1-1-2008

On binary reflected Gray codes and functions

Martin W. Bunder
University of Wollongong, mbunder@uow.edu.au

Keith P. Tognetti
University of Wollongong, tognetti@uow.edu.au

Glen Wheeler
University of Wollongong, glenw@uow.edu.au

On binary reflected Gray codes and functions

Abstract
The Binary Reflected Gray Code function b is defined as follows: If m is a nonnegative integer, then b(m) is the integer obtained when initial zeros are omitted from the binary reflected Gray code of length m. This paper examines this Gray code function and its inverse and gives simple algorithms to generate both. It also simplifies Conder's result that the jth letter of the kth word of the binary reflected Gray code of length n, is

$(2^n - 2^{n-j} - 1$

$[2^n - 2^{n-j-1} - k/2])$ mod 2,

by replacing the binomial coefficient by

$[(k-1)/(2^{n-j+1})+1/2]$.

Keywords
binary, reflected, Gray, codes, functions

Disciplines
Physical Sciences and Mathematics

Publication Details
Bunder, M. W., Tognetti, K. P. & Wheeler, G. (2008). On binary reflected Gray codes and functions. Discrete Mathematics, 308 (9), 1690-1700.

On Binary Reflected Gray Codes and Functions

Martin W. Bunder, Keith P. Tognetti and Glen E. Wheeler
School of Mathematics and Applied Statistics
University of Wollongong
Wollongong NSW 2522 AUSTRALIA
email: martin_bunder@uow.edu.au

Abstract

The Binary Reflected Gray Code function b is defined as follows: If m is a nonnegative integer, then $b(m)$ is the integer obtained when initial zeros are omitted from the binary reflected Gray code of length m.

This paper examines this Gray code function and its inverse and gives simple algorithms to generate both. It also simplifies Conder's result that the jth letter of the kth word of the binary reflected Gray code of length n, is

$$\binom{2^n - 2^{n-j} - 1}{\lfloor 2^n - 2^{n-j-1} - \frac{k}{2} \rfloor} \mod 2,$$

by replacing the binomial coefficient by

$$\left\lfloor \frac{k-1}{2^{n-j+1}} + \frac{1}{2} \right\rfloor.$$

1 Introduction

A binary Gray code of length n is a sequence $s_0, s_1, \ldots, s_{2^n-1}$ of the 2^n distinct n-bit strings (or words) of 0s and 1s, with the property that each s_i differs from s_{i+1} in only one digit. Gray codes were first designed to speed up telegraphy, but now have numerous applications such as in addressing microprocessors, hashing algorithms, distributed systems, detecting/correcting channel noise and in solving problems such as the Towers of Hanoi, Chinese Ring and Brain and Spinout. Cyclic binary Gray codes of length n also describe Hamiltonian paths around an n-dimensional hypercube.

A particular Gray code, the binary reflexive Gray code of length n, represents the integers 0 to $2^n - 1$, as n-bit strings $s_0, s_1, \ldots, s_{2^n-1}$. We will write $b(m)$ for s_m represented as an integer; $b(m)$ will be independent of n.

In this paper we study the function b, its orbits and the decoding function b^{-1}, and give new simple methods for evaluating $b(m)$ and $b^{-1}(m)$.

One result, in Theorem 6(ii), is that:

$$(b(m))_i = \left\lfloor \frac{m}{2^{i+1}} + \frac{1}{2} \right\rfloor \bmod 2$$

(where n_i denotes the coefficient of 2^i in the binary expansion of n); this is a major simplification of a result of Conder in [2].

Also we show that $b(m)$ can be represented using Nim sums.

2 Binary Reflexive Gray Codes

Table 1 below will help to illustrate the construction of the Binary Reflextive Gray Code (BRGC) of any length n. Decimal values of m and of the BRGC of m, written as $b(m)$, are also given.

