INFORMATION AND CONTROL **63**, 1–10 (1984)

An Approach for Generating Different Types of Gray Codes

BENJAMIN ARAZI

Department of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, Louisiana 70803

Given a certain Gray code consisting of 2^n codewords, it is possible to generate from it $n!\,2^n$ codes by permuting and/or complementing the bits in all the codewords in the same manner. The codes obtained this way are all defined to be of the same **type**. An approach for converting the standard Gray code (known as the reflected code) into other Gray codes of different types is presented in this paper. A systematic way of generating, for example, **all** types of Gray codes consisting of 16 codewords is given.

1. INTRODUCTION

A Gray code of order n is an ordered set of 2^n different binary n-tuples which differ one from the next (including the first and last) in only one bit position. The customary standard Gray code is the "reflected" code in which the code is generated recursively by the process demonstrated in Fig. 1. In Fig. 1(b) the code of order n is written twice, once below the other, where the bottom part is a reflection of the top part. A bit of value 0 is then attached at the left of every word in the top part, where a bit of value 1 is attached to the words at the bottom. The validity of this construction is self explanatory.

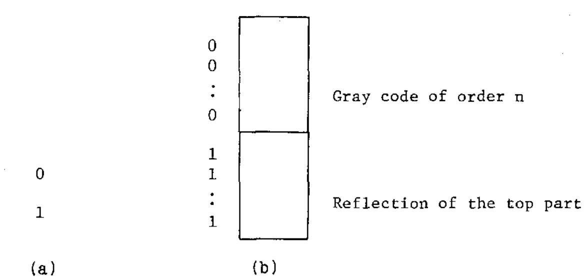

FIG. 1. A recursive way of generating the reflected Gray code: (a) lists the Gray code of order 1; (b) shows how a Gray code of order $n+1$ is generated form the Gray code of order n.

0019-9958/84 $3.00

A direct method of constructing the reflected Gray code of order n without iterations is the following. Let $(b_{n-1}, b_{n-2},..., b_0)$ be the binary representation of an integer $j \in [0, 2^n - 1]$, where b_{n-1} is the most significant bit. Let $(g_{n-1}, g_{n-2},..., g_0)$ be the n-tuple located at the j place in a reflected Gray code. Then $g_{n-1} = b_{n-1}$; $g_i = b_{i+1} \oplus b_i$, $i = n-2$, $n-3,..., 0$, where $\oplus$ denotes addition modulo 2.

Notation. Given a Gray code, N_i denotes the number of times that a pair of successive codewords in the code differ in bit position no. i, $0 \leqslant i \leqslant n-1$.

Arranging the codewords of a Gray code of order n, one below the other, yields a structure or n columns each consisting of 2^n bits. The bits is a column are arranged in "runs" of 1's and 0's. (If a column starts and ends with runs consisting of bits of the same value, then in view of the cyclic structure of the code the two runs are considered as one.) Clearly the number of runs in column i is N_i. (The right column is 0.)

It should be obsrved that in any Gray code, $\sum_{i=0}^{n-1} N_i = 2^n$. In the reflected Gray code of order n, $N_i = 2^{n-1-i}$ for $i = 0, 1,..., n-2$, and $N_{n-1} = 2$. It was shown in Robinson and Cohn (1981) that there always exists a Gray code, for any order $n \geqslant 4$ such that $|N_i - N_j| \leqslant 2$ for any two bit positions i and j. Such a code is defined as a "balanced Gray code." It should be noted that N_i is always even. This means that in a balanced code either $|N_i - N_j| = 2$ or $N_i = N_j$. For the case where 2^n is divisible by n, $N_i = 2^n/n$ for $i = 0, 1,..., n-1$. A way of constructing recursively balanced Gray codes was also presented in Robinson and Cohn (1981).

