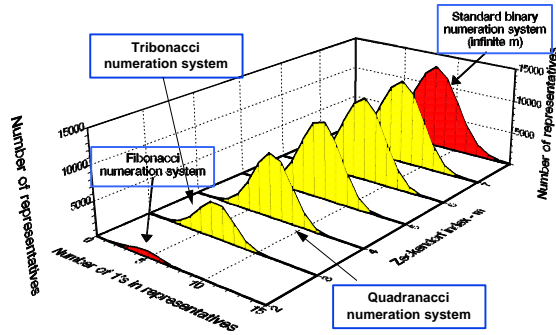


Figure 17. [4] The distribution of non-redundant representatives in 16-bit number systems.

interesting that the Fibonacci number system has very few representatives (in which pairs of 1's are not allowed) compared to the tribonacci number system (in which triples of 1's are not allowed). By way of comparison, the distribution of 1's in the standard binary number system (which corresponds to  $m = \infty$ ) is shown. Table 1 below

shows the proportion of bits that are 1 in the allowed representatives when the number of bits is large.

$m$	Proportion of 1's
2	0.2764
3	0.3816
4	0.4337
5	0.4621
6	0.4782
7	0.4875
8	0.4929
$\infty$	0.5000

Table 1.[4] The proportion of bits that are 1 in various number systems.

Fig. 18 shows the distribution of 1's for the generalized Fibonacci number systems. In these systems, a representative represents a number as in (1), where  $F_i = F_{i-1} + F_{i-m}$  for  $i > m+1$ , and  $F_i = i-1$ , for  $1 \leq i \leq m+1$ , where  $m \geq 2$ . As in the previous example,  $m = 2$  corresponds to the Fibonacci number system. It is interesting, that, in this case, that there are significantly more representatives when  $m = 2$  than when  $m = 3$ .

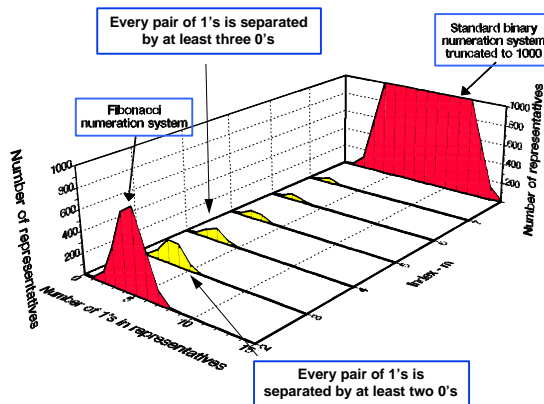


Figure 18. [4] The distribution of non-redundant representatives in 16-bit number systems.

Table 2 shows the proportion of digits that are 0, 1, ...,  $m-1$  in  $m$ -nacci number systems.

## V. REDUNDANCY IN ARITHMETIC OPERATIONS

Kaneyama et al [11] and Harata et al [10] have demonstrated the advantages of a redundant number system in multiple-valued multipliers. The following example is representative of their

implementation. Let an integer  $N$  have the representation

$m$	Proportion of 0's	Proportion of 1's, 2's, ..., $m-1$ 's
2	0.7236	0.2764
3	0.8057	0.3816
4	0.8492	0.4337
5	0.8762	0.4621
6	0.8948	0.4782
7	0.9084	0.4875
8	0.9188	0.4929
$\infty$	$1-1/m$	$1/m$

Table 2.[4] The proportion of bits that are 0, 1, ...,  $m-1$  in various number systems

$$N = a_{n-1}2^{n-1} + \dots + a_22^2 + a_12^1 + a_02^0,$$

where  $a_i \in \{0,1,2,3\}$ . This number system is redundant; for example,  $101 = 021 = 5$ . Addition, in this system, is done without carries across long strings of digits. Specifically, carries occur across only three digits. As a result, there is little delay associated with carry. To illustrate, consider the addition of  $10231 + 11111$ . Both  $10231$  and  $11111$  are 31 and their sum is 62. The first step is to perform the addition digit by digit, where there is no carry in nor out of each digit. This is illustrated by the first three lines in Fig. 19 below. Notice that the sum,  $Z$ , has a digit, 4, that is not allowed in this system. The next step is to convert all digits into their binary equivalent. The maximum digit is 6 ( $3+3$ ), and so three bits are necessary to represent such a number. The next three lines in Fig. 17 show the binary equivalent of the sum digits. These bits are then added, forming a sum  $S$ . In Fig. 19, this is the seventh line. Note that the largest digit in  $S$  is 3, since we add at most three 1 bits. The resulting sum is equivalent to  $Z$  and the digits are between 0 and 3. Thus, it is a number in the number system.

$$\begin{array}{r}
 A = 1 \ 0 \ 2 \ 3 \ 1 \\
 B = 1 \ 1 \ 1 \ 1 \ 1 \\
 \hline
 Z = A+B = 2 \ 1 \ 3 \ 4 \ 2 \\
 C_0 = \quad \quad 0 \ 1 \ 1 \ 0 \ 0 \\
 C_1 = \quad \quad 1 \ 0 \ 1 \ 0 \ 1 \\
 C_2 = \quad \quad 0 \ 0 \ 0 \ 1 \ 0 \\
 \hline
 S = 0 \ 1 \ 0 \ 3 \ 1 \ 1 \ 0
 \end{array}$$

Figure 19 [11]. An example of addition in a redundant number system.