Table 1

m	m in binary	BRGC of m	$b(m)$
0	0	0	0
1	1	1	1
2	10	11	3
3	11	10	2
4	100	110	6
5	101	111	7
6	110	101	5
7	111	100	4

The initial line, representing: 0 is the BRGC (of length 1) of 0, is given. Further lines are then generated by drawing (for successively $k = 0, 1, 2, \ldots, n-1$) a line below $m = 2^k - 1$ and doing a reflection about the line of all the numbers in the BRGC of m column. Then 2^k (i.e. a 1 in the currently empty $k+1$th place) is added. Finally, after the $k = n-1$ case of this algorithm, initial 0s can be added to make the words of length n, giving Binary Reflexive Gray Codes of length n (BRGC(n)). For each n, the top half of the table for the BRGC(n) of m (that is for $0 \leq m < 2^{n-1}$), with the initial zero deleted, will show the BRGC($n-1$) of m.

Table 1, once the initial 0s are added, gives BRGC(3).

3 The Function b

Clearly from the above description b is given by:

Definition 1 $\quad b(0) = 0, \quad b(2^k + i) = b(2^k - i - 1) + 2^k \quad (0 \leq i < 2^k).$

The Gray Code properties, given this definition, will be proved in Section 4. For this we need some notation and some lemmas.

Notation We will sometimes write a nonnegative integer m as $m_k m_{k-1} \ldots m_0$ where m_i (0 or 1) is the coefficient of 2^i in the binary expansion of m. We assume $m_k = 1$ unless $k = 0$ and $m_0 = 0$. If $k > 0$ we will let m_p denote the first 0 (if any) from the left in $m_k m_{k-1} \ldots m_0$.

Lemma 1 (i) $b(m_k m_{k-1} \ldots m_0) = 2^k + 2^p + b(m_{p-1} \ldots m_0)$.

(ii) If $m_k = m_{k-1} \ldots = m_0 = 1$ then $b(m_k m_{k-1} \ldots m_0) = 2^k$, that is $b(2^{k+1} - 1) = 2^k$ for all $k \geq 0$.

Proof By Definition 1:

(i)
$$\begin{aligned} & b(2^k + 2^{k-1} + \ldots + 2^{p+1} + m_{p-1}2^{p-1} + \ldots + m_0) \\ &= 2^k + b(2^k - 2^{k-1} \ldots - 2^{p+1} - m_{p-1}2^{p-1} - \ldots m_0 - 1) \\ &= 2^k + b(2^p + 2^p - m_{p-1}2^{p-1} - \ldots - m_0 - 1) \\ &= 2^k + 2^p + b(m_{p-1}2^{p-1} + \ldots + m_0). \end{aligned}$$

(ii)
$$\begin{aligned} b(2^k + 2^{k-1} + \ldots + 2 + 1) &= 2^k + b(2^k - 2^{k-1} - \ldots - 1 - 1) \\ &= 2^k + b(0) = 2^k. \end{aligned}$$

Corollary 2 If $2^k - 2^{p+1} \leq j < 2^k - 2^p$, then $b(2^k + j) = b(j + 2^{p+1} - 2^k) + 2^p + 2^k$.

Corollary 3 If $0 \leq j < 2^{k-1}$, then $b(2^k + j) = b(j) + 2^k + 2^{k-1}$.

Lemma 4 If $2^k \leq m < 2^{k+1}$ then $2^k \leq b(m) < 2^{k+1}$.

Proof By an easy induction, using Lemma 1.

Lemma 5 (i) If $0 \leq p < k$ and $2^k - 2^{p+1} \leq j < 2^k - 2^p$, then

(a) If $i = k$ or p, $b(2^k + j))_i = 1$.

(b) If $p < i < k$, $(b(2^k + j))_i = 0$.

(c) If $i < p$, $(b(2^k + j))_i = (b(j))_i$.

(ii) If $j = 2^k - 1$, then $(b(2^k + j))_k = 1$, while $(b(2^k + j))_i = 0$ for $i < k$.

Proof (i) (a) and (b) follow from Corollary 2 and Lemma 4.

(c) If $i < p = k - 1$, by Corollary 3

$$(b(2^k + j))_i = (b(j))_i.$$

If $i < p < k-1$ and $2^k - 2^{p+1} \leq j < 2^k - 2^p$ we have $0 \leq 2^{k-1} - 2^{p+1} \leq j - 2^{k-1} < 2^{k-1} - 2^p$ and by Corollary 2:

$$\begin{aligned} b(j) &= b(2^{k-1} + (j - 2^{k-1})) \\ &= 2^{k-1} + 2^p + b(j + 2^{p+1} - 2^k) \\ b(j) + 2^{k-1} &= b(2^k + j). \end{aligned}$$

So if $i < p < k - 1$, then $(b(j))_i = (b(2^k + j))_i$.