Given a Gray code, it is possible to generate from it $n!\, 2^n$ codes by permuting the bits in the codewords and/or complementing the bits in the codewords in the same manner for all the codewords. Two Gray codes are defined to be of the same "type" Gilbert (1958) if they are obtained one form the other by the described permutation and/or complementation operations.

It should be noted that a Gray code represents a certain Hamiltonian path on an n-dimensional cube and the described permutation and complementation operations correspond to simple symmetric operations on the cube, under which the path does not change basically.

It is shown in this paper how the reflected Gray code can be converted into another Gray code of that order, having other (valid) values of N_is. This includes the balanced code as a special case. It is then shown how some different types of Gray codes can be generated. These include all the existing types of codes of order 4.

2. Some Basic Operations Performed on a Reflected Gray Code

2.1. *Removal and Insertion of Pairs of Codewords*

The following property of a reflected Gray code of order n follows directly from the way by which it is constructed. Let $y \bmod 2^{i+1} \equiv 2^i - 1$ for a certain $y \in [0, 2^n - 1]$ and a certain $i \in [0, n-2]$. Then the codewords with indices y and $y + 2^{i+1} + 1$ differ in only one bit position. (The addition of $2^{i+1} + 1$ to y is always made mod 2^n). This property is clarified by observing Fig. 2(a) which lists the reflected Gray code of order 4. For $i = 1$, for example, 1, 5, 9, and 13 mod 4 all equal 1 and it is observed that the codewords with these indices differ in one bit position from those which appear 5 places later.

It follows from the described property that if a group of 2^{i+1} successive codewords starting with index z for $z \bmod 2^{i+1} \equiv 2^i$, is removed from the code, then the remaining code (which consists of $2^n - 2^{i+1}$ words) still has the basic property of the code. (Namely, two successive words differ in only one bit position.) Note that for $i = 0$, the described property means that removing any pair of successive codewords, starting with an odd index, leaves the basic property of the code unchanged.

Property 1. Let $i \in [0, n-2]$. By removing a group of 2^{i+1} successive codewords from a reflected Gray code of order n, starting with index z, where $z \bmod 2^{i+1} \equiv 2^i$, the value of N_k is decreased by 2^{i-k} for $k = 0, 1,..., i-1$. The value of N_i is decreased by 2 and the values of N_k, $k = i+1, i+2,..., n-1$ remain unchanged.

Property 1 is clarified by observing the reflected Gray code of Fig. 2a. Choose $i = 2$, for example, in which case a group of 8 successive codewords can be removed from the code starting with word 4. The value of N_0 is then decreased by 4 and the values of N_1 and N_2 are decreased by 2.

Let x be the index of a certain codeword B in a reflected Gray code of order n. The n codewords which differ from B in only one bit position are the ones with indices $x + 2^i - 1 - 2(x \bmod 2^i)$, $i = 1, 2,..., n$. For example, take $n = 4$ and $x = 13$. Observing Fig. 2a, the four codewords which differ form $B = (1011)$ in only one bit position have indices 12, 14, 10, 2.

The location of the described n codewords can be defined verbally in the following way. Partition the codewords of the code into 2^j successive groups. The word which is the reflection of B, across the center of the group which contains B, differs from B in only one bit position.

This applies to $j = 0, 1,..., n-1$. The described property follows, of course, directly from the construction described in Fig. 1.

