RC23410 (W0411-034) November 3, 2004
Electrical Engineering

IBM Research Report

DC-Balanced 6B/8B-P Transmission Code with Local Parity

Albert X. Widmer
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

March 2, 2004

DC-Balanced 6B/8B-P Transmission Code with Local Parity

Albert X. Widmer

IBM T.J. Watson Research Center
1101 Kitchawan Rd. Route 134, Yorktown Heights, NY 10598-0218
914-945-2047, widmer@us.ibm.com

Abstract: A transmission code which packs six bits of data and four control vectors into an eight-bit format is presented. All 68 valid 8-bit vectors are dc-balanced with a maximum digital sum variation of 6 and a maximum run length of 6. Any single error can be instantly detected and attributed to a particular 8-bit vector. So a parity vector computed over a block of vectors can be used to correct the error. The circuit implementation requires no more than 69 standard primitive logic cells for encoding and 78 cells for the combined operations of decoding and validity checks. There are at most 5 primitive logic gates in any logical path.

Introduction

Since the start of the digital age, it has been common practice to append a parity bit to a group of bits such as a byte so a byte afflicted with a single error could be identified and perhaps corrected by another set of parity bits. For reliable serial transmission, redundancy is often added to control the run length and bandwidth characteristics of the serial bit stream. While transmission codes usually can detect many types of errors in a string of coded vectors, they usually cannot always point to the exact error location or identify the specific faulty vector; extra redundancy is required to do so. References 2, 3, and 4 are examples of this approach. The overall coding efficiency can be raised if parity and transmission aspects are solved by a single solution as was done in reference 1 for a dc-balanced 8B/10B code. However, the encoding and decoding circuitry for reference 1 is complex and difficult to implement for very fast serial links while preserving low latency as required for computer bus applications. For new applications of transmission codes in wide computer busses, compatibility with the 8-bit byte format carries less weight. Remainders of a few bits may be allotted to spare assignments or are readily handled by compatible codes such as 1B/2B, 3B/4B, or 5B6B, or any combination thereof. In other situations, where the bus width n is a multiple of 6 and 8 such as nx24, it is just the number of coding circuits and perhaps transmission lanes which changes.

A new solution with the 6B/8B-P code is presented here. The code is slightly less efficient than reference 1 but for applications which require error correction, it can be more efficient than solutions according to reference 2, 3, or 4 depending on the particular configuration. It is implemented with very simple circuits suitable for extremely high operating rates. Short circuit delays are compatible with low latency requirements. Also, the ratio of the serial transmission rate and the parallel electrical interface clocks is a preferred power of two versus a multiple of 3 or 5 for solutions based on any of the above references. The simple circuitry also helps to contain power dissipation in a critical area.

While the new code is primarily aimed at applications with statistically independent single errors, such as well designed optical links, it can also have advantages for applications with no forward error correction where the local parity feature has a subordinated role.

Additionally, the 6B/8B code provides another design point among several alternatives. As an example, an electrical bus with 72 data lines may be transmitted over nine 8B/10B coded lines. If the distance and baud rate of the electrical lines is aggressive, Decision Feedback Equalizers may be required which have a tendency to generate multiple errors. To overcome this problem, five lines are added carrying an error correction Hamming code. To transport a single 72-bit word over the 14 high speed lines requires then 14x10 = 140 bits. Using 6B/8B code over 12+5 = 17 high speed lines requires only 17x8 = 136 bits which is surprising considering the larger overhead of 6B/8B code. The savings result from less overhead for error correction, because of the wider correction entities. The larger number of serial lines can be used to either lower the serial transmission rate to 8 times the bus rate rather than 10 times. Alternatively, the bus rate and throughput can be increased by 25% assuming in both cases an entire 72-bit word is dispatched with each bus-rate clock cycle. For comparison, similar performance can be obtained using the more complex 7B/8B code which can transmit words of 77 bits with Hamming correction on just 16 lines operating at eight times the bus rate. The well known 5B/6B code can handle 75-bit words with Hamming correction on 20 lines at a serial rate of just six times the word rate.

General Description and Novelty of the 6B/8B-P Code

The input to the encoding apparatus consists of seven lines plus a clock. Six unrestricted lines represent 64 data vectors if the seventh line, the control line, is not asserted. If the control line is asserted together with one of four specified data vectors, a coded control vector is generated which is recognizable as other than data. So there are a total of 68 coded vectors and they are all balanced. Therefore, any single bit error or any odd number of bit errors in the coded domain will generate an invalid vector instantly recognizable as such.

For purposes of encoding and decoding, the 64 source vectors are classified into four sets:

1. A first set of 20 source vectors comprises all balanced 6B vectors.
2. A second set of 14 source vectors comprises all 6B vectors with a disparity of plus two with the exception of the vector with a trailing run of four ones.
3. A third set of 14 source vectors comprises all 6B vectors with a disparity of minus two with the exception of the vector with a trailing run of four zeros.

4. A fourth set of 20 source vectors comprises the 14 vectors with a disparity of four or six, the two vectors with a disparity of two and a trailing run of four, and the four control vectors.

In the encoding process, all four sets obtain a two-bit prefix as described in more detail below. Alternatively, the two bits could also be added as a suffix or at other specified positions. However, the prefix is the preferred implementation for reasons explained below. The source bits of the first three sets remain unchanged for encoding and decoding. Only the 16 data vectors of set four require changes in two or three bit positions to generate balanced coded vectors. The prefix is selected as follows:

1. The first set takes a two-bit prefix with complementary bit values, i.e. bit values of 10, or 01 in an alternate implementation.
2. The second set takes a two-bit prefix with bit values of 00.
3. The third set takes a two-bit prefix with bit values of 11.
4. The fourth set takes a two-bit prefix which is the complement of that of the first set, i.e. bit values of 01, or 10 in an alternate implementation.

Notation

The six bits of the source vectors are identified by the capital letters A, B, C, D, E, and F. An additional control input carries the label K. The eight bits of the coded vectors are identified by the respective lower case letters a, b, c, d, e, and f; the two extra bits are identified by the letters g and h. In the circuit diagrams described below in reference to FIGS. 7A/B and 8A/B, the coded inputs and outputs ‘a’ through ‘h’ are prefixed with the capital letter C because some chip design tools do not differentiate between lower and upper case letters. All vectors are assigned a name starting with the letter D or K for data or control vectors, respectively, followed by a two position octal number for source vectors, or a three position octal number for coded vectors. The octal number represents the binary bit pattern with the low order bit on the right side (bit A or a). The high order octal position for coded vectors indicates the value of the bits g and h which identify the class to which the vector belongs.

This document assumes, that the high order bit h is transmitted first. The code is not sensitive to the order, but because the bits g and h are used to classify the coded vectors, it is conceivable that their position at the leading end of coded vectors could be used to slightly reduce the latency of the receiver or to improve the timing margin. Note that a reversal of the transmission order requires reversals in the definition of the synchronizing vector pair described below.

The signal names used in the equations of this document do not reflect any logic levels, they are to be interpreted as abstract logic statements. However, in the circuit diagrams, the signal names may be prefixed with the letter P or N to indicate whether the function is true at the upper or lower level, respectively. The P and N prefixes are normally not used for net names which start with P and N, respectively. Net numbers starting with ‘n’ or ‘m’ are true at the lower level and take the P prefix if true at the upper level. In the logic equations, the symbols •, +, and ⊕ represent the Boolean AND, OR, and EXCLUSIVE OR functions, respectively. The apostrophe (’) represents negation.

Source Vectors and Coded Vectors

FIG. 1 is a trellis diagram of the 64 source vectors, ignoring the value of the K-input. FIG. 2 is a trellis diagram of the 68 coded vectors which include four control vectors which are other than data. The numbers in the diagrams indicate the number of vectors ending with the node to the left.

Low Frequency Characteristics

From the trellis of FIG. 2 it is evident, that the code is dc-balanced with a maximum digital sum variation of six. The normalized dc-offset which is related to the area between the zero disparity level and the extreme contour of the trellis is 1.75. As a point of reference, the offset value for the Fibre Channel 8B/10B code of Ref. 5 is 1.9. The low frequency cut-off point for high pass filters can be located as low or below that for 8B/10B code depending on the low pass filter parameters for equal eye closure (Ref. 6).

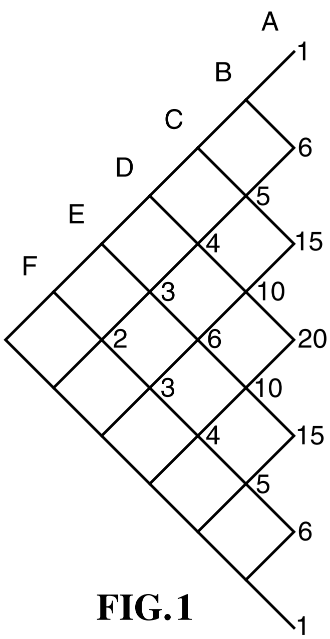


FIG.1

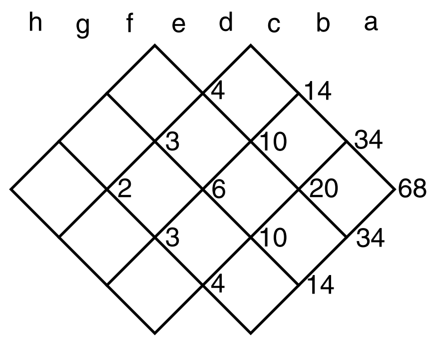


FIG.2

Synchronization Characteristics

The maximum run length is six centered across the 8B boundaries. There are no contiguous runs of six. The run of six is singular, i.e. it cannot appear with any other alignment with reference to the 8B boundaries and can serve as the comma.