(ii) By Lemma 1(ii).

The following theorem gives three ways of quickly evaluating $b(m)$.

Theorem 6 For $m \geq 0$,

(i) $(b(m))_i = (m + 2^i)_{i+1}$

(ii) $(b(m))_i = \lfloor \frac{m}{2^{i+1}} + \frac{1}{2} \rfloor \bmod 2$

(iii) $(b(m))_i = (m_{i+1} + m_i) \bmod 2$.

Proof (i) Let $0 \leq m < 2^{k+1}$, then by Lemma 4, $0 \leq b(m) < 2^{k+1}$ and if $i > k$, $(m)_i = (b(m))_{i+1} = 0$.

We now assume $i \leq k$ and proceed by induction on m.

Case 1 $\underline{m = 0}$ $(b(0))_i = 0_i = 0 = (0 + 2^i)_{i+1}$.

Case 2 $\underline{m = 2^{k+1} - 1,\ k \geq 0}$.

$$\begin{aligned} m + 2^i &= 2^{k+1} + 2^{i-1} + 2^{i-2} + \ldots + 1 \\ \text{so} \quad (m + 2^i)_{i+1} &= 0 \quad \text{if} \quad i \neq k, \\ (m + 2^k)_{k+1} &= 1 \end{aligned}$$

so we have the result.

Case 3 $\underline{m = 2^k + j \text{ where } 2^k - 2^{p+1} \leq j < 2^k - 2^p \text{ and } 0 \leq p < k}$

By Lemma 5(i)(a)

$$\begin{aligned} (b(m))_k &= 1 = (2^k + j + 2^k)_{k+1} \\ (b(m))_p &= 1 = (2^k + j + 2^p)_{p+1} \end{aligned}$$

as $2^{k+1} - 2^p \leq 2^k + j + 2^p < 2^{k+1}$.

By Lemma 5(i)(b) if $p < i < k$,

$$(b(m))_i = 0 = (2^k + j + 2^i)_{i+1}$$

as $2^{k+1} + 2^i - 2^{p+1} \leq 2^k + j + 2^i < 2^{k+1} + 2^i - 2^p$.

By Lemma 5(i)(c) if $i < p$, by the induction hypothesis:

$$(b(m))_i = (b(j))_i = (j + 2^i)_{i+1} = (j + 2^k + 2^i)_{i+1}$$

as $i < p < k$.

(ii) By (i) and $n_{i+1} = \lfloor \frac{n}{2^{i+1}} \rfloor \bmod 2$.

(iii) By (i) if $m_i = 0$, $(b(m)) = m_{i+1} = m_{i+1} + m_i \bmod 2$.

If $m_i = 1$ $(b(m))_i = (m_{i+1} + 1) \bmod 2$

$$= (m_{i+1} + m_i) \bmod 2.$$

Note that, in Sharma and Khanna [5], part (ii) of our Theorem 6 is used as the definition of the BRGC; our Definition 1 is later proved as a theorem.

4 b has BRGC properties

We require b to be a one to one and onto map and, for each m, $b(m)$ and $b(m+1)$, in binary notation (i.e. the BRGC of m and $m+1$) to differ by one digit.

Lemma 7 $b : \{0, 1, \ldots, 2^n - 1\} \to \{0, 1, \ldots, 2^n - 1\}$ is one to one and onto.

Proof We prove by induction on i that

$$b(k) = b(m) \Rightarrow k_{n-i} = m_{n-i}$$

which proves that b is one to one. It then follows by Lemma 4 that b is onto.

$\underline{i = 0}$ If $b(k) = b(m)$ by Theorem 6 (i) and $m, k < 2^n$,

$(b(k))_n = (b(m))_n = (m + 2^n)_{n+1} = (k + 2^n)_{n+1} = 0 = m_n = k_n.$

$\underline{i > 0}$ We assume $b(k) = b(m)$ and $k_{n-i+1} = m_{n-i+1}$.