Definition. Let x be the index of a codeword B of a reflected Gray

0 0000	0 0000	0 0000	0 0000	0 0000	0 0000	0 0000
1 0001	1 0001	3 0010	1 0001	7 0100	1 0001	1 0001
2 0011	2 0011	12 1010	2 0011	6 0101	14 1001	2 0011
3 0010	3 0010	11 1110	3 0010	1 0001	9 1101	13 1011
4 0110	12 1010	4 0110	4 0110	2 0011	10 1111	14 1001
5 0111	11 1110	5 0111	5 0111	3 0010	13 1011	9 1101
6 0101	4 0110	2 0011	6 0101	4 0110	2 0011	10 1111
7 0100	5 0111	1 0001	7 0100	5 0111	3 0010	11 1110
8 1100	6 0101	6 0101	8 1100	10 1111	4 0110	12 1010
9 1101	7 0100	7 0100	11 1110	11 1110	5 0111	3 0010
10 1111	8 1100	8 1100	10 1111	8 1100	6 0101	4 0110
11 1110	9 1101	9 1101	9 1101	9 1101	7 0100	5 0111
12 1010	10 1111	10 1111	14 1001	14 1001	8 1100	6 0101
13 1011	13 1011	13 1011	13 1011	13 1011	11 1110	7 0100
14 1001	14 1001	14 1001	12 1010	12 1010	12 1010	8 1100
15 1000	15 1000	15 1000	15 1000	15 1000	15 1000	15 1000
(a)	(b)	(c)	(d)	(e)	(f)	(g)

FIG. 2. Various orderings of the codewords of a reflected Gray code of order 4. All the indexing refer to the order listed in (a).

code of order n. **The reflection of order i of B**, $i = 0, 1, \ldots, n-1$, is the codeword whose index is $x + 2^{i+1} - 1 - 2(x \bmod 2^{i+1})$.

Property 2. Let B and D be two successive codewords of a reflected Gray code. (The index of D is higher.) Let E and F be the respective reflections or order i of B and D. For the case where EF are successive, and are not BD themselves, let BD be repeated in the code by inserting them again between EF in the order $FDBE$. Then any two successive codewords in the resultant extended code still differ in only one bit position, where the value of N_i was increased by 2, and the rest of the N_j's have their values unchanged.

Proof. B and D differ in the kth bit, for some $0 \leqslant k \leqslant n-1$, and only there. Observe first that B and D are one the reflection of order k of the other. It follows that if EF are the reflections of order i of BD, and BD can be inserted between EF, then $i \neq k$. Both B and D then have the same bit x in location i, where E and F have the bit $\bar{x}$ in that location. The insertion of BD between EF will then add a run of two bits of value x between two bits of value $\bar{x}$ in column i of the code. (The concept of "columns" was introduced before, following the definition of N_i.) No changes in the values of other N_j's are caused by the described operation since the bits of EB and FD are equal in any location j for $j \neq i$. Q.E.D.

Based on Properties 1 and 2, **a reflected Gray code can be converted into a Gray code of another type by applying a removal-insertion procedure.**

Some basic operations performed on the reflected Gray code, and which will result in a Gray code, are treated next. Their validity is based on the above observation. The reason for defining these specific operations is clarified in Section 3.

Basic operation 1. *Removing and inserting a single pair of codewords with odd–even indices.* Removing a single pair of codewords from a reflected Gray code, with the index of the first word being odd, and inserting this pair between its reflection of any order, will yield another Gray code. As an example observe the reflected Gray code of order 4 listed in Fig. 2a. The codewords $B = 1110$ and $D = 1010$ with respective indices 11 and 12, can be removed from the code and inserted between codewords $F = 0010$ and $E = 0110$ (with indices 3 and 4), which are the respective reflections of order 3 of B and D. The obtained code is listed in Fig. 2b, with $N_0 = 6$, $N_1 = 4$, $N_2 = 2$, $N_3 = 4$.

Basic opration 2. *Removing and inserting two pairs of codewords, each with odd–even indices.* Two independent basic operations 1 are performed here, where the index of the first codeword out of every removed pair is odd. As an example consider the balanced Gray code listed in Fig. 2c. It was obtained by inserting codewords 1, 2 and 11, 12 of the reflected code between codewords 5, 6 and 3, 4, respectively, which are their respective reflections of order 2 and 3. The insertions thus increased the values of N_2 and N_3 by 2 (from 2 to 4) and decreased N_0 by 4. The value of N_1 which was 4 initially, was left unchanged.