To generate the comma of six zeros in the context of a control character, one of the control characters (K170) is defined with a trailing run of three zeros. This character may be followed by any of the four data characters (D027, D033, D035, D036) from the set of FIG. 5 with a leading run of three zeros. An equivalent, alternate comma of six ones is generated from another control character (K107) with a trailing run of three ones followed by any of the four data characters (D341, D342, D344, D350) from the set of FIG. 6 with a leading run of three ones. These two vector pairs allow the signalling and checking of both the vector alignment and the start or the end of a frame.

In normal data traffic, there are also sequences of six ones or zeros with identical alignment which can also be used for alignment or alignment checks. A third possibility to gain alignment with a random sequence of coded vectors is to monitor the validity of the received coded vectors or the running disparity at 6-baud intervals and stepping the alignment until no invalid characters appear or until the running disparity value at the boundaries assumes a steady value which then can be assumed to be zero and should remain there in the absence of errors.

6B/8B Encoding Table

As described in the General Description above, for purposes of this design, the sixty-eight coded 8B vectors of FIG. 2 are divided into four sets as illustrated by the trellis diagrams of FIG. 3, 4, 5, and 6. The solid lines in FIG. 3, 5, and 6 represent 48 of the source vectors which remain unchanged in the encoded domain except for the two-bit dotted prefix. The trellis of FIG. 4 represents the four control vectors and the remaining 16 data vectors.

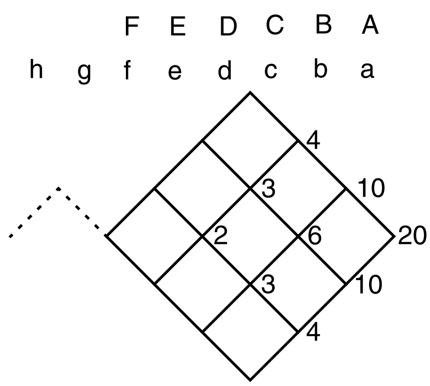


FIG.3

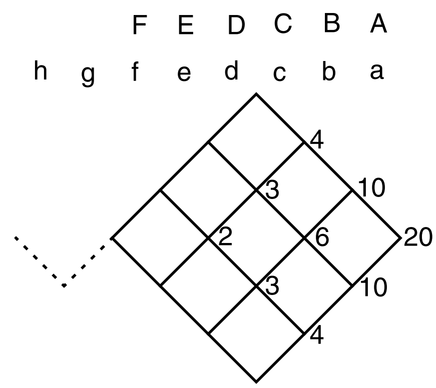


FIG.4

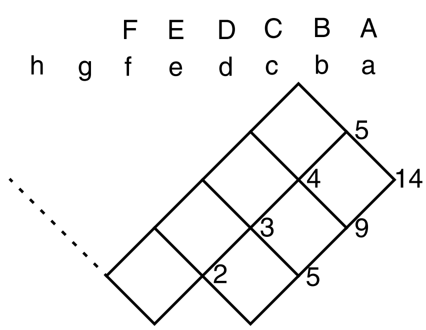


FIG.5

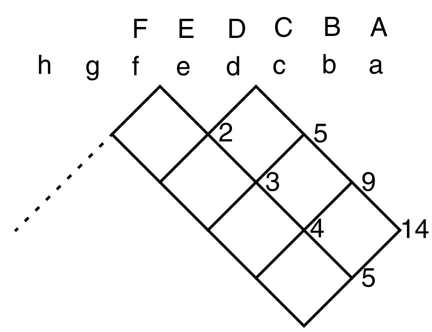


FIG.6

The 48 coded 8-bit vectors of FIG. 3, 5, and 6 which require no changes for encoding are listed in Table 1.

Table 1. 48 Encoded Vectors with no Changes

FIG. 3		FIG. 5		FIG. 6	
Name	**h g f e d c b a**	**Name**	**h g f e d c b a**	**Name**	**h g f e d c b a**
D207	1 0 0 0 0 1 1 1	D027	0 0 0 1 0 1 1 1	D303	1 1 0 0 0 0 1 1
D213	1 0 0 0 1 0 1 1	D033	0 0 0 1 1 0 1 1	D305	1 1 0 0 0 1 0 1
D215	1 0 0 0 1 1 0 1	D035	0 0 0 1 1 1 0 1	D306	1 1 0 0 0 1 1 0
D216	1 0 0 0 1 1 1 0	D036	0 0 0 1 1 1 1 0	D311	1 1 0 0 1 0 0 1
D223	1 0 0 1 0 0 1 1	D047	0 0 1 0 0 1 1 1	D312	1 1 0 0 1 0 1 0
D225	1 0 0 1 0 1 0 1	D053	0 0 1 0 1 0 1 1	D314	1 1 0 0 1 1 0 0
D226	1 0 0 1 0 1 1 0	D055	0 0 1 0 1 1 0 1	D321	1 1 0 1 0 0 0 1
D231	1 0 0 1 1 0 0 1	D056	0 0 1 0 1 1 1 0	D322	1 1 0 1 0 0 1 0
D232	1 0 0 1 1 0 1 0	D063	0 0 1 1 0 0 1 1	D324	1 1 0 1 0 1 0 0
D234	1 0 0 1 1 1 0 0	D065	0 0 1 1 0 1 0 1	D330	1 1 0 1 1 0 0 0
D243	1 0 1 0 0 0 1 1	D066	0 0 1 1 0 1 1 0	D341	1 1 1 0 0 0 0 1
D245	1 0 1 0 0 1 0 1	D071	0 0 1 1 1 0 0 1	D342	1 1 1 0 0 0 1 0
D246	1 0 1 0 0 1 1 0	D072	0 0 1 1 1 0 1 0	D344	1 1 1 0 0 1 0 0
D251	1 0 1 0 1 0 0 1	D074	0 0 1 1 1 1 0 0	D350	1 1 1 0 1 0 0 0
D252	1 0 1 0 1 0 1 0				
D254	1 0 1 0 1 1 0 0				
D261	1 0 1 1 0 0 0 1				
D262	1 0 1 1 0 0 1 0				
D264	1 0 1 1 0 1 0 0				
D270	1 0 1 1 1 0 0 0				

The 20 vectors of FIG. 4 are listed in Table 2. The ten source vectors on the left side of the table have a negative disparity. The ten source vectors with positive disparity on the right side are the exact complements of those on the left side. Sixteen of these vectors require changes for encoding as follows:

1. The two vectors with all ones or zeros and a disparity of six are balanced by complementing three bits.
2. The 12 vectors with a disparity of four are encoded by complementing two bits.
3. The two data vectors with a disparity of two and a trailing run length of four are balanced by the complementation of a single bit.

In Table 2 and 3, the coded bits which are the complements of the respective source bits are printed in bold type and underlined. Note the symmetries for the set of 16 data vectors between the left and right side of Table 2 and between the 'abc' bits of the vectors D10, D20, and D40 and the 'def' bits of the vectors D04, D02, and D01, respectively. The bit positions complemented for encoding are identical for both vectors of a complementary pair. This feature can be exploited for simplifications in the encoding and decoding equations as described below under Alternate Implementation of Encoder and Decoder.

The complete 6B/8B coding assignments are shown in Table 3.