By Theorem 6 (iii)

$$\begin{aligned} k_{n-i+1} + k_{n-i} &= (m_{n-i+1} + m_{n-i}) \bmod 2 \\ \text{and so} \qquad k_{n-i} &= m_{n-i}. \end{aligned}$$

Lemma 8 There is exactly one i such that $(b(m))_i \neq (b(m+1))_i$.

Proof By Theorem 6 (ii) and the fact that

$$\left\lfloor \frac{m}{2^{i+1}} + \frac{1}{2} \right\rfloor \quad \text{and} \quad \left\lfloor \frac{m+1}{2^{i+1}} + \frac{1}{2} \right\rfloor$$

cannot differ by more than 1, we have $(b(m))_i \neq (b(m))_{i+1}$ if and only if

$$\left\lfloor \frac{m+1}{2^{i+1}} + \frac{1}{2} \right\rfloor - \left\lfloor \frac{m}{2^{i+1}} + \frac{1}{2} \right\rfloor = 1.$$

Letting $m = 2^{i+1}\ell + k$ where $0 \leq k < 2^{i+1}$ this condition becomes

$$\left\lfloor \frac{k+1}{2^{i+1}} + \frac{1}{2} \right\rfloor - \left\lfloor \frac{k}{2^{i+1}} + \frac{1}{2} \right\rfloor = 1,$$

which holds if and only if $k = 2^i - 1$, that is, if and only if $m + 1 = 2^i(2\ell + 1)$.

Note The i for which $(b(m))_i \neq (b(m+1))_i$ is the highest power of 2 to divide $m+1$.

5 BRGC Algorithms

A standard algorithm (given in Nashelsky [4]) for generating a BRGC is:

Algorithm 1 for b For each digit in an n-digit word m, starting from the right, if the digit to its left is 0 leave it as it is, while if the digit to its left is 1, change the digit.

This is effectively what we get from Theorem 6 (iii):

Algorithm 2 for b $(b(m))_i$ is $(m_{i+1} + m_i) \bmod 2$.

Even simpler is Algorithm 3, which follows from Algorithm 1 or 2.

Algorithm 3 for b For $m > 0$, $b(m)$ is m, in binary, with the first of any sequence of 1s or 0s becoming a 1 and every other digit a 0.

Example $m = 111101101111$. The 1st, 5th, 6th, 8th and 9th digits start new sequences of 1s or 0s, so $b(m) = 100011011000$.

6 Some Recurrence Relations for b

The following two lemmas give some interesting recurrence relations for b.

Lemma 9 $b(2m+1) = b(2m) + (-1)^m$.

Proof As, for $i > 0$, $(2m)_i = (2m+1)_i$ by Theorem 6 (iii),

$$\begin{aligned} (b(2m))_i &= ((2m)_{i+1} + (2m)_i) \bmod 2 \\ &= ((2m+1)_{i+1} + (2m+1)_i) \bmod 2 \\ &= (b(2m+1))_i. \end{aligned}$$

Hence, by Theorem 6 (i),

$$\begin{aligned} b(2m+1) - b(2m) &= (b(2m+1))_0 - (b(2m))_0 \\ &= (2m+2)_1 - (2m+1)_1 \\ &= (-1)^m, \end{aligned}$$

as $(2m+2)_1 = 1$ and $(2m+1)_1 = 0$ if m is even

and $(2m+2)_1 = 0$ and $(2m+1)_1 = 1$ if m is odd.

Lemma 10 $b(2m) = 2b(m) + \dfrac{1-(-1)^m}{2}$.