2.2. *Removing Pairs of Codewords with Even–Odd Indices*

Removal of a pair of successive codewords from a reflected Gray code will leave the basic property of the code unchanged (i.e., any two adjacent codewords in the remained code still differ in only one bit position) iff the index of the first word out of the removed pair is odd. The codes depicted in Figs. 2b and c were constructed by a removal and insertion of pairs with such indices. Removal of a pair with even–odd index can leave the basic property of the code unchanged only if it is accompanied by another operation.

Basic operation 3. *Removing and inserting two adjacent pairs of codewords, each with even–odd indices.* Removal of two adjacent pairs of codewords, the first one of which having an even index will not change the basic property of the code (see Property 1). The two pairs can then each be inserted between their reflections of any order. As an example consider the code of Fig. 2d which was constructed by removing from the reflected code the two pairs of codewords with indices 10-11 and 12-13, and inserting them, respectively, between codewords 8-9 and 14-15.

Basic operation 4. *Removing and inserting two pairs of codewords which are* 4 *places apart, each with even–odd indices.* Consider the case where the two pairs of codewords 6-7 and 12-13 are removed from the reflected code of Fig. 2a. This will leave codewords 5 and 8 being adjacent, as well as codewords 11 and 14. Codewords 5, 8 and 11, 14 do not differ in only one bit position. In order to get back to the basic property of a Gray code (i.e., a contiguous arrangement of codewords which differ in only one bit position), one can switch the location of the pairs 8, 9 and 10, 11 resulting the following order: 0, 1, 2, 3, 4, 5, 10, 11, 8, 9, 14, 15. The originally removed two pairs can then be inserted between their reflections of any order. The code of Fig. 2e was obtained by inserting codewords 6-7 between 0, 1 and 12-13 between 14, 15. The described operation is valid for the cases where the indices of the initially removed pairs are of the form $x, x+1$ and $x+4, x+5 \pmod{2^n}$ for $x \bmod 4 \equiv 2$. It is not valid for $x \bmod 4 \equiv 0$.

2.3. *The Chaining Principle*

When converting a reflected Gray code into another Gray code using the described process of decreasing the values of some N_i's and increasing the values of others, consider the case where a pair EF of successive codewords in the reflected code, is the reflection of order i of a pair BD and the reflection of order j, $i \neq j$, or another pair HJ. It would be impossible to insert both BD and JH between E and F, for the purpose of increasing N_i and N_j by 2 while decreasing N_0 by 4. However, after installing HJ between E and F, the effect of further decreasing N_0 by 2, while increasing N_i by that amount, can still be achieved by removing the whole group $EJHF$ and inserting it between B and D. A "chaining" operation is then observed here, where after the insertion of HJ in a new location, it is moved again together with its neighbors.

EXAMPLE. Referring to the reflected Gray code listed in Fig. 2a, let $B = 0001$, $D = 0011$, $E = 1001$, $F = 1011$, $H = 1101$, $J = 1111$, with respective indices 1, 2, 14, 13, 9, 10. EF is a reflection of order 3 of BD and a reflection of order 2 of HJ. The removal of HJ and its insertion between E and F reduces N_0 from 8 to 6 and increases N_2 from 2 to 4. If the group $EHJF$ is now removed and inserted between B and D, the value of N_0 is reduced from 6 to 4 and that of N_3 is increased from 2 to 4, resulting in the balanced code of Fig. 2f.

2.4. *Inserting a Cyclic Shift of a Group of Codewords between a Pair of Codewords*

Let B be the reflection of order i, for some $0 \leqslant i \leqslant n-1$, of a codeword D, in a reflected Gray code of order n. The removal of all the codewords between B and D will leave a code in which any pair of adjacent codewords still differs in only one bit position. The first and last codewords in the removed group of codewords also differ in one bit position. If the removed group consists of N codewords, then shifting it cyclically between 1 and $N-1$ places, yields a group of codewords each one differing from its neighbors one bit position, where the first and last codewords were neighbors in the original reflected code. The obtained cyclically shifted group can therefore be inserted, as a whole, between any reflection of the codewords which presently form its edges. (As long as these reflections are not inside that same group.)