Table 3. 6B/8B Code

Name	K FEDCBA	Name	h g f e d c b a	Name	K FEDCBA	Name	h g f e d c b a
D00	0 0 0 0 0 0 0	D131	0 1 0 **1 1** 0 0 **1**	D42	0 1 0 0 0 1 0	D342	1 1 1 0 0 0 1 0
D01	0 0 0 0 0 0 1	D161	0 1 **1 1** 0 0 0 1	D43	0 1 0 0 0 1 1	D243	1 0 1 0 0 0 1 1
D02	0 0 0 0 0 1 0	D162	0 1 **1 1** 0 0 1 0	D44	0 1 0 0 1 0 0	D344	1 1 1 0 0 1 0 0
D03	0 0 0 0 0 1 1	D303	1 1 0 0 0 0 1 1	D45	0 1 0 0 1 0 1	D245	1 0 1 0 0 1 0 1
D04	0 0 0 0 1 0 0	D145	0 1 **1** 0 0 1 0 **1**	D46	0 1 0 0 1 1 0	D246	1 0 1 0 0 1 1 0
D05	0 0 0 0 1 0 1	D305	1 1 0 0 0 1 0 1	D47	0 1 0 0 1 1 1	D047	0 0 1 0 0 1 1 1
D06	0 0 0 0 1 1 0	D306	1 1 0 0 0 1 1 0	D50	0 1 0 1 0 0 0	D350	1 1 1 0 1 0 0 0
D07	0 0 0 0 1 1 1	D207	1 0 0 0 0 1 1 1	D51	0 1 0 1 0 0 1	D251	1 0 1 0 1 0 0 1
D10	0 0 0 1 0 0 0	D151	0 1 **1** 0 1 0 0 **1**	D52	0 1 0 1 0 1 0	D252	1 0 1 0 1 0 1 0
D11	0 0 0 1 0 0 1	D311	1 1 0 0 1 0 0 1	D53	0 1 0 1 0 1 1	D053	0 0 1 0 1 0 1 1
D12	0 0 0 1 0 1 0	D312	1 1 0 0 1 0 1 0	D54	0 1 0 1 1 0 0	D254	1 0 1 0 1 1 0 0
D13	0 0 0 1 0 1 1	D213	1 0 0 0 1 0 1 1	D55	0 1 0 1 1 0 1	D055	0 0 1 0 1 1 0 1
D14	0 0 0 1 1 0 0	D314	1 1 0 0 1 1 0 0	D56	0 1 0 1 1 1 0	D056	0 0 1 0 1 1 1 0
D15	0 0 0 1 1 0 1	D215	1 0 0 0 1 1 0 1	D57	0 1 0 1 1 1 1	D154	0 1 1 0 1 1 **0 0**
D16	0 0 0 1 1 1 0	D216	1 0 0 0 1 1 1 0	D60	0 1 1 0 0 0 0	D164	0 1 1 1 0 **1** 0 0
D17	0 0 0 1 1 1 1	D113	0 1 0 0 1 **0** 1 1	D61	0 1 1 0 0 0 1	D261	1 0 1 1 0 0 0 1
D20	0 0 1 0 0 0 0	D123	0 1 0 1 0 0 **1 1**	D62	0 1 1 0 0 1 0	D262	1 0 1 1 0 0 1 0
D21	0 0 1 0 0 0 1	D321	1 1 0 1 0 0 0 1	D63	0 1 1 0 0 1 1	D063	0 0 1 1 0 0 1 1
D22	0 0 1 0 0 1 0	D322	1 1 0 1 0 0 1 0	D64	0 1 1 0 1 0 0	D264	1 0 1 1 0 1 0 0
D23	0 0 1 0 0 1 1	D223	1 0 0 1 0 0 1 1	D65	0 1 1 0 1 0 1	D065	0 0 1 1 0 1 0 1
D24	0 0 1 0 1 0 0	D324	1 1 0 1 0 1 0 0	D66	0 1 1 0 1 1 0	D066	0 0 1 1 0 1 1 0
D25	0 0 1 0 1 0 1	D225	1 0 0 1 0 1 0 1	D67	0 1 1 0 1 1 1	D126	0 1 **0** 1 0 1 1 **0**
D26	0 0 1 0 1 1 0	D226	1 0 0 1 0 1 1 0	D70	0 1 1 1 0 0 0	D270	1 0 1 1 1 0 0 0
D27	0 0 1 0 1 1 1	D027	0 0 0 1 0 1 1 1	D71	0 1 1 1 0 0 1	D071	0 0 1 1 1 0 0 1
D30	0 0 1 1 0 0 0	D330	1 1 0 1 1 0 0 0	D72	0 1 1 1 0 1 0	D072	0 0 1 1 1 0 1 0
D31	0 0 1 1 0 0 1	D231	1 0 0 1 1 0 0 1	D73	0 1 1 1 0 1 1	D132	0 1 **0** 1 1 0 1 **0**
D32	0 0 1 1 0 1 0	D232	1 0 0 1 1 0 1 0	D74	0 1 1 1 1 0 0	D074	0 0 1 1 1 1 0 0
D33	0 0 1 1 0 1 1	D033	0 0 0 1 1 0 1 1	D75	0 1 1 1 1 0 1	D115	0 1 **0 0** 1 1 0 1
D34	0 0 1 1 1 0 0	D234	1 0 0 1 1 1 0 0	D76	0 1 1 1 1 1 0	D116	0 1 **0 0** 1 1 1 0
D35	0 0 1 1 1 0 1	D035	0 0 0 1 1 1 0 1	D77	0 1 1 1 1 1 1	D146	0 1 1 **0 0** 1 1 **0**
D36	0 0 1 1 1 1 0	D036	0 0 0 1 1 1 1 0	K07	1 0 0 0 1 1 1	K107	0 1 0 0 0 1 1 1
D37	0 0 1 1 1 1 1	D134	0 1 0 1 1 1 **0 0**	K25	1 0 1 0 1 0 1	K125	0 1 0 1 0 1 0 1
D40	0 1 0 0 0 0 0	D143	0 1 1 0 0 0 **1 1**	K52	1 1 0 1 0 1 0	K152	0 1 1 0 1 0 1 0
D41	0 1 0 0 0 0 1	D341	1 1 1 0 0 0 0 1	K70	1 1 1 1 0 0 0	K170	0 1 1 1 1 0 0 0

Table 2. 20 Encoded Vectors of FIG. 4 with individual Bit Changes

Name	K FEDCBA	Name	h g f e d c b a	Name	K FEDCBA	Name	h g f e d c b a
D00	0 000000	D131	0 1 0 **1 1** 0 0 **1**	D77	0 111111	D146	0 1 1 **0 0** 1 1 **0**
D01	0 000001	D161	0 1 **1 1** 0 0 0 1	D76	0 111110	D116	0 1 **0 0** 1 1 1 0
D02	0 000010	D162	0 1 **1 1** 0 0 1 0	D75	0 111101	D115	0 1 **0 0** 1 1 0 1
D04	0 000100	D145	0 1 **1** 0 0 1 0 **1**	D73	0 111011	D132	0 1 **0** 1 1 0 1 **0**
D10	0 001000	D151	0 1 **1** 0 1 0 0 **1**	D67	0 110111	D126	0 1 **0** 1 0 1 1 **0**
D20	0 010000	D123	0 1 0 1 0 0 **1 1**	D57	0 101111	D154	0 1 1 0 1 1 **0 0**
D40	0 100000	D143	0 1 1 0 0 0 **1 1**	D37	0 011111	D134	0 1 0 1 1 1 **0 0**
D60	0 110000	D164	0 1 1 1 0 **1** 0 0	D17	0 001111	D113	0 1 0 0 1 **0** 1 1
K07	1 000111	K107	0 1 0 0 0 1 1 1	K70	1 111000	K170	0 1 1 1 1 0 0 0
K25	1 010101	K125	0 1 0 1 0 1 0 1	K52	1 101010	K152	0 1 1 0 1 0 1 0

Generation of Encoded 8B Vectors

For the derivation of the encoding equations refer to the Tables 2 and/or 3. Generally, the encoded bits retain the value of the unencoded bit (a=A, b=B, etc), but a specific source bit is complemented (a=A', b=B', etc) if and only if (iff) the respective equation is true. In the Coding Labels and equations, some bit values are included redundantly to allow more circuit sharing for the complementation of other bits of the same vector which are also typed in bold type and underlined in the Tables 2 and 3. Redundant bit values in the Tables 4 through 18 and in the equations are overlined and the names of redundant vectors in the tables are preceded by an asterisk.

Encoded Bit a

The 'a' column has bold entries for D00, D04, D10, D20, D40, D37, D57, D67, D73, and D77. The respective uncoded bits FEDCBA are listed in Table 4, the A-bit is overlined, and common patterns in the source bits are marked by bold type to logically classify the vectors by simple expressions.

Table 4. a-bit Encoding

Name	K FEDCBA	Coding Label	a	Name	K FEDCBA	Coding Label	a
D00	$x\ 00000\overline{0}$	$\overline{A'}\bullet B'\bullet C'\bullet D'\bullet E'\bullet F'$	1	D77	$x\ 11111\overline{1}$	$\overline{A}\bullet B\bullet C\bullet D\bullet E\bullet F$	0
D04	$x\ \mathbf{00}01\mathbf{0}\overline{0}$	$\overline{A'}\bullet B'\bullet E'\bullet F'\bullet C\neq D$	1	D73	$x\ \mathbf{11}10\mathbf{1}\overline{1}$	$\overline{A}\bullet B\bullet E\bullet F\bullet C\neq D$	0
D10	$x\ \mathbf{00}10\mathbf{0}\overline{0}$	$\overline{A'}\bullet B'\bullet E'\bullet F'\bullet C\neq D$	1	D67	$x\ \mathbf{11}01\mathbf{1}\overline{1}$	$\overline{A}\bullet B\bullet E\bullet F\bullet C\neq D$	0
D20	$x\ 01\mathbf{000}\overline{0}$	$\overline{A'}\bullet B'\bullet C'\bullet D'\bullet E\neq F$	1	D57	$x\ 10\mathbf{111}\overline{1}$	$\overline{A}\bullet B\bullet C\bullet D\bullet E\neq F$	0
D40	$x\ 10\mathbf{000}\overline{0}$	$\overline{A'}\bullet B'\bullet C'\bullet D'\bullet E\neq F$	1	D37	$x\ 01\mathbf{111}\overline{1}$	$\overline{A}\bullet B\bullet C\bullet D\bullet E\neq F$	0

Using these identifiers, the encoding equation for bit 'a' can be written as follows:

$$a = A\bullet[\overline{A}\bullet B\bullet E\bullet F\bullet C\oplus D+\overline{A}\bullet B\bullet C\bullet D\bullet E\oplus F+\overline{A}\bullet B\bullet C\bullet D\bullet E\bullet F]'+ \overline{A'}\bullet B'\bullet E'\bullet F'\bullet C\oplus D+\overline{A'}\bullet B'\bullet C'\bullet D'\bullet E\oplus F+\overline{A'}\bullet B'\bullet C'\bullet D'\bullet E'\bullet F'$$

In the circuit diagram s68encode of FIG. 7B, the following net names are used:

$$n1 = \overline{A'} \bullet B' \bullet E' \bullet F' \bullet C \oplus D + \overline{A'} \bullet B' \bullet C' \bullet D' \bullet E \oplus F + \overline{A'} \bullet B' \bullet C' \bullet D' \bullet E' \bullet F'$$
$$n2 = A \bullet (n3)'$$
$$n3 = \overline{A} \bullet B \bullet E \bullet F \bullet C \oplus D + \overline{A} \bullet B \bullet C \bullet D \bullet E \oplus F + \overline{A} \bullet B \bullet C \bullet D \bullet E \bullet F$$

Encoded Bit b

The ‘b’ column has bold entries for D20, D40, D37, and D57. The respective uncoded bits FEDCBA are listed in Table 5, the B-bit is overlined, and common patterns are marked.

