International Journal of Security and Its Applications
Vol.7, No.6 (2013), pp.269-274
http://dx.doi.org/10.14257/ijsia.2013.7.6.27

Some Explicit Formulae of NAF and its Left-to-Right Analogue Based on Booth Encoding

Dong-Guk Han, Okyeon Yi †, and Tsuyoshi Takagi

Kookmin University, Kyushu University

Abstract. Non-Adjacent Form (NAF) is a canonical form of signed binary representation of integers. Joye-Yen proposed a left-to-right analogue of NAF (FAN). It is known that NAF and FAN can be generated by applying a sliding window method with width-2 to the Booth encoding in right-to-left and left-to-right direction, respectively. In this article, we derive some properties of Booth encoding such as pattern, classification, extension, adjacency, and length. [1]

Keywords: *signed binary representation, non-adjacent form, Booth encoding.*

1 Introduction

In some exponentiation-based public-key cryptosystems including RSA and Elliptic Curve Cryptosystems (ECC), a binary representation of a given integer (which may be a secret in most cases) is commonly used as a standard technique. While a non-signed representation of an integer is unique, we have some ways for representing the integer in signed form. For example, an integer 13 can be represented in signed form such as $10\bar{1}01$, $100\bar{1}\bar{1}$, or $10\bar{1}1\bar{1}$, where $\bar{1}$ denotes -1. Such signed binary representations are especially useful in ECC, since inversions of arbitrary points can be obtained with almost free operations over elliptic curves. Some properties of such signed binary representations are related to the cost of an exponentiation. Especially, the number of non-zero bits (Hamming weight) is important since this value rules the number of multiplications in the exponentiation. Thus analyzing signed representations implies a cost evaluation of exponentiations.

The non-adjacent form (NAF) is a well-known signed binary representation [7]. A NAF of a positive integer a is an expression $a = \sum_{i=0}^{n-1} \nu_i 2^i$ where $\nu_i \in \{-1, 0, 1\}$, $\nu_{n-1} \neq 0$ and no two successive digits are non-zero, i.e. $\nu_i \cdot \nu_{i+1} = 0$ for $i = 0, 1, ..., n-2$ [7]. Each integer a has a unique NAF representation denoted by $\textsc{Naf}(a)$. Moreover, $\textsc{Naf}(a)$ can be efficiently computed by right-to-left operations ([3], for example). The average Hamming weight of NAF has been found by Gollmann et al. (Corollary 2.6 in [2]), and more recently Wu and Hasan have presented a closed form expression for the average number of Hamming weight and length in a minimal weight radix-r signed-digit representation where the special case $r = 2$ is NAF [8].

In [4], Joye-Yen proposed a left-to-right analogue of NAF. We call it "FAN" as the reverse order of NAF. It is known that NAF and FAN can be generated by applying a sliding window method with width-2 to the Booth encoding [1] in right-to-left and left-to-right, respectively. Note that the Booth encoding was also introduced as the reversed binary representation by Knuth [5, Exercise 4.1-27]. The Booth encoding and FAN of an integer a are denoted by $\textsc{Booth}(a)$ and $\textsc{Fan}(a)$, respectively.

[1] Corresponding Author†

ISSN: 1738-9976 IJSIA

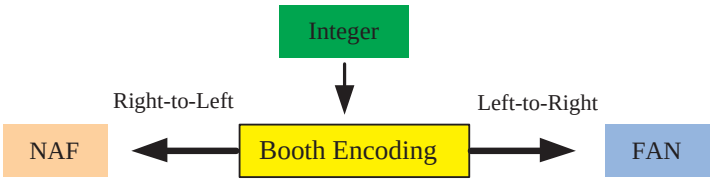


Fig. 1. A relation of the Booth encoding, NAF, and FAN.

In this paper, we provide some explicit formulae of NAF and FAN by using fundamental properties induced from Booth encoding. We propose some results for properties of Booth encoding and relations about Pattern, Classification, Extension, Adjacency, and Length among Booth, NAF, and FAN(Section 3).

2 Preliminaries

2.1 Booth Encoding, NAF and FAN

The n-bit Booth encoding [1] is an n-bit signed binary representation that satisfies the following two conditions:

- Signs of adjacent non-zero bits (without considering zero bits) are opposite.
- The most significant non-zero bit and the least significant non-zero bit are 1 and $\bar{1}$, respectively, unless all bits are zero.

In [6], they proved that for each integer there exists only one representation that satisfies the Booth recoding properties and showed a simple conversion method from an n-bit binary string to $(n+1)$-bit Booth encoding. Given an integer a, the Booth encoding of a is obtained by

$$2a \ominus a,$$

where $\ominus$ stands for a bitwise subtraction.