EXAMPLE. The balanced code of Fig. 2g was constructed as follows. A group of 6 successive codewords, starting with 9, was shifted cyclically such that codewords 12, 13 form the edges of the obtained group. The obtained group was then inserted between codewords 2 and 3 which are the reflections of order 3 of 13, 12.

The above example clarified the fact that shifting cyclically a group of codewords, and then inserting them between two codewords, enables reducing the value of some N_i by 4 and increasing the values of other N_i's. There is no way of reducing the value of some N_i by more than 4, using single execution of the described operation. This statement is apparent by observing the effect of the operation in the example on the structure of column 0 of Fig. 2a.

3. CONSTRUCTING DIFFERENT TYPES OF GRAY CODES

As was already mentioned before, a Gray code of order n can be analyzed by considering the arrangement of the n vertical binary "columns" of length 2^n. Two Gray codes are of the same type iff the columns in one code are obtained form those of the other by permuting them and/or complementing any subset of them. It should be noted that complementing a column does not change the number of the runs in it or their distribution. Any of the following three conditions is therefore sufficient for two Gray codes of order n to be of **different** types.

(1) The set $\{N_i\}$, $i=0, 1,..., n-1$ of one code contains at least one value which does not appear in the corresponding set of the other code.

(2) The sets $\{N_i\}$ of the two codes contain the same values (the order of these values is irrelevant). However, at least one column in one code contains a run of a length which cannot be found in any column having the same number of runs in the other code.

(3) For any column in one code there is a column in the other code with the same number of runs, and of the same length. However, at least one column in one code has distribution of runs which cannot be found in the other code. (The two vectors 1111000011001100 and 1111001100001100 have the same number of runs and of the same length, but different distributions.)

Based on the above observation it follows that the five Gray codes of Figs. 2a, b, c, d, and e are all of different types. Note first that the codes of Figs. 2a, b, c, and d have N_i values $\{8, 4, 2, 2\}$, $\{6, 4, 4, 2\}$, $\{4, 4, 4, 4\}$, and $\{6, 6, 2, 2\}$, respectively, and are therefore of different types. The codes of Figs. 2b and c both have N_i values $\{6, 4, 4, 2\}$. However, they are different types since the column with 6 runs has runs of length 3 only in the second code.

The above approach for differentiating between Gray codes of different types **also offers a way of how to actually generate them.**

There are in total nine different types of Gray code of order 4. (See [2]). The four types of codes depicted in Figs. 2b, c, d, and e were obtained from the reflected code (which forms a fifth type) by operating on it with one of the four basic operations, described in Section 2. The rest of the four types of codes can also be obtained from the reflected code by operating on it with these basic operations. The four additional codes are depicted in Fig. 3. The code of Fig. 3a is obtained by inserting codewords 1-2 of the reflected code, between codewords 5, 6. (Basic operation 1.) The code of Fig. 3b is obtained by inserting codewords 1-2 of the reflected code between 5, 6 and inserting 9-10 between 13, 14. (Basic operation 2.) The code of Fig. 3c is obtained from the reflected code by splitting the group of codewords 6, 7, 8, 9 into two pairs. Codewords 6-7 were then inserted between 0, 1, codewords 8-9 between 10, 11. (Basic operation 3.) The code of Fig. 3d is obtained from the reflected code by first removing codewords 6-7 and 12-13 from the code. The pairs of codewords 8-9 and 10-11 are then interchanged. Codewords 6-7 is then inserted between 4, 5, and 12-13 between 14, 15. (Basic operation 4.) These codes are of new types due to the following observation. The codes of Figs. 2d and 3d are the only ones with N_i values $\{6, 6, 2, 2\}$. However, the six runs in both columns 0 and 1 in the code of Fig. 2d are of length 3, 2, 2, 2, 3, 4, whereas the corresponding runs in Fig. 3d are of length 3, 2, 3, 3, 2, 3. Note that this distribution of runs not only indicates that the two codes are of different types. What is equally important is the fact that the way by which these