Table 5. b-bit Encoding

Name	K FEDCBA	Coding Label	b	Name	K FEDCBA	Coding Label	b
D20	x 01$\mathbf{00\overline{0}0}$	$A' \bullet \overline{B'} \bullet C' \bullet D' \bullet E \neq F$	1	D57	x 10$\mathbf{11\overline{1}1}$	$A \bullet \overline{B} \bullet C \bullet D \bullet E \neq F$	0
D40	x 10$\mathbf{00\overline{0}0}$	$A' \bullet \overline{B'} \bullet C' \bullet D' \bullet E \neq F$	1	D37	x 01$\mathbf{11\overline{1}1}$	$A \bullet \overline{B} \bullet C \bullet D \bullet E \neq F$	0

Using these identifiers, the encoding equation for bit ‘b’ can be written as follows:

$$b = B \bullet (A \bullet \overline{B} \bullet C \bullet D \bullet E \oplus F)' + A' \bullet \overline{B'} \bullet C' \bullet D' \bullet E \oplus F$$

In the circuit diagram of FIG. 7B, the following net name is used:

$$n11 = B \bullet (A \bullet \overline{B} \bullet C \bullet D \bullet E \oplus F)'$$

Encoded Bit c

The ‘c’ column has bold entries for D60 and D17. The respective uncoded bits FEDCBA are listed in Table 6, the C-bit is overlined.

Table 6. c-bit Encoding

Name	K FEDCBA	Coding Label	c	Name	K FEDCBA	Coding Label	c
D60	x 110$\overline{0}$00	$A' \bullet B' \bullet D' \bullet E \bullet F$	1	D17	x 001$\overline{1}$11	$A \bullet B \bullet D \bullet E' \bullet F'$	0

Using these identifiers, the encoding equation for bit ‘c’ can be written as follows:

$$c = C \bullet (A \bullet B \bullet D \bullet E' \bullet F')' + A' \bullet B' \bullet D' \bullet E \bullet F$$

In the circuit diagram of FIG. 7B, the following net name is used:

$$n21 = C \bullet (A \bullet B \bullet D \bullet E' \bullet F')'$$

Encoded Bit d

The ‘d’ column has bold entries for D00 and D77. The respective uncoded bits FEDCBA are listed in Table 7, the D-bit is overlined.

Table 7. d-bit Encoding

Name	K FEDCBA	Coding Label	d	Name	K FEDCBA	Coding Label	d
D00	x 00$\overline{0}$000	$A' \bullet B' \bullet C' \bullet \overline{D'} \bullet E' \bullet F'$	1	D77	x 11$\overline{1}$111	$A \bullet B \bullet C \bullet \overline{D} \bullet E \bullet F$	0

Using these identifiers, the encoding equation for bit 'd' can be written as follows:

$$d = D\bullet(A\bullet B\bullet C\bullet \overline{D}\bullet E\bullet F)' + A'\bullet B'\bullet C'\bullet \overline{D'}\bullet E'\bullet F'$$

In the circuit diagram of FIG. 7B, the following net name is used:

$n31 = D\bullet(A\bullet B\bullet C\bullet \overline{D}\bullet E\bullet F)'$

Encoded Bit e

The 'e' column has bold entries for D00, D01, D02, D75, D76, and D77. Table 8 lists the respective uncoded bits FEDCBA, the E-bit is overlined, and common patterns are marked by bold entries.

Table 8. e-bit Encoding

Name	K FEDCBA	Coding Label	e	Name	K FEDCBA	Coding Label	e
D00	x $0\overline{0}0000$	$A'\bullet B'\bullet C'\bullet D'\bullet \overline{E'}\bullet F'$	1	D77	x $1\overline{1}1111$	$A\bullet B\bullet C\bullet D\bullet \overline{E}\bullet F$	0
D01	x $\mathbf{0\overline{0}00}01$	$C'\bullet D'\bullet \overline{E'}\bullet F'\bullet A\neq B$	1	D76	x $\mathbf{1\overline{1}11}10$	$C\bullet D\bullet \overline{E}\bullet F\bullet A\neq B$	0
D02	x $\mathbf{0\overline{0}00}10$	$C'\bullet D'\bullet \overline{E'}\bullet F'\bullet A\neq B$	1	D75	x $\mathbf{1\overline{1}11}01$	$C\bullet D\bullet \overline{E}\bullet F\bullet A\neq B$	0

Using these identifiers, the encoding equation for bit 'e' can be written as follows:

$$e = E\bullet(C\bullet D\bullet \overline{E}\bullet F\bullet A\oplus B + A\bullet B\bullet C\bullet D\bullet \overline{E}\bullet F)' + C'\bullet D'\bullet \overline{E'}\bullet F'\bullet A\oplus B + A'\bullet B'\bullet C'\bullet D'\bullet \overline{E'}\bullet F'$$

In the circuit diagram of FIG. 7B, the following net name is used:

$n41 = E\bullet(n42)'$
$n42 = C\bullet D\bullet \overline{E}\bullet F\bullet A\oplus B + A\bullet B\bullet C\bullet D\bullet \overline{E}\bullet F$

Encoded Bit f

The 'f' column has bold entries for *D01, D02,* D04, D10, D67, D73, D75, and D76. The respective uncoded bits FEDCBA are listed in Table 9. The F-bit is overlined, and common patterns are marked by bold entries.

Table 9. f-bit Encoding

Name	K FEDCBA	Coding Label	f	Name	K FEDCBA	Coding Label	f
D01	x $\overline{0}\mathbf{000}01$	$C'\bullet D'\bullet E'\bullet \overline{F'}\bullet A\neq B$	1	D76	x $\overline{1}11110$	$C\bullet D\bullet E\bullet \overline{F}\bullet A\neq B$	0
D02	x $\overline{0}\mathbf{000}10$	$C'\bullet D'\bullet E'\bullet \overline{F'}\bullet A\neq B$	1	D75	x $\overline{1}11101$	$C\bullet D\bullet E\bullet \overline{F}\bullet A\neq B$	0
D04	x $\overline{0}\mathbf{0}01\mathbf{00}$	$A'\bullet B'\bullet E'\bullet \overline{F'}\bullet C\neq D$	1	D73	x $\overline{1}11011$	$A'\bullet B\bullet E\bullet \overline{F}\bullet C\neq D$	0
D10	x $\overline{0}\mathbf{0}10\mathbf{00}$	$A'\bullet B'\bullet E'\bullet \overline{F'}\bullet C\neq D$	1	D67	x 110111	$A'\bullet B\bullet E\bullet \overline{F}\bullet C\neq D$	0

The encoding equation for bit 'f' can be written as follows:

$$f = F\bullet(C\bullet D\bullet E\bullet \overline{F}\bullet A\oplus B + A'\bullet B\bullet E\bullet \overline{F}\bullet C\oplus D)' + C'\bullet D'\bullet E'\bullet \overline{F'}\bullet A\oplus B + A'\bullet B'\bullet E'\bullet \overline{F'}\bullet C\oplus D$$

In the circuit diagram of FIG. 7B, the following net name is used:

$n51 = F \bullet (n52)'$

$n52 = C \bullet D \bullet E \bullet \overline{F} \bullet A \oplus B + A \bullet B \bullet E \bullet \overline{F} \bullet C \oplus D$

Encoded Bit g

The value for bit 'g' is one for the 34 vectors of FIG. 4 and FIG. 6. These vectors are enumerated in the right column of Table 1 and in Table 2. Note that the source vectors for all coded vectors of Table 1 are identical to the trailing 6 bits of the respective coded vector and are not listed explicitly.

The 34 source vectors for which the value for bit 'g' is one are listed again in Table 10. All 22 source vectors with four or more zeros are part of this set. For the derivation of logical encoding equations, these 22 vectors are grouped into 3 overlapping sets. The redundant vectors are marked by an asterisk. The set of 7 source vectors at the top left side of the table is characterized by three trailing zeros and at least one bit with a value of zero in the leading three bit positions which is described by the logic expression $A' \bullet B' \bullet C' \bullet (D'+E'+F')$. The set of 6 source vectors (not counting the redundant vector *D00) at the top right side of the table is characterized by three leading zeros and at least one bit with a value of zero in the trailing three bit positions which is described by the logic expression $(A'+B'+C') \bullet D' \bullet E' \bullet F'$. The set of nine vectors (not counting the redundant vectors) at the bottom of the left side is characterized by at least 2 zeros in the leading three positions and at least 2 zeros in the trailing three positions which is described by the logic expression $(A' \bullet B'+A' \bullet C'+B' \bullet C') \bullet (D' \bullet E'+D' \bullet F'+E' \bullet F')$.

The four vectors on the right side with a trailing run of four ones are identified by the logic expression $A \bullet B \bullet C \bullet D$. The two vectors with four leading ones are identified by the logic expression $C \bullet D \bullet E \bullet F \bullet A \oplus B$ and the two vectors with two leading ones and two trailing ones are identified by the logic expression $A \bullet B \bullet E \bullet F \bullet C \oplus D$.

Finally, all four control vectors identified by a K-value of one have a g-value of one.