The non-adjacent form (NAF) also represents an integer in signed form [7]. Since there is no successive non-zero bits in the representation, NAF is a standard technique for computing exponentiations [3]. NAF can be interpreted as a combination of the Booth encoding and a right-to-left sliding window method with width-2 (SW_2^{r-t-l}), i.e. width-2 window, moving right-to-left, skipping consecutive zero entries after a nonzero digit is processed. 01, $0\bar{1}$, $1\bar{1}$, and $\bar{1}1$ are converted to 01, $0\bar{1}$, 01, and $0\bar{1}$, respectively.

$$\text{Integer} \Longrightarrow \text{Booth Recoding} \overset{SW_2^{r-t-l}}{\Longrightarrow} \text{NAF representation}$$

FAN was introduced as a left-to-right analogue of NAF [4]. In fact, FAN can be also interpreted as a combination of the Booth encoding and a left-to-right sliding window method with width-2 (SW_2^{l-t-r}). 10, $\bar{1}0$, $1\bar{1}$, and $\bar{1}1$ are converted to 10, $\bar{1}0$, 01, and $0\bar{1}$, respectively.

$$\text{Integer} \Longrightarrow \text{Booth Recoding} \overset{SW_2^{l-t-r}}{\Longrightarrow} \text{FAN representation}$$

2.2 Notations

For a given binary n-bit integer a (between 0 and 2^n-1) we use the following notations:

- $\textsc{Binary}(a)$, $\textsc{Booth}(a)$, $\textsc{Naf}(a)$, and $\textsc{Fan}(a)$ denote the binary, Booth encoding, NAF, and FAN representation of the integer a respectively.
  - $\textsc{Binary}(a) := (a_{n-1}, ..., a_1, a_0)_2$ with $a_i \in \{0, 1\}$,
  - $\textsc{Booth}(a) := (\beta_n, ..., \beta_1, \beta_0)_2$ with $\beta_i \in \{-1, 0, 1\}$,
  - $\textsc{Naf}(a) := (\nu_n, ..., \nu_1, \nu_0)_2$ with $\nu_i \in \{-1, 0, 1\}$,
  - $\textsc{Fan}(a) := (\phi_n, ..., \phi_1, \phi_0)_2$ with $\phi_i \in \{-1, 0, 1\}$.

  Note that throughout this paper each of the symbols β_i, ν_i, and ϕ_i are only utilized to denote a digit of the representation of Booth, NAF, and FAN respectively.
- $\mathcal{B}(n) := \{\textsc{Booth}(a) \mid 0 \le a \le 2^n - 1\}$, that is a set of all Booth encodings of integers between 0 and $2^n - 1$.
  - $Case\_I\_\mathcal{B}(n) := \{\textsc{Booth}(a)$ with $\beta_n = 0 \mid 0 \le a \le 2^n - 1\}$.
  - $Case\_II\_\mathcal{B}(n) := \{\textsc{Booth}(a)$ with $(\beta_n, \beta_{n-1}) = (1, \bar{1}) \mid 0 \le a \le 2^n - 1\}$.
  - $Case\_III\_\mathcal{B}(n) := \{\textsc{Booth}(a)$ with $(\beta_n, \beta_{n-1}) = (1, 0) \mid 0 \le a \le 2^n - 1\}$.
- $\mathcal{N}(n) := \{\textsc{Naf}(a) \mid 0 \le a \le 2^n - 1\}$, that is a set of all NAF representations of integers between 0 and $2^n - 1$.
- $\mathcal{F}(n) := \{\textsc{Fan}(a) \mid 0 \le a \le 2^n - 1\}$, that is a set of all FAN representations of integers between 0 and $2^n - 1$.
- ε_n is the negligible function in n, namely for every constant $c \ge 0$ there exists an integer m such that $|\varepsilon_n| \le 1/n^c$ for all $n \ge m$.
- If t is a real number, then $\lfloor t \rfloor$ is the largest integer $\le t$ and $\lceil t \rceil$ is the smallest integer $\ge t$.

2.3 Some cases of NAF and FAN

In this section, we show that how to compute NAF and FAN representations from the Booth encoding. For example, for an integer $13 = (1, 1, 0, 1)_2$, we have $\textsc{Booth}(13) = (1, 0, \bar{1}, 1, \bar{1})_2$ from Algorithm **??**. Then we divide $\textsc{Booth}(13)$ (as a string) into width-2 windows from *right to left*: 01, $0\bar{1}$, $1\bar{1}$ (the leftmost 0 was padded), and convert $1\bar{1}$ to 01 and $\bar{1}1$ to $0\bar{1}$, if any. Thus we have $\textsc{Naf}(13) = (1, 0, \bar{1}, 0, 1)_2$.