0	0000	0	0000	0	0000	0	0000
3	0010	3	0010	7	0100	1	0001
4	0110	4	0110	6	0101	2	0011
5	0111	5	0111	1	0001	3	0010
2	0011	2	0011	2	0011	4	0110
1	0001	1	0001	3	0010	7	0100
6	0101	6	0101	4	0110	6	0101
7	0100	7	0100	5	0111	5	0111
8	1100	8	1100	10	1111	10	1111
9	1101	11	1110	9	1101	11	1110
10	1111	12	1010	8	1100	8	1100
11	1110	13	1011	11	1110	9	1101
12	1010	10	1111	12	1010	14	1001
13	1011	9	1101	13	1011	13	1011
14	1001	14	1001	14	1001	12	1010
15	1000	15	1000	15	1000	15	1000
(a)		(b)		(c)		(d)	

FIG. 3. Four additional types of Gray code of order 4. The indexing refers to the order defined by the reflected code.

codes were constructed exhibits guidelines on how to actually use the four basic operations of Section 2 for generating different types of codes. Similar considerations show why the codes of Figs. 2b, e, 3a, b, and c are all of different types, although they all have N_i values $\{6, 4, 4, 2\}$.

The significance of the above discussion lies in the face that **all** the types of Gray code of order 4 can be derived by using the four basic operations of Section 2. We conclude by stating the following general property.

Property 3. Let C_i denote a Gray code of order n obtained from the reflected code of that order by removing a pair of codewords with odd–even indices and inserting them between their reflections of order i. (Basic operation 1.) Then C_i and C_k are of different types for $i \neq j$.

Proof. The removal of a pair of codewords with odd–even indices creates a run of length 4 in column 0. Inserting this pair between its reflection of order i creates a further run of length 4 in this column. The distance between the described two runs of length 4 then differs when considering the codes C_i and C_j for $i \neq j$. The rest of the runs in column 0 in both codes are of length 2, where this column is the only one having in total $2^{n-1}-2$

runs. Since the distribution of the runs in this column is different, C_i and C_j are of different types. Q.E.D.

Property 3 lays the foundations for a general approach in controlling the distribution of runs in various columns of generated codes, thus generating different types. The details are mostly self-explanatory and can be clarified by observing, for example, the distribution of the runs in column 0 in the code of Fig. 2d as compared to that of Fig. 3c, and Fig. 2e as compared to Fig. 3d.

4. Further Research Directions

The principles presented in this paper open several possibilities for a further research. The first question is whether it is possible to characterize some or all the types of a Gray code of a certain order, by indexing them according to the method by which they are obtained from the reflected code of that order. This could also give a way of evaluating how many types of a Gray code of a certain order are there.

Another problem is devising a scheme which enables generating **sequentially** the codewords of Gray codes of different types. An interesting practical case would be to generate sequentially the codewords of a balanced Gray code thus having a counter in which the changes from one count to the next are shared equally among the various bit locations with the number of total changes being the minimum possible. The reflected Gray code has the important property that a simple direct relation exists between the indices of the codewords and their internal structure, which enables generating its codewords sequentially. The methods presented in this paper for converting a reflected Gray code into codes of other types may therefore present a potential tool for a sequential generation of the codewords of these codes.

Received: June 27, 1983; accepted: December 5, 1984

References

1. Robinson, J. P., and Cohn, M. (1981), Counting sequences, *IEEE Trans. Comput.* **C-30**, 17–23.
2. Gilbert, E. N. (1958), Gray codes and paths on the n-cube, *Bell System Tech. J.* **37**, 815–126.