The logic equation for the encoding of the g-bit can thus be expressed as follows:

$$g = A' \bullet B' \bullet C' \bullet (D'+E'+F') + (A'+B'+C') \bullet D' \bullet E' \bullet F' + (A \bullet B \bullet C \bullet D) + C \bullet D \bullet E \bullet F \bullet A \oplus B + A \bullet B \bullet E \bullet F \bullet C \oplus D + (A' \bullet B'+A' \bullet C'+B' \bullet C') \bullet (D' \bullet E'+D' \bullet F'+E' \bullet F') + K$$

In the circuit diagram of FIG. 7B, the following net names are used:

$n61 = n64 \bullet n65$	$n62 = n66 + A \bullet B \bullet C \bullet D + A \bullet B \bullet E \bullet F \bullet C \oplus D$
$n63 = n67 + n68 + K$	$n64 = A' \bullet B'+A' \bullet C'+B' \bullet C'$
$n65 = D' \bullet E'+D' \bullet F'+E' \bullet F'$	$n66 = C \bullet D \bullet E \bullet F \bullet A \oplus B$
$n67 = A' \bullet B' \bullet C' \bullet (D'+E'+F')$	$n68 = (A'+B'+C') \bullet D' \bullet E' \bullet F'$

Table 10. g-bit Encoding

Name	K FEDCBA	Coding Label	g	Name	K FEDCBA	Coding Label	g
D00	x 000**000**	A'•B'•C'•(D'+E'+F')	1	*D00	x **000**000	(A'+B'+C')•D'•E'•F'	1
D10	x 001**000**	A'•B'•C'•(D'+E'+F')	1	D01	x **000**001	(A'+B'+C')•D'•E'•F'	1
D20	x 010**000**	A'•B'•C'•(D'+E'+F')	1	D02	x **000**010	(A'+B'+C')•D'•E'•F'	1
D30	x 011**000**	A'•B'•C'•(D'+E'+F')	1	D03	x **000**011	(A'+B'+C')•D'•E'•F'	1
D40	x 100**000**	A'•B'•C'•(D'+E'+F')	1	D04	x **000**100	(A'+B'+C')•D'•E'•F'	1
D50	x 101**000**	A'•B'•C'•(D'+E'+F')	1	D05	x **000**101	(A'+B'+C')•D'•E'•F'	1
D60	x 110**000**	A'•B'•C'•(D'+E'+F')	1	D06	x **000**110	(A'+B'+C')•D'•E'•F'	1
D11	x **00**1001	(A'•B'+A'•C'+B'•C')•E'•F'	1	D17	x 00**1111**	A•B•C•D	1
D12	x **00**1010	(A'•B'+A'•C'+B'•C')•E'•F'	1	D37	x 01**1111**	A•B•C•D	1
D14	x **00**1100	(A'•B'+A'•C'+B'•C')•E'•F'	1	D57	x 10**1111**	A•B•C•D	1
D21	x **0**1**0**001	(A'•B'+A'•C'+B'•C')•D'•F'	1	D77	x 11**1111**	A•B•C•D	1
D22	x **0**1**0**010	(A'•B'+A'•C'+B'•C')•D'•F'	1	D75	x **1111**01	C•D•E•F•A≠B	1
D24	x **0**1**0**100	(A'•B'+A'•C'+B'•C')•D'•F'	1	D76	x **1111**10	C•D•E•F•A≠B	1
D41	x 1**00**001	(A'•B'+A'•C'+B'•C')•D'•E'	1	D67	x **11**01**11**	A•B•E•F•C≠D	1
D42	x 1**00**010	(A'•B'+A'•C'+B'•C')•D'•E'	1	D73	x **11**10**11**	A•B•E•F•C≠D	1
D44	x 1**00**100	(A'•B'+A'•C'+B'•C')•D'•E'	1	K07	**1** 000111	K	1
*D00	x 000000		1	K25	**1** 010101	K	1
*D01	x 000001		1	K52	**1** 101010	K	1
*D02	x 000010		1	K70	**1** 111000	K	1
*D04	x 000100		1				
*D10	x 001000		1				
*D20	x 010000		1				
*D40	x 100000		1				

Encoded Bit h

The value for bit 'h' is zero for the vectors of FIG. 4 and FIG. 5. These vectors are enumerated in the center column of Table 1 and in Table 2. Note that the source vectors for all coded vectors of Table 1 are identical to the trailing 6 bits of the respective coded vector and are not listed explicitly.

The 34 source vectors for which the value for bit 'h' is zero are listed again in Table 11. All 22 source vectors with four or more ones are part of this set. For the derivation of logical encoding equations, these 22 vectors are grouped into 3 overlapping sets. The names of redundant vectors are marked by an asterisk.

Table 11. h-bit Encoding

Name	K FEDCBA	Coding Label	h	Name	K FEDCBA	Coding Label	h
D17	x 001**111**	A•B•C•(D+E+F)	0	D71	x **111**001	(A+B+C)•D•E•F	0
D27	x 010**111**	A•B•C•(D+E+F)	0	D72	x **111**010	(A+B+C)•D•E•F	0
D37	x 011**111**	A•B•C•(D+E+F)	0	D73	x **111**011	(A+B+C)•D•E•F	0
D47	x 100**111**	A•B•C•(D+E+F)	0	D74	x **111**100	(A+B+C)•D•E•F	0
D57	x 101**111**	A•B•C•(D+E+F)	0	D75	x **111**101	(A+B+C)•D•E•F	0
D67	x 110**111**	A•B•C•(D+E+F)	0	D76	x **111**110	(A+B+C)•D•E•F	0
D77	x 111**111**	A•B•C•(D+E+F)	0	*D77	x **111**111	(A+B+C)•D•E•F	0
D33	x 0**11**011	(A•B+A•C+B•C)•D•E	0	D00	x 00**0000**	A’•B’•C’•D’	0
D35	x 0**11**101	(A•B+A•C+B•C)•D•E	0	D20	x 01**0000**	A’•B’•C’•D’	0
D36	x 0**11**110	(A•B+A•C+B•C)•D•E	0	D40	x 10**0000**	A’•B’•C’•D’	0
D53	x **1**0**1**011	(A•B+A•C+B•C)•D•F	0	D60	x 11**0000**	A’•B’•C’•D’	0
D55	x **1**0**1**101	(A•B+A•C+B•C)•D•F	0	D01	x **0000**01	C’•D’•E’•F’•A≠B	0
D56	x **1**0**1**110	(A•B+A•C+B•C)•D•F	0	D02	x **0000**10	C’•D’•E’•F’•A≠B	0
D63	x **11**0011	(A•B+A•C+B•C)•E•F	0	D04	x **00**01**00**	A’•B’•E’•F’•C≠D	0
D65	x **11**0101	(A•B+A•C+B•C)•E•F	0	D10	x **00**10**00**	A’•B’•E’•F’•C≠D	0
D66	x **11**0110	(A•B+A•C+B•C)•E•F	0	K07	**1** 000111	K	0
*D37	x 011111		0	K25	**1** 010101	K	0
*D57	x 101111		0	K52	**1** 101010	K	0
*D67	x 110111			K70	**1** 111000	K	0
*D73	x 111011						
*D75	x 111101						
*D76	x 111110						
*D77	x 111111						

The set of 7 source vectors at the top of the left side is characterized by three trailing ones and at least one bit with a value of one in the leading three bit positions which is described by the logic expression A•B•C•(D+E+F). The set of 6 source vectors (not counting the redundant vector *D77) at the top of the right side is characterized by three leading ones and at least one bit with a value of one in the trailing three bit positions which is described by the logic expression (A+B+C)•D•E•F. The set of nine vectors (not counting the redundant vectors) at the bottom of the left side is characterized by at least two ones in the trailing three positions and at least two ones in the leading three positions which is described by the logic expression (A•B+A•C+B•C)•(D•E+D•F+E•F).

The four vectors with a trailing run of four zeros on the right side are identified by the logic expression A’•B’•C’•D’. The two vectors with four leading zeros are identified by the logic expression C’•D’•E’•F’•A≠B. The two vectors with two leading and two trailing zeros are identified by the logic expression A’•B’•E’•F’•C≠D.

Finally, all four control vectors identified by a K-value of one have an h-value of zero. The logic equation for the encoding of the h-bit can thus be expressed as follows:

$$h = [A\bullet B\bullet C\bullet(D+E+F)+(A+B+C)\bullet D\bullet E\bullet F+(A\bullet B+A\bullet C+B\bullet C)\bullet(D\bullet E+D\bullet F+E\bullet F) + (A'\bullet B'\bullet C'\bullet D') + C'\bullet D'\bullet E'\bullet F'\bullet A\oplus B + A'\bullet B'\bullet E'\bullet F'\bullet C\oplus D + K]'$$

In the circuit diagram of FIG. 7B, the following net names are used:

$n71 = n74 \bullet n75$	$n72 = n78 + A'\bullet B'\bullet C'\bullet D' + A'\bullet B'\bullet E'\bullet F'\bullet C\oplus D$
$n73 = n76 + n77 + K$	$n74 = A\bullet B+A\bullet C+B\bullet C$
$n75 = D\bullet E+D\bullet F+E\bullet F$	$n76 = A\bullet B\bullet C\bullet(D+E+F)$
$n77 = (A+B+C)\bullet D\bullet E\bullet F$	$n78 = C'\bullet D'\bullet E'\bullet F'\bullet A\oplus B$

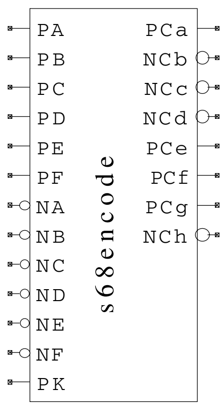


FIG. 7A.

Circuit Implementation of the 6B/8B-P Encoder

FIG. 7A is a symbol of the encoder circuit showing the inputs and outputs.

FIG. 7B is the logic diagram of the encoder circuit. It has not yet been simulated and verified by design tools, so it may be afflicted with minor errors.