In order to generate FAN representation of 13, we divide the string of $\textsc{Booth}(13)$ into width-2 windows from *left to right*: 10, $\bar{1}1$, $\bar{1}0$ (the rightmost 0 was padded). Then, similarly to NAF, convert $1\bar{1}$ to 01 and $\bar{1}1$ to $0\bar{1}$, if any. Thus we have $\textsc{Fan}(13) = (1, 0, 0, \bar{1}, \bar{1})_2$. Note that FAN can have successive non-zero bits unlike NAF.

3 Several results of Booth encodings

In this section, we prove several results for the Booth encodings such as Pattern, Classifications, Extension, Adjacency, and Length.

3.1 Some Properties of Booth Encoding

Property 1. Due to the definition of Booth encoding (refer to Section 2.1), the Hamming weight of $\textsc{Booth}(a)$ is always even, if the original integer a is positive.

Table 1. NAF, FAN, Booth encoding representations of some integers

Integer	Signed Binary			Non-signed
a	$\textsc{Naf}(a)$	$\textsc{Fan}(a)$	$\textsc{Booth}(a)$	Binary
0	0000000	0000000	0000000	0000000
1	0000001	0000001	000001$\bar{1}$	0000001
2	0000010	0000010	00001$\bar{1}$0	0000010
3	000010$\bar{1}$	000010$\bar{1}$	000010$\bar{1}$	0000011
4	0000100	0000100	0001$\bar{1}$00	0000100
5	0000101	0000101	0001$\bar{1}$1$\bar{1}$	0000101
6	00010$\bar{1}$0	00010$\bar{1}$0	00010$\bar{1}$0	0000110
7	000100$\bar{1}$	000100$\bar{1}$	000100$\bar{1}$	0000111
8	0001000	0001000	001$\bar{1}$000	0001000
9	0001001	0001001	001$\bar{1}$01$\bar{1}$	0001001
10	0001010	0001010	001$\bar{1}$1$\bar{1}$0	0001010
11	0010$\bar{1}$0$\bar{1}$	000110$\bar{1}$	001$\bar{1}$10$\bar{1}$	0001011
12	0010$\bar{1}$00	0010$\bar{1}$00	0010$\bar{1}$00	0001100
13	0010$\bar{1}$01	00100$\bar{1}$$\bar{1}$	0010$\bar{1}$1$\bar{1}$	0001101
14	00100$\bar{1}$0	00100$\bar{1}$0	00100$\bar{1}$0	0001110
15	0010000$\bar{1}$	0010000$\bar{1}$	0010000$\bar{1}$	0001111

Let $\langle 1\bar{1}\rangle^k$ be a pattern of non-zero bits in Booth encoding such that $\overbrace{1,\bar{1}}^{k},\ldots,\overbrace{1,\bar{1}}^{2},\overbrace{1,\bar{1}}^{1}$ (exactly k-times) after omitting all zero bits between 1 and $\bar{1}$. Let $\#[\langle 1\bar{1}\rangle^k]$ be the total number of strings having $\langle 1\bar{1}\rangle^k$ pattern. For example, in $\mathcal{B}(4)$, $\#[\langle 1\bar{1}\rangle^1] = 10$ (i.e. integers 1,2,3,4,6,7,8,12,14,15) and $\#[\langle 1\bar{1}\rangle^2] = 5$ (i.e. integers 5,9,10,11,13). Refer to the fourth column of Table 1.

Theorem 1 (Pattern). *$\mathcal{B}(n)$ consists of exactly all possible representations with $\langle 1\bar{1}\rangle^k$ pattern for $0 \le k \le \lceil n/2 \rceil$.*

Proof. For $0 \le k \le \lceil n/2 \rceil$,

$$\#[\langle 1\bar{1}\rangle^0] = \binom{n+1}{0}, \#[\langle 1\bar{1}\rangle^1] = \binom{n+1}{2}, \ldots, \#[\langle 1\bar{1}\rangle^k] = \binom{n+1}{2k}, \ldots, \#[\langle 1\bar{1}\rangle^{\lceil n/2 \rceil}] = \binom{n+1}{2\lceil n/2 \rceil},$$

where the binomial coefficient $\binom{a}{b}$ denotes the number of b-combinations from a set S with a elements.