Proof By Theorem 6 (iii) and (ii),

$$\begin{aligned} (b(2m))_{i+1} &= ((2m)_{i+2} + (2m)_{i+1}) \bmod 2 \\ &= (m_{i+1} + m_i) \bmod 2 \\ &= (b(m))_i = (2b(m))_{i+1}. \end{aligned}$$

So

$$\begin{aligned} b(2m) &= 2b(m) + (b(2m))_0 \\ &= 2b(m) + \left\lfloor m + \frac{1}{2} \right\rfloor \bmod 2 \\ &= 2b(m) + \frac{1-(-1)^m}{2}. \end{aligned}$$

A number of other recurrence relations can be obtained from these. For example:

$$\begin{aligned} b(8m+2) &= 4b(2m)+3 & \\ b(2^k) &= 3.2^{k-1} & \text{if} \quad k>0 \\ b(2^k+1) &= 3.2^{k-1}+1 & \text{if} \quad k>1. \end{aligned}$$

We can also get a more general expression for $b(m+1)$ in terms of $b(m)$.

Lemma 11 If $m+1 = 2^k(2\ell+1)$ then $b(m+1) - b(m) = 2^k(-1)^\ell$.

Proof By induction on k. If $k=0$ we have the result by Lemma 9.
If $k>0$, we have by Lemmas 9 and 10:

$$\begin{aligned} b(m+1) - b(m) &= 2b\left(\frac{m+1}{2}\right) + \frac{1-(-1)^{\frac{m+1}{2}}}{2} - b(m-1) - (-1)^{\frac{m-1}{2}} \\ &= 2\left(b\left(\frac{m+1}{2}\right) - b\left(\frac{m-1}{2}\right)\right) + \frac{1-(-1)^{\frac{m+1}{2}}}{2} \\ &\quad -(-1)^{\frac{m-1}{2}} - \left(\frac{1-(-1)^{\frac{m-1}{2}}}{2}\right) \\ &= 2^k(-1)^\ell, \end{aligned}$$

by the induction hypothesis.

7 b and Nim Sums

The Nim Sum $m\#k$ of two nonnegative binary integers is the addition of these numbers without carry over. This is used in the study of the game of Nim in Berlekamp, Conway and Guy [1].

i.e. $(m\#k)_i \equiv m_i + k_i \bmod 2$.

This, using $\left\lfloor \frac{m}{2} \right\rfloor_i = m_{i+1}$ and Theorem 6 (iii) proves:

Theorem 12 $b(m) = m\# \left\lfloor \frac{m}{2} \right\rfloor$.

8 Orbits of b

An orbit of b is a set consisting of a number m and its successive images under powers of b, and we are interested in finding the size (or length) of this set for each m, viz. the smallest positive integer k for which $b^k(m) = m$.
First we need two lemmas.

Lemma 13 (i) $\binom{j}{k}$ is odd iff

$$\sum_{i=1}^{\infty} \left\lfloor \frac{j}{2^i} \right\rfloor - \left\lfloor \frac{j-k}{2^i} \right\rfloor - \left\lfloor \frac{k}{2^i} \right\rfloor = 0.$$

(ii) $\binom{j}{k}$ is even for $1 \le k \le p < j$ iff $2^{\lfloor \log_2 p \rfloor + 1} \Big| \, j$.

Proof (i) This follows from the well known result (see for example Griffin [3] Theorem 3.16) that the highest power of 2 to divide $n!$ is $\sum_{i=1}^{\infty} \left\lfloor \frac{n}{2^i} \right\rfloor$.

(ii) Let $u = \lfloor \log_2 p \rfloor + 1$ and $j = 2^{u-1}w + v$, where $0 \le v < 2^{u-1} \le p$. Then if $i < u$,

$$\left\lfloor \frac{j}{2^i} \right\rfloor = 2^{u-1-i}w + \left\lfloor \frac{v}{2^i} \right\rfloor = \left\lfloor \frac{j-v}{2^i} \right\rfloor + \left\lfloor \frac{v}{2^i} \right\rfloor$$

and if $i \ge u$, as $v < 2^{u-1} < 2^i$,

$$\left\lfloor \frac{j}{2^i} \right\rfloor = \left\lfloor \frac{w}{2^{i+1-u}} \right\rfloor = \left\lfloor \frac{j-v}{2^i} \right\rfloor + \left\lfloor \frac{v}{2^i} \right\rfloor.$$

So by (i), $\binom{j}{v}$ is odd.

If $\binom{j}{1}, \binom{j}{2} \ldots, \binom{j}{p}$ are all even, it follows that $v = 0$.