Alternate Implementation of Encoder

Because of the symmetries between the left and the right side of Table 2, all the encoding equations for the bits ‘a’ through ‘f’ have complementary features which can be exploited by the extensive use of the Exclusive OR function. The transformed coding equations for the bits ‘a’ thorough ‘f’ are presented here:

$$a=A\oplus[\overline{A}\oplus B'\bullet(B\oplus E'\bullet E\oplus F'\bullet C\oplus D+B\oplus C'\bullet C\oplus D'\bullet E\oplus F+B\oplus C'\bullet C\oplus D'\bullet D\oplus E'\bullet E\oplus F')]$$

$$b = B\oplus(A\oplus\overline{B}'\bullet\overline{B}\oplus C'\bullet C\oplus D'\bullet E\oplus F)$$

$$c=C\oplus(A\oplus B'\bullet B\oplus D'\bullet D\oplus E\bullet E\oplus F')$$

$$d=D\oplus(A\oplus B'\bullet B\oplus C'\bullet C\oplus\overline{D}\bullet\overline{D}\oplus E'\bullet E\oplus F')$$

$$e=E\oplus[(C\oplus D'\bullet D\oplus\overline{E}'\bullet\overline{E}\oplus F')\bullet(A\oplus B+A\oplus B'\bullet B\oplus C')]$$

$$f=F\oplus[(E\oplus\overline{F}')\bullet(C\oplus D'\bullet D\oplus E'\bullet A\oplus B+A\oplus B'\bullet B\oplus E'\bullet C\oplus D)]$$

An implementation based on these alternate equations may be advantageous in terms of silicon area. To support a selection for a particular technology and application, the circuit delay and the total circuit capacity related to power dissipation must also be considered.

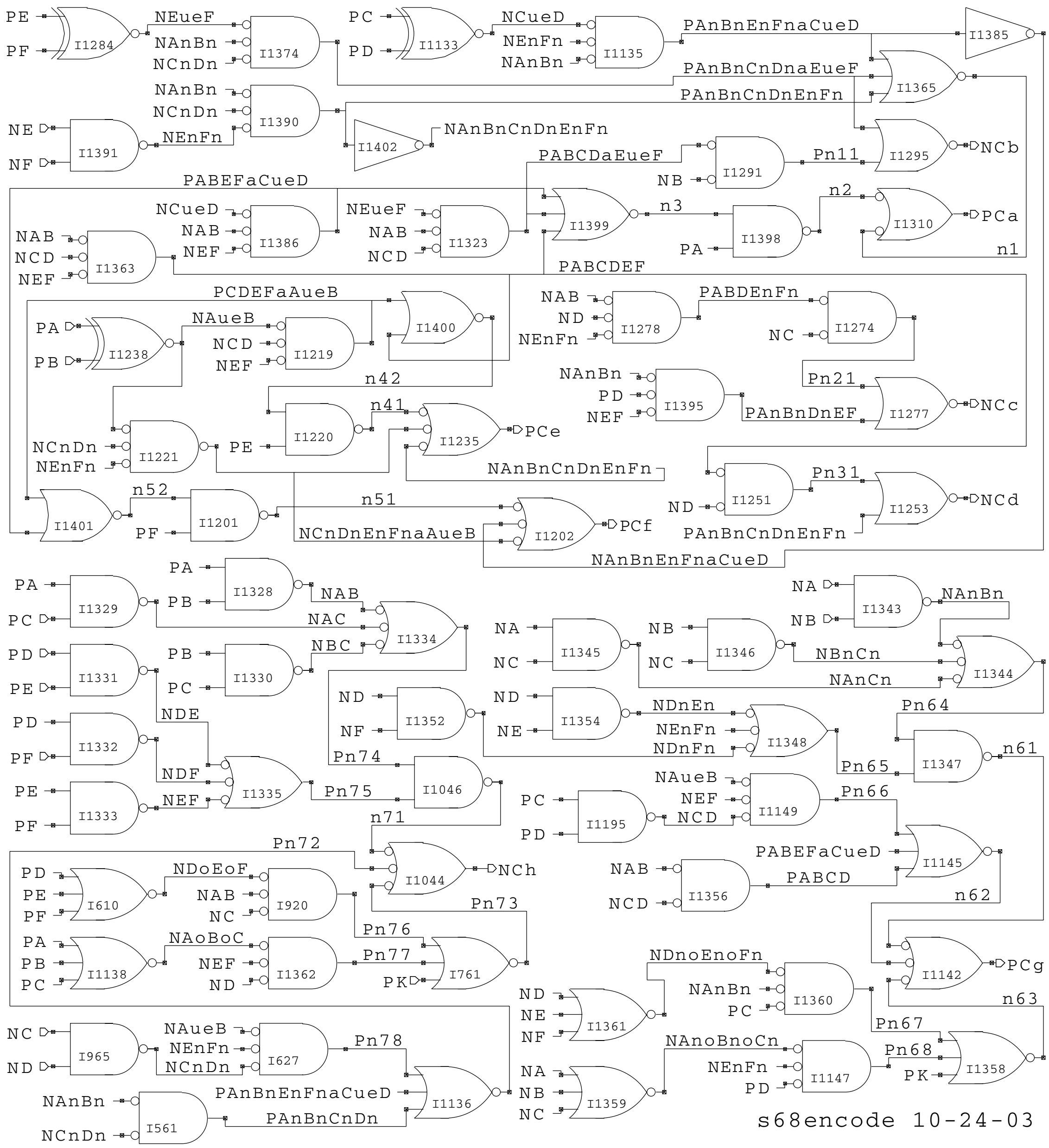


FIG. 7B. 6B/8B ENCODING CIRCUIT

Generation of Decoded 6B Vectors

For all encoded vectors ‘hgfedcba’ with a value hg≠01 the decoded bits FEDCBA are equal to the encoded bits ‘fedcba’ and the value of the K-bit is zero. If hg=01, the decoding equations can be derived from Table 2 or 3 as shown below similar to encoding. Redundancies are introduced for the same purposes as for encoding and marked in the same way.

Decoded Bit A

The 'a' column of Table 2 has bold entries for D131, D145, D151, D123, D143, D146, D132, D126, D154, and D134. The respective coded bits 'hgfedcba' are listed in Table 12, the a-bit is overlined to indicate redundancy, and common patterns are marked by bold type to logically classify the vectors by simple expressions.

Table 12. A-bit Decoding

Name	hgfedcba	Coding Label	AK	Name	hgfedcba	Coding Label	AK
D131	0101100$\overline{1}$	$\overline{a}\bullet b'\bullet c'\bullet d\bullet e\bullet f'\bullet g\bullet h'$	00	D146	0110011$\overline{0}$	$\overline{a'}\bullet b\bullet c\bullet d'\bullet e'\bullet f\bullet g\bullet h'$	10
D145	**0110**01**0**$\overline{1}$	$\overline{a}\bullet b'\bullet e'\bullet f\bullet g\bullet h'\bullet c\neq d$	00	D132	**0101**10**1**$\overline{0}$	$\overline{a'}\bullet b\bullet e\bullet f'\bullet g\bullet h'\bullet c\neq d$	10
D151	**0110**10**0**$\overline{1}$	$\overline{a}\bullet b'\bullet e'\bullet f\bullet g\bullet h'\bullet c\neq d$	00	D126	**0101**01**1**$\overline{0}$	$\overline{a'}\bullet b\bullet e\bullet f'\bullet g\bullet h'\bullet c\neq d$	10
D123	**01**01**001**$\overline{1}$	$\overline{a}\bullet b\bullet c'\bullet d'\bullet g\bullet h'\bullet e\neq f$	00	D154	**01**10**110**$\overline{0}$	$\overline{a'}\bullet b'\bullet c\bullet d\bullet g\bullet h'\bullet e\neq f$	10
D143	**01**10**001**$\overline{1}$	$\overline{a}\bullet b\bullet c'\bullet d'\bullet g\bullet h'\bullet e\neq f$	00	D134	**01**01**110**$\overline{0}$	$\overline{a'}\bullet b'\bullet c\bullet d\bullet g\bullet h'\bullet e\neq f$	10

Using these identifiers, the decoding equation for bit 'A' can be written as follows:

$$A = a\bullet(\overline{a}\bullet b'\bullet c'\bullet d\bullet e\bullet f'\bullet g\bullet h' + \overline{a}\bullet b'\bullet e'\bullet f\bullet g\bullet h'\bullet c\oplus d + \overline{a}\bullet b\bullet c'\bullet d'\bullet g\bullet h'\bullet e\oplus f)' +$$
$$\overline{a'}\bullet b\bullet c\bullet d'\bullet e'\bullet f\bullet g\bullet h' + \overline{a'}\bullet b\bullet e\bullet f'\bullet g\bullet h'\bullet c\oplus d + \overline{a'}\bullet b'\bullet c\bullet d\bullet g\bullet h'\bullet e\oplus f$$

In the circuit diagram r86decode of FIG. 8B, the following net names are used:

$n1 = a\bullet n3'$

$n2 = a'\bullet b\bullet c\bullet d'\bullet e'\bullet f\bullet g\bullet h' + a'\bullet b\bullet e\bullet f'\bullet g\bullet h'\bullet c\oplus d + a'\bullet b'\bullet c\bullet d\bullet g\bullet h'\bullet e\oplus f$

$n3 = a\bullet b'\bullet c'\bullet d\bullet e\bullet f'\bullet g\bullet h' + a\bullet b'\bullet e'\bullet f\bullet g\bullet h'\bullet c\oplus d + a\bullet b\bullet c'\bullet d'\bullet g\bullet h'\bullet e\oplus f$

Decoded Bit B

The 'b' column of Table 2 has bold entries for D123, D143, D154, and D134. The respective coded bits 'hgfedcba' are listed in Table 13, the b-bit is overlined, and common patterns are marked by bold type.