In performing the addition, it is important to note that carries occur across three digits only. Fig. 20 shows an adder which performs the addition shown in the example of Fig. 19. This shows clearly the extent to which the carries affect the sum. The longest path in this circuit occurs through a (wired) summation gate, an MVL-to-Binary-Converter, and another summation gate.

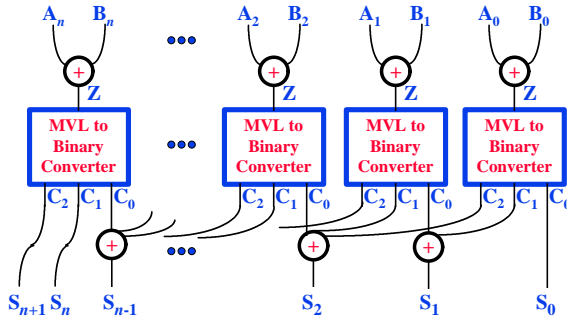


Figure 20 [11]. An example circuit for the addition of redundant numbers.

It is interesting to note that addition is done in constant time; that is, it is done in time independent of the number of digits.

## VI. CONCLUSIONS

We have considered number systems spanning from binary to multiple-valued and from non-redundant to redundant. While non-redundant number systems represent more numbers than redundant number systems, the latter has significant advantages with respect to transmission or storage of data and with respect to high-speed arithmetic operations.

## REFERENCES

1. J. L. Brown, "Zeckendorf's theorem and some applications," *The Fibonacci Quarterly* Vol. 2.2 1964, pp. 163-168.
2. J. L. Brown, "A new characterization of the Fibonacci numbers" *The Fibonacci Quarterly*, Vol. 3.1, 1965, pp. 1-8.
3. M. W. Bunder, "Zeckendorf representation using negative Fibonacci numbers," *The Fibonacci Quarterly*, May 1992, pp. 111-115.
4. J. T. Butler and T. Sasao, "On the proportion of digits in redundant numeration systems," *The Fibonacci Quarterly*, Vol. May 1997, pp. 172-180.
5. R. M. Capocelli, G. Gerbone, P. Cull, and J. Hollaway, "Fibonacci facts and formulas," and "Sequences" in *Int. Conf. on Combinatorics,*

- Compression, Security, and Transmission, Ed. R. M. Capocelli, New York: Springer-Verlag, 1990, pp. 133-137.
6. D. H. Davies, "The CD-ROM medium," *J. Amer. Soc. for Inform. Sci.* Vol. 39.1, 1988, pp. 34-42.
7. V. S. Dimitrov and B. D. Donevsky, "Faster multiplication of medium large numbers via the Zeckendorf representation," *The Fibonacci Quarterly*, Vol. 33.1, 1995, pp. 74-75.
8. A. S. Fraenkel, "Systems of numeration," *Amer. Math. Monthly*, Vol. 92, 1985, pp. 105-114.
9. A. Glaser, *History of binary and other nondecimal numeration*, Los Angeles: Tomash Publishers, 1971.
10. Y. Harata, Y. Nakamura, H. Nagase, M. Takigawa, and N. Takagi, "A high-speed multiplier using redundant binary adder tree," *IEEE J. of Solid-State Cir.*, Feb. 1987, pp. 28-34.
11. M. Kameyama, S. Kawahito, and T. Higuchi, "A multiplier chip with multiple-valued bidirectional current-mode logic circuits," *Computer*, Vol. 21, No. 4, Apr. 1988, pp. 43-56.
12. W. H. Kautz, "Fibonacci codes for synchronization control," *IEEE Trans. on Infor. Th.*, Vol. IT-11, 1965, pp. 284-292.
13. S. T. Klein, "Combinatorial representations of generalized Fibonacci numbers," *The Fibonacci Quarterly*, Vol. 29.2, 1991, pp. 124-131.
14. P. Morrison and P. Morrison, "The physics of binary numbers," *Scientific American*, Feb. 1996, pp. 130-131.
15. J. Newman, *The World of Mathematics*, New York: Simon and Shuster, 1956
16. E. Zeckendorf, "Representation des nombres naturels par une somme de nombres de Fibonacci ou de nombres de Lucas," *Bull. Soc. Royale Sci. Liege* 41, 1972, pp. 179-182.

## **ABSTRACT**

We survey number systems in which the digits are multiple-valued and the representations are redundant. In a redundant number system, there is at least one value that can be written in at least two ways. As a basis of comparison, we consider also non-redundant binary number systems, including the standard binary number system. We compare systems on the basis of redundancy - that is, how many redundant numbers exist.