If $w = 2r + 1$ and $v = 0$, $j = 2^u r + 2^{u-1}$ and we can show, exactly as above, that $\binom{j}{2^{u-1}}$ is odd.

Hence as $2^{u-1} \le p$, if $\binom{j}{1}, \binom{j}{2}, \ldots, \binom{j}{p}$ are all even, $v = 0$ and w must be even so that $2^{\lfloor \log_2 p \rfloor + 1} | j$.

If $j = 2^u r$ and $k = 2^{k_1} + 2^{k_2} + \ldots + 2^{k_h}$, where $k_1 > k_2 > \ldots > k_h \ge 0$, $h \ge 1$ and $0 < k \le p$, then $k_1 \le u - 1$,

$$j - k = 2^u(r-1) + 2^{u-1} + \ldots + 2^{k_1+1} + 2^{k_1-1} + \ldots 2^{k_2+1} + \ldots + 2^{k_{h-1}-1} + \ldots + 2^{k_h}$$

and $\left\lfloor \frac{j}{2^u} \right\rfloor - \left\lfloor \frac{j-k}{2^u} \right\rfloor - \left\lfloor \frac{k}{2^u} \right\rfloor = r - (r-1) > 0.$

Hence by (i) $\binom{j}{1}, \binom{j}{2}, \ldots, \binom{j}{p}$ are all even.

Lemma 14 If m, i and j are nonnegative integers, then $(b^j(m))_i = \sum_{k=0}^{j} \binom{j}{k} m_{i+k} \bmod 2$.

Proof By induction on j.

$\underline{j = 0}$ Obvious.

$\underline{j > 0}$ Assume the lemma holds for j, then by Theorem 6 (iii),

$$\begin{aligned}
(b^{j+1}(m))_i &= \sum_{k=0}^{j} \binom{j}{k} (b(m))_{i+k} \bmod 2 \\
&= \sum_{k=0}^{j} \binom{j}{k} (m_{i+k+1} + m_{i+k}) \bmod 2 \\
&= \sum_{k=0}^{j-1} \left(\binom{j}{k} + \binom{j}{k+1} \right) m_{i+k+1} + \binom{j}{0} m_i + \binom{j}{j} m_{i+j+1} \bmod 2 \\
&= \sum_{k=0}^{j-1} \binom{j+1}{k+1} m_{i+k+1} + \binom{j+1}{0} m_i + \binom{j+1}{j+1} m_{i+j+1} \bmod 2 \\
&= \sum_{k=0}^{j+1} \binom{j+1}{k} m_{i+k} \bmod 2.
\end{aligned}$$

Hence, by induction the lemma holds.

Theorem 15 $b^j(m) = m$ iff $m = 0$ or 1 or $2^{\lfloor \log_2 \lfloor \log_2 m \rfloor \rfloor + 1} | j$.

Proof The result holds for $j = 0$, so assume $j > 0$.

By Lemma 14, $b^j(m) = m$ iff for all $i \geq 0$, $\sum_{k=1}^{j} \binom{j}{k} m_{i+k} = 0 \bmod 2$.

If $m = 0$ or 1, this is true for all j. If $m > 1$, for $q = \lfloor \log_2 m \rfloor$, $2^q \leq m < 2^{q+1}$ and $m_q = 1$.

For $i + k > q$, $m_{i+k} = 0$. Hence:

$$b^j(m) = m \quad \text{iff} \quad \text{for } q \geq i \geq 0, \sum_{k=1}^{\min(j, q-i)} \binom{j}{k} m_{i+k} = 0 \bmod 2.$$

If the statement to the right of the iff, which we will call (*), holds, we have, for $i = q - 1$, $q - i = 1 \leq j$ and

$$\binom{j}{1} m_q \equiv \binom{j}{1} \equiv 0 \bmod 2.$$

Now assume $\binom{j}{t} \equiv 0 \bmod 2$, for $1 \leq t < r \leq \min(j, q)$. Then if (*) holds we have, for $i = q - r$,

$$\sum_{k=1}^{r} \binom{j}{k} m_{i+k} = \binom{j}{r} m_q \equiv \binom{j}{r} \equiv 0 \bmod 2.$$

Hence, by induction, if (*) holds,

$$\binom{j}{1}, \binom{j}{2}, \ldots, \binom{j}{\min(j,q)}$$

are all even and as $\binom{j}{j} = 1$, $\binom{j}{\min(j,q)} = \binom{j}{q}$.