Table 13. B-bit Decoding

Name	hgfedcba	Coding Label	BK	Name	hgfedcba	Coding Label	BK
D123	**01**01**00**$\overline{1}$**1**	$a\bullet\overline{b}\bullet c'\bullet d'\bullet g\bullet h'\bullet e\neq f$	00	D154	**01**10**11**$\overline{0}$**0**	$a'\bullet\overline{b'}\bullet c\bullet d\bullet g\bullet h'\bullet e\neq f$	10
D143	**01**10**00**$\overline{1}$**1**	$a\bullet\overline{b}\bullet c'\bullet d'\bullet g\bullet h'\bullet e\neq f$	00	D134	**01**01**11**$\overline{0}$**0**	$a'\bullet\overline{b'}\bullet c\bullet d\bullet g\bullet h'\bullet e\neq f$	10

Using these identifiers, the decoding equation for bit 'B' can be written as follows:

$$B = b\bullet(a\bullet\overline{b}\bullet c'\bullet d'\bullet g\bullet h'\bullet e\oplus f)' + a'\bullet\overline{b'}\bullet c\bullet d\bullet g\bullet h'\bullet e\oplus f$$

In the circuit diagram of FIG. 8B, the following net name is used:

$n11 = b\bullet(a\bullet b\bullet c'\bullet d'\bullet g\bullet h'\bullet e\oplus f)'$

Decoded Bit C

The 'c' column of Table 2 has bold entries for D164 and D113. The respective coded bits 'hgfedcba' are listed in Table 14.

Table 14. C-bit Decoding

Name	hgfedcba	Coding Label	CK	Name	hgfedcba	Coding Label	CK
D164	$01110\overline{1}00$	a'•b'•d'•e•f•g•h'	00	D113	$01001\overline{0}11$	a•b•d•e'•f'•g•h'	10

Using these identifiers, the decoding equation for bit 'C' can be written as follows:
$C = c \bullet (a' \bullet b' \bullet d' \bullet e \bullet f \bullet g \bullet h')' + a \bullet b \bullet d \bullet e' \bullet f' \bullet g \bullet h'$

In the circuit diagram of FIG. 8B, the following net name is used:
n21 = c•(a'•b'•d'•e•f•g•h')'

Decoded Bit D

The 'd' column of Table 2 has bold entries for D131 and D146. The respective coded bits 'hgfedcba' are listed in Table 15.

Table 15. D-bit Decoding

Name	hgfedcba	Coding Label	DK	Name	hgfedcba	Coding Label	DK
D131	$0101\overline{1}001$	$a \bullet b' \bullet c' \bullet \overline{d} \bullet e \bullet f' \bullet g \bullet h'$	00	D146	$0110\overline{0}110$	$a' \bullet b \bullet c \bullet \overline{d'} \bullet e' \bullet f \bullet g \bullet h'$	10

Using these identifiers, the decoding equation for bit 'D' can be written as follows:

$D = d \bullet (a \bullet b' \bullet c' \bullet \overline{d} \bullet e \bullet f' \bullet g \bullet h')' + a' \bullet b \bullet c \bullet \overline{d'} \bullet e' \bullet f \bullet g \bullet h'$

In the circuit diagram of FIG. 8B, the following net name is used:
n31 = d•(a•b'•c'•d•e•f'•g•h')'

Decoded Bit E

The 'e' column of Table 2 has bold entries for D131, D161, D162, D146, D116, and D115. Table 16 lists the respective coded bits 'hgfedcba', the e-bit is overlined, and common patterns are marked by bold entries.

Table 16. E-bit Decoding

Name	hgfedcba	Coding Label	EK	Name	hgfedcba	Coding Label	EK
D131	$010\overline{1}1001$	$a \bullet b' \bullet c' \bullet d \bullet \overline{e} \bullet f' \bullet g \bullet h'$	00	D146	$011\overline{0}0110$	$a' \bullet b \bullet c \bullet d' \bullet \overline{e'} \bullet f \bullet g \bullet h'$	10
D161	**011**$\overline{1}$**00**01	$c' \bullet d' \bullet \overline{e} \bullet f \bullet g \bullet h' \bullet a \neq b$	00	D116	**010**$\overline{0}$**111**0	$c \bullet d \bullet \overline{e'} \bullet f' \bullet g \bullet h' \bullet a \neq b$	10
D162	**011**$\overline{1}$**00**10	$c' \bullet d' \bullet \overline{e} \bullet f \bullet g \bullet h' \bullet a \neq b$	00	D115	**010**$\overline{0}$**11**01	$c \bullet d \bullet \overline{e'} \bullet f' \bullet g \bullet h' \bullet a \neq b$	10

Using these identifiers, the encoding equation for bit 'e' can be written as follows:

$$E = e \bullet (a \bullet b' \bullet c' \bullet d \bullet \overline{e} \bullet f' \bullet g \bullet h' + c' \bullet d' \bullet \overline{e} \bullet f \bullet g \bullet h' \bullet a \oplus b)' +$$
$$a' \bullet b \bullet c \bullet d' \bullet \overline{e'} \bullet f \bullet g \bullet h' + c \bullet d \bullet \overline{e'} \bullet f' \bullet g \bullet h' \bullet a \oplus b$$

In the circuit diagram of FIG. 8B, the following net name is used:

n41 = e•n43'   n42 = a'•b•c•d'•e'•f•g•h' + c•d•e'•f'•g•h'•a⊕b

n43 = (a•b'•c'•d•e•f'•g•h' + c'•d'•e•f•g•h'•a⊕b)'

Decoded Bit F

The 'f' column of Table 2 has bold entries for D161, D162, D145, D151, D116, D115, D132, and D126. The respective coded bits 'hgfedcba' are listed in Table 17, the e-bit is overlined, and common patterns are marked by bold entries.

Table 17. F-bit Decoding

Name	hgfedcba	Coding Label	FK	Name	hgfedcba	Coding Label	FK
D161	**01**$\overline{\mathbf{1}}$**100**01	c'•d'•e•$\overline{f}$•g•h'•a≠b	00	D116	**01**$\overline{0}$**011**10	c•d•e'•$\overline{f'}$•g•h'•a≠b	10
D162	**01**$\overline{\mathbf{1}}$**100**10	c'•d'•e•$\overline{f}$•g•h'•a≠b	00	D115	**01**$\overline{0}$**011**01	c•d•e'•$\overline{f'}$•g•h'•a≠b	10
D145	**01**$\overline{\mathbf{1}}$**0**01**01**	a•b'•e'•$\overline{f}$•g•h'•c≠d	00	D132	**01**$\overline{0}$**1**10**10**	a'•b•e•$\overline{f'}$•g•h'•c≠d	10
D151	**01**$\overline{\mathbf{1}}$**0**10**01**	a•b'•e'•$\overline{f}$•g•h'•c≠d	00	D126	**01**$\overline{0}$**1**01**10**	a'•b•e•$\overline{f'}$•g•h'•c≠d	10

The decoding equation for bit 'F' can be written as follows:

$$F = f \bullet (c' \bullet d' \bullet e \bullet \overline{f} \bullet g \bullet h' \bullet a \oplus b + a \bullet b' \bullet e' \bullet \overline{f} \bullet g \bullet h' \bullet c \oplus d)' +$$
$$c \bullet d \bullet e' \bullet \overline{f'} \bullet g \bullet h' \bullet a \oplus b + a' \bullet b \bullet e \bullet \overline{f'} \bullet g \bullet h' \bullet c \oplus d$$

In the circuit diagram of FIG. 8B, the following net names are used:

n51 = f•n52'   n52 = c'•d'•e•f•g•h'•a⊕b + a•b'•e'•f•g•h'•c⊕d

Decoded Bit K

The 'K' column of Table 2 and 3 has a value of one for the four coded vectors K107, K125, K170, and K152. The respective coded bits 'hgfedcba' are listed in Table 18.

Table 18. K-bit Decoding

Name	hgfedcba	Coding Label	K	Name	hgfedcba	Coding Label	K
K107	**01000111**	a•c•d'•f'•g•h'•b≠e	1	K170	**01111000**	a'•c'•d•f•g•h'•b≠e	1
K125	**01010101**	a•c•d'•f'•g•h'•b≠e	1	K152	**01101010**	a'•c'•d•f•g•h'•b≠e	1

The decoding equation for bit 'K' can be written as follows:

$$K = a \bullet c \bullet d' \bullet f' \bullet g \bullet h' \bullet b \oplus e + a' \bullet c' \bullet d \bullet f \bullet g \bullet h' \bullet b \oplus e$$

In the circuit diagram of FIG. 8B, the following net name is used:

n60 = c'•d•g•h'•b⊕e

Validity Checks

Any received vector which does not fit the trellis of FIG. 2 is invalid. Since only 68 vectors out of the total of 256 8-bit vectors are valid, there are a total of 188 invalid vectors. The circuitry to verify that a vector belongs to the set of valid vectors is less complex than circuitry which flags invalid vectors. The validity checks can be derived directly from the trellis of FIG. 2 by using three sets of overlapping complementary circuits which pass through the center nodes labelled with the numbers 4, 6, and 4.

$$
\begin{aligned}
VALID = {} & (a\oplus b\bullet c\oplus d+b\oplus c\bullet a\oplus d)\bullet(e\oplus f\bullet g\oplus h+f\oplus g\bullet e\oplus h) + \\
& (c\bullet d\bullet a\oplus b+a\bullet b\bullet c\oplus d)\bullet(g'\bullet h'\bullet e\oplus f+e'\bullet f'\bullet g\oplus h) + \\
& (a\oplus b\bullet c'\bullet d'+a'\bullet b'\bullet c\oplus d)\bullet(g\bullet h\bullet e\oplus f+e\bullet f\bullet g\oplus h)
\end{aligned}
$$