Thus $\sum_{k=0}^{\lceil n/2 \rceil} \#[\langle 1\bar{1}\rangle^k] = \sum_{k=0}^{\lceil n/2 \rceil} \binom{n+1}{2k} = 2^n$ from $\sum_{k=0}^{n+1} \binom{n+1}{k} = 2^{n+1}$ and $\sum_{k=0} \binom{n+1}{2k} = \sum_{k=0} \binom{n+1}{2k+1}$. This implies that there are 2^n different representations with $\langle 1\bar{1}\rangle^k$ pattern. As the Booth recoding is unique, the assertion is proved. □

Theorem 2 (Classification). *$\mathcal{B}(n)$ can be divided into the following three cases;*

i) $Case\_I\_\mathcal{B}(n)$ with $\#[Case\_I\_\mathcal{B}(n)]=2^{n-1}$,
ii) $Case\_II\_\mathcal{B}(n)$ with $\#[Case\_II\_\mathcal{B}(n)] = 2^{n-2}$,
iii) $Case\_III\_\mathcal{B}(n)$ with $\#[Case\_III\_\mathcal{B}(n)] = 2^{n-2}$.

Proof. From Property 1 and Lemma 1,

$$\#[Case\_I\_\mathcal{B}(n)] = \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n}{2k} = 2^{n-1},$$

$$\#[Case\_II\_\mathcal{B}(n)] = \sum_{k=0}^{\lfloor \frac{n-1}{2} \rfloor} \binom{n-1}{2k} = 2^{n-2},$$

$$\#[Case\_III\_\mathcal{B}(n)] = 2^{n-2} = \begin{cases} \sum_{k=0}^{\frac{n}{2}-1} \binom{n-1}{2k+1}, & \text{(if } n \text{ is even)} \\ \sum_{k=0}^{\frac{n-3}{2}} \binom{n-1}{2k+1}. & \text{(otherwise)} \end{cases}$$

□

Theorem 3 (Extension). *$\mathcal{B}(n)$ can be constructed from $\mathcal{B}(n-1)$ according to the following rules;*

i) $Case\_I\_\mathcal{B}(n) = \{(\beta_n = 0) \| (\beta_{n-1}, \ldots, \beta_0) \mid (\beta_{n-1}, \ldots, \beta_0) \in \mathcal{B}(n-1)\}$,
ii) $Case\_II\_\mathcal{B}(n) = (\beta_n, \beta_{n-1}) = (1, \bar{1}) \| (\beta_{n-2}, \ldots, \beta_0) \mid (\beta_{n-2}, \ldots, \beta_0) \in \mathcal{B}(n-2)$,
iii) $Case\_III\_\mathcal{B}(n) = (\beta_n, \beta_{n-1}) = (1, 0) \| (\beta_{n-2}, \ldots, \beta_0) \mid (1, \beta_{n-2}, \ldots, \beta_0) \in \{\mathcal{B}(n-1) - Case\_I\_\mathcal{B}(n-1)\}$.

Proof. From Property 1 and Lemma 1, 2, we can see that the assertion is true. □

Theorem 4 (Case II-Classification).
i) $\#\big[a \in Case\_II\_\mathcal{B}(n) \mid$ *the number of most significant consecutive non-zero bits of a is t, where t is even*$\big] = \frac{2^{n-1}+\kappa_n}{3}$,
ii) $\#\big[a \in Case\_II\_\mathcal{B}(n) \mid$ *the number of most significant consecutive non-zero bits of a is t, where t is odd*$\big] = \frac{2^{n-2}-\kappa_n}{3}$,
where $\kappa_n = 2$ if n is odd and $\kappa_n = 1$ if n is even. Especially,
iii) $\#\big[a \in Case\_II\_\mathcal{B}(n) \mid$ *the number of most significant consecutive non-zero bits of a is 2*$\big] = 2^{n-3}$.

Proof. Let n be even and $s_t := \#\big[a \in Case\_II\_\mathcal{B}(n) \mid$ the number of most significant consecutive non-zero bits of a is even $t\big]$. Then, based on Property 1

$$s_2 = \sum_{k=0}^{\frac{n}{2}-1} \binom{n-2}{2k} = 2^{n-3}, s_4 = \sum_{k=0}^{\frac{n}{2}-2} \binom{n-4}{2k} = 2^{n-5}, \cdots, s_{2t} = \sum_{k=0}^{\frac{n}{2}-t} \binom{n-t}{2k} = 2^{n-2t-1},$$