If $\binom{j}{1}, \ldots, \binom{j}{q}$ are all even (*) holds.

Hence, by Lemma 13, as $q = \lfloor \log_2 m \rfloor$, $b^j(m) = m$ iff $2^{\lfloor \log_2 \lfloor \log_2 m \rfloor \rfloor + 1} | j$.

Corollary 16 If $2^{2^k} \leq m < 2^{2^{k+1}}$, $b^j(m) = m$ iff $2^{k+1} | j$.

17. For $m > 1$, the length of the orbit of b is $2^{\lfloor \log_2 \lfloor \log_2 m \rfloor \rfloor + 1}$.

9 The Decoding Function $d = b^{-1}$

We define a new function d recursively and then show that this is the inverse of b.

Definition 2 $d(0) = 0$

$$d(2^k + i) = 2^{k+1} - 1 - d(i) \qquad (0 \leq i < 2^k).$$

We now prove lemmas similar to those for b.

Lemma 18 (i) $d(1) = 1$.

(ii) If m_p is the second 1 from the left in $m_k m_{k-1} \ldots m_0$, where $m_k = 1$, then

$$d(m_k m_{k-1} \ldots m_0) = 2^{k+1} - 2^{p+1} + d(m_{p-1} \ldots m_0).$$

(iii) If $m_k = 1$ and $m_{k-1} = m_{k-2} = \ldots = m_0 = 0$, then

$$d(m_k m_{k-1} \ldots m_0) = 2^{k+1} - 1,$$

that is $d(2^k) = 2^{k+1} - 1$ for all $k \geq 0$.

Proof (i) From Definition 2.

(ii) By Definition 2,

$$\begin{aligned} d(2^k + 2^p + m_{p-1}2^{p-1} \ldots + m_0) &= 2^{k+1} - 1 - d(2^p + m_{p-1}2^{p-1} + \ldots + m_0) \\ &= 2^{k+1} - 1 - (2^{p+1} - 1 - d(m_{p-1}2^{p-1} + \ldots + m_0)) \\ &= 2^{k+1} - 2^{p+1} + d(m_{p-1} \ldots m_0). \end{aligned}$$

(iii) $d(2^k) = 2^{k+1} - 1 - d(0) = 2^{k+1} - 1$.

Corollary 19 If $2^p \leq j < 2^{p+1} \leq 2^k$, then $d(2^k + j) = 2^{k+1} - 2^{p+1} + d(j - 2^p)$.

Lemma 20 If $2^k \leq m < 2^{k+1}$ then $2^k \leq d(m) < 2^{k+1}$.

Proof By induction on m.

We can now show that d is the inverse of b.

Theorem 21 $d = b^{-1}$.

Proof (i) We show, by induction on j, that $d(b(j)) = j$.

This is obvious for $j = 0$.

If $j > 0$, we let $j = 2^k + i$ for $0 \leq i < 2^k$.

Then $b(j) = b(2^k - i - 1) + 2^k$ and as, by Lemma 4, $0 \leq b(2^k - i - 1) < 2^k$, we have by the induction hypothesis and Definition 2:

$$\begin{aligned} d(b(j)) &= 2^{k+1} - 1 - d(b(2^k - i - 1)) \\ &= 2^{k+1} - 1 - (2^k - i - 1) \\ &= 2^k + i = j. \end{aligned}$$

(ii) We prove, by induction on j, that $b(d(j)) = j$.

This is obvious for $j = 0$.

If $j > 0$, we let $j = 2^k + i$ for $0 \leq i < 2^k$, then

$$\begin{aligned} d(j) &= 2^{k+1} - 1 - d(i) \\ &= 2^k + (2^k - 1 - d(i)). \end{aligned}$$

As by Lemma 20, $0 \leq 2^k - 1 - d(i) < 2^k$, by Definition 1 and the induction hypothesis:

$$\begin{aligned} b(d(j)) &= b(2^k - 1 - (2^k - 1 - d(i))) + 2^k \\ &= b(d(i)) + 2^k \\ &= i + 2^k = j. \end{aligned}$$

We now write down a lemma for d, similar to Lemma 5 for b.