In the circuit diagram of FIG. 8B, the following net names are used:

$n61 = a\oplus b\bullet c\oplus d+b\oplus c\bullet a\oplus d$	$n62 = e\oplus f\bullet g\oplus h+f\oplus g\bullet e\oplus h$
$n63 = c\bullet d\bullet a\oplus b+a\bullet b\bullet c\oplus d$	$n64 = g'\bullet h'\bullet e\oplus f+e'\bullet f'\bullet g\oplus h$
$n65 = c'\bullet d'\bullet a\oplus b+a'\bullet b'\bullet c\oplus d$	$n66 = g\bullet h\bullet e\oplus f+e\bullet f\bullet g\oplus h$
$n67 = n61\bullet n62$	$n68 = n63\bullet n64$
$n69 = n65\bullet n66$	

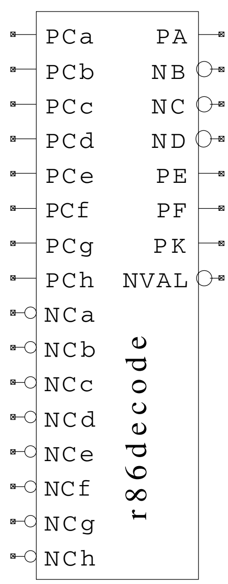


FIG. 8A

Circuit Implementation of the 8B/6B-P Decoder

FIG. 8A is a symbol of the decoder circuit showing the inputs and outputs. FIG. 8B is the logic diagram of the decoder circuit. It has not yet been simulated and verified by design tools, so it may be afflicted with minor errors.

Alternate Implementation of Decoder

As for encoding, all the decoding equations for the bits A through F, bit K, and the validity check have complementary features which can be exploited for simplification by the use of the Exclusive OR function. The transformed equations A through F, bit K, and VALID are presented here:

$$B = b\oplus(a\oplus\overline{b'}\bullet\overline{b}\oplus c\bullet c\oplus d'\bullet e\oplus f\bullet g\bullet h')$$

$$C=c\oplus(a\oplus b'\bullet b\oplus d'\bullet d\oplus e\bullet\oplus'\bullet g\bullet h')$$

$$D=d\oplus(a\oplus b\bullet b\oplus c'\bullet c\oplus\overline{d'}\bullet\overline{d}\oplus e'\bullet e\oplus f\bullet g\bullet h')$$

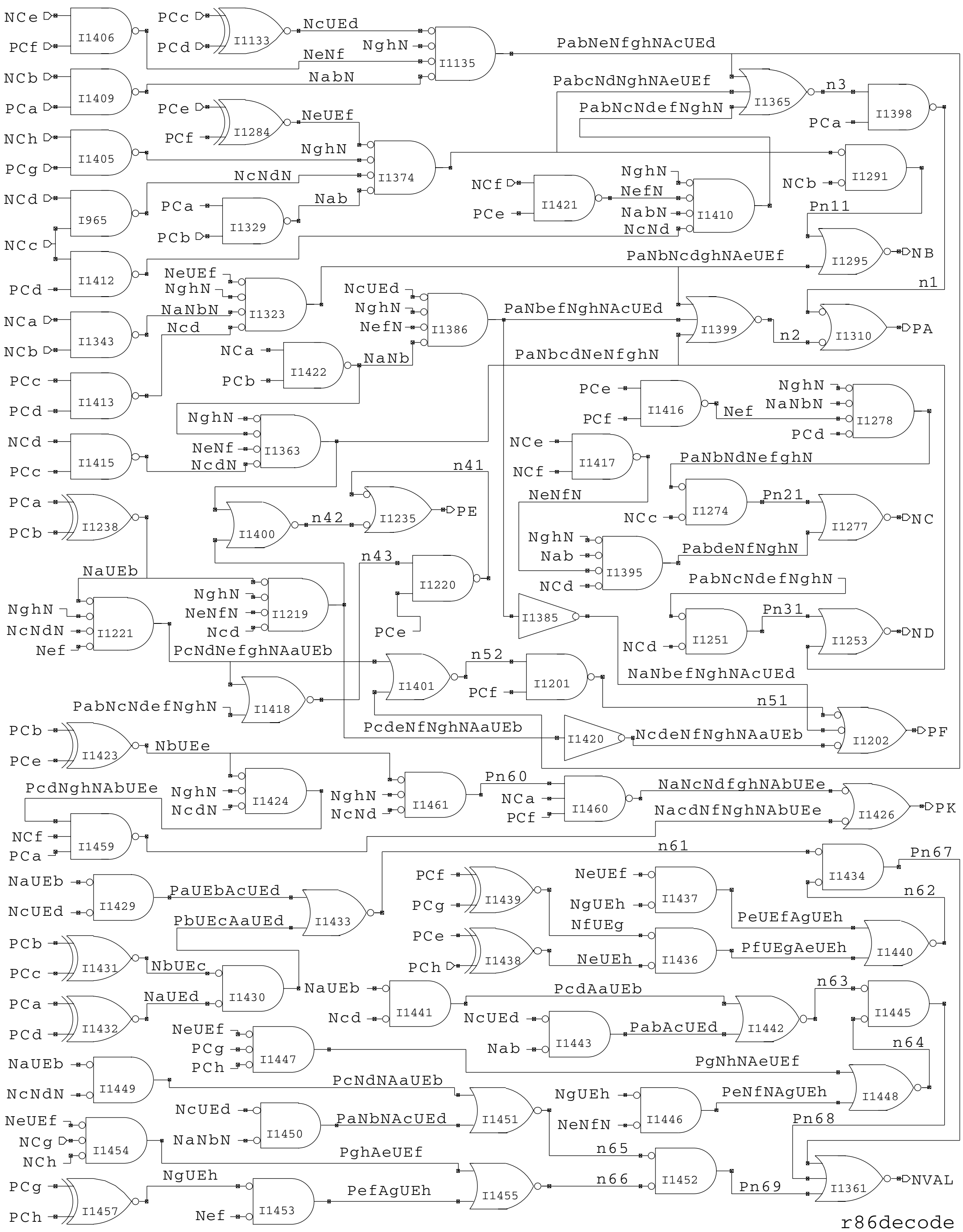
$$A=a\oplus[(e\oplus f\bullet g\bullet h')\bullet(\overline{a}\oplus b\bullet b\oplus c'\bullet c\oplus d\bullet d\oplus e'+\overline{a}\oplus b\bullet b\oplus e'\bullet c\oplus d+\overline{a}\oplus b'\bullet b\oplus c\bullet c\oplus d')]$$

$$E=e\oplus[(a\oplus b\bullet g\bullet h')\bullet(b\oplus c'\bullet c\oplus d\bullet d\oplus\overline{e'}\bullet\overline{e}\oplus f\bullet c\oplus d'\bullet d\oplus\overline{e}\bullet\overline{e}\oplus f')]$$

$$F=f\oplus[(a\oplus b\bullet g\bullet h')\bullet(c\oplus d'\bullet d\oplus e\bullet e\oplus\overline{f'}+b\oplus e'\bullet c\oplus d\bullet e\oplus\overline{f})]$$

$$K=d\oplus[(b\oplus e\bullet g\bullet h')\bullet(a\oplus c'\bullet c\oplus\bar{d}\bullet d\oplus f')]$$

$$
\begin{aligned}
VALID = {} & (a\oplus b\bullet c\oplus d+b\oplus c\bullet a\oplus d)\bullet(e\oplus f\bullet g\oplus h+f\oplus g\bullet e\oplus h) + \\
& (a\oplus b\bullet c\oplus d'+a\oplus b'\bullet c\oplus d)\bullet(e\oplus f\bullet g\oplus h'+e\oplus f'\bullet g\oplus h)\bullet(c\oplus g\bullet d\oplus h+a\oplus e\bullet b\oplus f)
\end{aligned}
$$



r86decode
10-29-03

FIG. 8B. 8B/6B DECODING CIRCUIT

Implementation Summary

An encoding circuit for the 6B/8-P code can be built with 69 Standard Primitive Logic Cells (2 x INV, 8 x NAND3, 20 x NAND2, 26 x NOR3, 10 x NOR2, 3 x XNOR2). Assuming complementary inputs, there are no more than five cells in any logic path. Similarly, the decoding circuit including the validity check requires no more than 78 cells (2 x INV, 3 x NAND3, 19 x NAND2, 10 x NOR4, 7 x NOR3, 28 x NOR2, 9 x XNOR2) and no more than five cells in any logic path.

References

1. IBM US Patent 5,699,062, Widmer, "DC Balanced (0,5) 8B/10B Transmission Code having Local Parity", Dec. 16, 1997.
2. IBM Research Report RC19106 (83280) 8/13/93, A. X. Widmer, "Error Correction Based on Transmission Code Violations and Parity".
3. IBM US Patent 5,740,186, Widmer, "Apparatus and Method for Error Correction Based on Transmission Code Violations and Parity", Apr. 14, 1998.
4. US Patent Application Serial No. 10\323,520, A. X. Widmer, "Error Correction with Low Latency for Bus Structures".
5. IBM US Patent 4,486,739, Franaszek and Widmer, "Byte Oriented DC Balanced (0,4) 8B/10B Partitioned Block Transmission Code", Dec. 4, 1984.
6. Petar Pepeljugoski, "Comparison of ISI penalty and sensitivity to low-frequency cut-off for 5B/6B, 6B/8B and 8B/10B codes, Internal Memo, dated Sept. 2003.

File Name: Code6B8BLPAR_RC