$$\cdots, \; s_{n-2} = \sum_{k=0}^{1} \binom{2}{2k} = 2, s_n = 1.$$

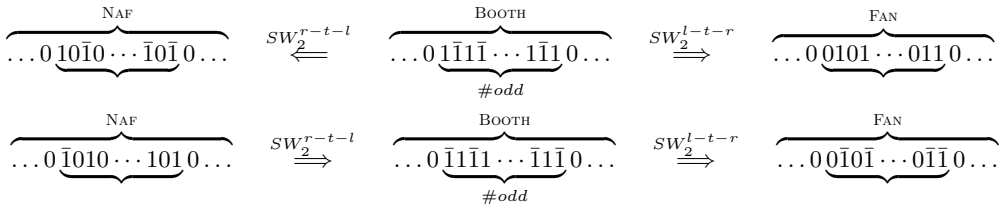
Thus, the result of Lemma 4.*i*) when n is even is $\sum_{k=1}^{\frac{n}{2}} s_{2k} = \frac{2^{n-1}+1}{3}$. From Lemma 2.*ii*), we can derive the result of Lemma 4.*ii*) when n is even from the above result that is $2^{n-2} - \frac{2^{n-1}+1}{3} = \frac{2^{n-2}-1}{3}$.

Similarly, the result of Lemma 4.*i*) when n is odd is $\frac{2^{n-1}+2}{3}$ and the result of Lemma 4.*ii*) when n is odd is $2^{n-2} - \frac{2^{n-2}+2}{3} = \frac{2^{n-1}-2}{3}$. The third one is clearly s_2. □

3.2 Relations among Booth, NAF, and FAN

Theorem 5 (Adjacent). *A substring with an odd number (> 1) of consecutive non-zero bits in Booth representations is converted into a substring with 11 or $\bar{1}\bar{1}$ at the least significant bits of the substring when transforming to FAN representation. Moreover,*

the resulting string in FAN is shorter than the original string in Booth, i.e. the most significant 1 *of the FAN string will be one position lower than that of the Booth string. In the case of NAF string, the most significant* 1 *or* $\bar{1}$ *stays at the same position.*



Define $L[a]$ as the bit-length of a representation of a, counting the bits from the least significant to the most significant 1. For example, if $a = (10110)_2$ then $L[\text{BINARY}(a)] = 5$. The results of Lemma 5 directly serves the proof of Lemma 6.

Theorem 6 (Length). *For an arbitrary integer a*

i.) $L[\text{BOOTH}(a)$ with $\#(\textit{the most significant consecutive nonzero bits}) = \textit{even}]$
$= L[\text{NAF}(a)] + 1, = L[\text{FAN}(a)] + 1,$

ii.) $L[\text{BOOTH}(a)$ with $\#(\textit{the most significant consecutive nonzero bits}) = \textit{odd}\ (> 1)]$
$= L[\text{NAF}(a)] = L[\text{FAN}(a)] + 1,$

iii.) $L[\text{BOOTH}(a)$ with $\#(\textit{the most significant consecutive nonzero bits}) = 1]$
$= L[\text{NAF}(a)] = L[\text{FAN}(a)].$

4 Conclusion

In this article, we derived several interesting contributions from the relation between NAF and FAN based on the Booth encoding. The results would be applied to analyze the probability of the NAF representation of an n-bit integer, the average Hamming weight of NAF and FAN, and the average length of zero runs in both the NAF and FAN.

This work was supported by the IT R&D program of MKE/KEIT[10039140, Developement of Crypto Algorithms(ARIA, SEED, KCDSA, etc.) for Smart Devices(ARM7,9,11, UICC)].

References

1. A. Booth, "A signed binary multiplication technique", Journ. Mech. and Applied Math., 4(2), pp.236-240, 1951.
2. D. Gollmann, Y. Han, and C.J. Mitchell, "Redundant Integer Representations and Fast Exponentation", Designs, Codes, and Cryptography, vol. 7, pp.135-151, 1996.
3. IEEE 1363-2000, IEEE Standard Specifications for Public-Key Cryptography, 2000.
4. M. Joye, and S.-M. Yen, "Optimal Left-to-Right Binary Signed-digit Exponent Recoding", IEEE Transactions on Computers 49(7), pp.740-748, 2000.
5. D.E. Knuth, "*The Art of Computer Programming, vol. 2, Seminumerical Algorithms*, 2nd ed., Addison-Wesley, Reading, Mass, 1981.
6. K. Okeya, K. Schmidt-Samoa, C. Spahn, and T. Takagi, "Signed Binary Representations Revisited", IACR Cryptology ePrint Archive, 2004.
7. G.W. Reitwiesner, *Binary arithmetic*, Advances in Computers, vol.1, pp.231-308, 1960.
8. H. Wu, and M.A. Hasan, "Closed-Form Expression for the Average Weight of Signed-Digit Representations", IEEE Transactions on Computers 48(8), pp.848-851, 1999.