Lemma 22 (i) If $2^p \leq j < 2^{p+1} \leq 2^k$ then:

(a) $(d(2^k + j))_p = 0$

(b) $(d(2^k + j))_i = 1$ for $p + 1 \leq i \leq k$

(c) $(d(2^k + j))_i = (d(j))_i$ if $0 \leq i < p$.

(ii) $(d(2^k))_i = 1$ if $0 \leq i \leq k$.

Proof By Lemma 18.

Using Lemma 5, we were able to prove Theorem 6 which gave simple methods for finding $(b(m))_i$. The formula for $(d(m))_i$ given below is not quite as simple and its proof does not use Lemma 22.

Theorem 23 $\quad (d(m))_i = \sum_{j=i}^{k} m_j \bmod 2$

where k is the largest value of j for which m_j is non-zero.

Proof By induction on i.

Let the non-zero values of m_i be

$$m_{k_1}, m_{p_1}, m_{k_2}, m_{p_2}, \ldots, m_{p_r} \quad (\text{and} \quad m_{k_{r+1}})$$

where $k = k_1 > p_1 > k_2 \ldots > p_r$ $(> k_{r+1})$.

$\underline{i = 0}$ If there is an even number of these non-zero m_is then by Lemma 18:

$$d(m) = 2^{k_1+1} - 2^{p_1+1} + 2^{k_2+1} - 2^{p_2+1} + \ldots 2^{k_r+1} - 2^{p_r+1}$$

so $\quad (d(m))_0 = 0 = \sum_{j=0}^{k} m_j \bmod 2.$

If this number is odd

$$d(m) = 2^{k_i+1} - 2^{p_i+1} + \ldots 2^{k_r+1} - 2^{p_r+1} + 2^{k_{r+1}} - 1$$

so $\quad (d(m))_0 = 1 = \sum_{j=0}^{k} m_j \bmod 2.$

$\underline{i > 0}$ By Theorem 6 (iii) and Theorem 21:

$$m_{i-1} = (d(m))_i + (d(m))_{i-1} \bmod 2.$$

That is, using the induction hypothesis:

$$\begin{aligned} (d(m))_i &= (d(m))_{i-1} + m_{i-1} \bmod 2 \\ &= \sum_{j=i-1}^{k} m_j + (m)_{i-1} \bmod 2 \\ &= \sum_{j=i}^{k} m_j \bmod 2. \end{aligned}$$

Corollary 24 $\quad (d(m))_i = 0$ if there is an even number of 1s to the left of m_{i-1} in the binary representation of m, and $d(m)_i = 1$ otherwise.

From this we have two forms of an algorithm to evaluate $d(m)$.

Algorithm 1 **for** d

For each digit in the binary representation of m, put a 0 if there is an even number of 1s from this digit (including it) to the left and a 1 otherwise.

Algorithm 2 **for** d

To form $d(m)$ from the binary representation of m replace the 1st, 3rd, 5th, etc. occurrences of 1 and any subsequent 0s by 1s and replace the 2nd, 4th, etc. occurrences of 1 and any subsequent 0s by 0s.

Example $\quad d(1100010110001) = 1000011011110.$

References

[1] Berlekamp, E.R., Conway, J.H. and Guy, R.K., "Winning Ways - for your mathematical plays", Vol.1: Games in General, Academic Press, London, 1982.

[2] Conder, Marston, "Explicit definition of the binary reflected Gray Code", Discrete Mathematics, Vol.195, (1999), pp245-249.

[3] Griffin, H., "Elementary Theory of Numbers", McGraw-Hill, New York, 1954.

[4] Nashelsky, L,. "Introduction to Digit Computer Technology", John Wiley, 1982.

[5] Sharma, Bhu Dev and Khanna, Ravindar Kumar, "Integer characterization of binary and m-org Gray Codes", Journal of Combinatorics, Information and System Sciences, Vol.4, (1979), pp227-236.