1004-B (2-54)

Technical Memorandum

54-57-51

A BINARY-TO-DECIMAL CONVERTER

by

DISCLAIMER

SC 1004-A (3-55)

AECU-3472

Technical Memorandum

54-57-51

A BINARY-TO-DECIMAL CONVERTER

by

Samuel D. Stearns, 5112-2

ABSTRACT

A system of standard electronic logical circuits which stores a seven-bit binary number, converts the binary number to its decimal equivalent, and then indicates the decimal equivalent in direct-reading neon lights is described.

Case No. 408.02

September 1, 1956

DISTRIBUTION:
G. A. Fowler, 5000
L. A. Hopkins, Jr., 1400
S. C. Hight, 5100
R. A. Bice, 5200
T. S. Church, 1410
T. F. Marker, 1420
M. L. Merritt, 5110
A. P. Gruer, 5210
D. B. Shuster, 5230
W. T. Moffat, 5240
J. D. Shreve, Jr., 5112 (2)
K. E. Ball, 5126
A. E. Bentz, 5213
R. C. Spence, 5222
B. W. Washburn, 5243
R. D. Jones, 5112-2
S. D. Stearns, 5112-2 (6)
Dr. R. K. Moore, Chairman, Department of Electrical Engineering,
University of New Mexico (4)
R. K. Smeltzer, 7221-3
A. D. Pepmueller, 7223 (6)

A BINARY-TO-DECIMAL CONVERTER

CHAPTER 1. INTRODUCTION AND SUMMARY DESCRIPTION

Introduction

This paper describes a device which electronically converts a binary number into its decimal equivalent. Such a device might be of use in digital computers, many of which use the binary number system, and also in similar machines, such as pulse code modulated data collectors.

The binary-to-decimal converter described here was designed specifically for use in Sandia Corporation's High-Speed Digital Recording and Data-Handling System, which actually uses the Gray code rather than the binary number system. However, conversion from Gray to binary (see Appendix B) is simple, and a binary-to-decimal converter seems much simpler than a Gray-to-decimal converter.

The converter is designed to have a parallel input of 7 binary bits, representing any number from 0 to 127. Extension of the converter to more bits is easy. Its output is presented on neon lights, representing the decimal equivalent of the input as shown in Fig. 1.1. The output is in lights because this converter is used as an indicating device; however, the output voltages could also be used to drive other memory or recording systems.

The converter is designed to make conversions at any rate from 0 to about 100 conversions/sec. When used in the Sandia Corporation data system mentioned previously, it will run at about 25 conversions/sec.

It seems reasonable that a design at least similar to the one used here might already have been built. However, a search of all available literature has not revealed any similar device either proposed or in use.

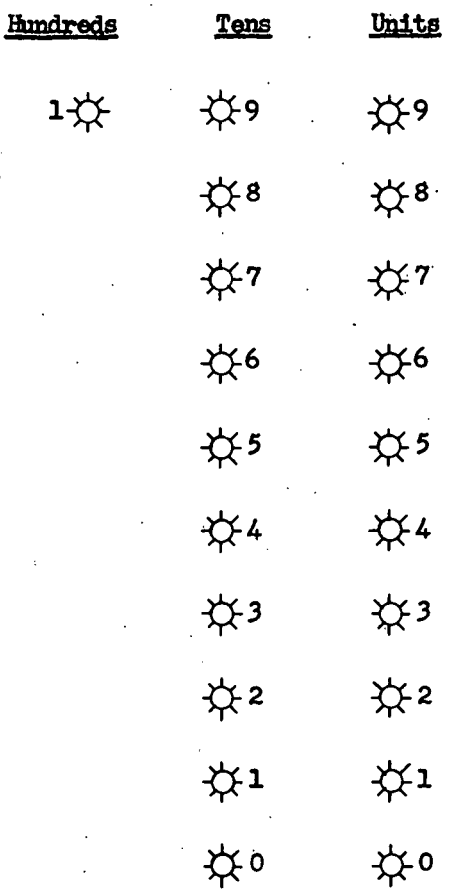


Fig. 1.1 — Output of Converter is in Lights Which Represent Decimal Equivalent of Input

Following this introduction is a summary and general description of the converter, designed to give the reader an idea of how it works in a minimum of reading time.

Following the summary is an account of some of the methods for binary-to-decimal conversion, to show why the particular design used was chosen. Finally, there is a detailed description of the circuit operation of the converter.

Summary Description

In order to change a 7-bit binary number to its decimal equivalent, this converter first changes the binary number (N) to its two's complement (2^n-N, where n = 7 in this case). The converter then stores the two's complement in seven flip-flop stages.

After the two's complement of N, or 128-N, has been stored, an oscillator begins to increase this quantity in storage, one unit at a time. The oscillator continues to increase the number in storage until the number reaches 128; then the oscillator stops. In order to increase the number in storage to 128, the oscillator has put out a number of pulses equal to the difference between 128 and the two's complement of N, or

$$128 - (128-N), \text{ or } N$$

Thus the oscillator puts out a number of pulses exactly equal to the number represented by the binary input to the converter. These pulses from the oscillator are also fed into a decade scaler which is capable of counting up to 127. The decade scaler then displays the number N in decimal form on direct-reading neon lights.

The entire converter (less power supply) uses 21 vacuum tubes, including twenty 5963 twin triodes and one 6CL6 pentode. Power requirements are as follows:

B+ — 250-300 v dc	at 80 ma
Fil. — 6.3 v ac	at 6.65 a

While, as mentioned, this converter was built specifically for use with the Sandia Corporation High Speed Digital Recording and Data Handling System, it may also be used for output and display purposes in digital computers which use the binary system. A photograph of the converter is shown in Fig. 1.2.

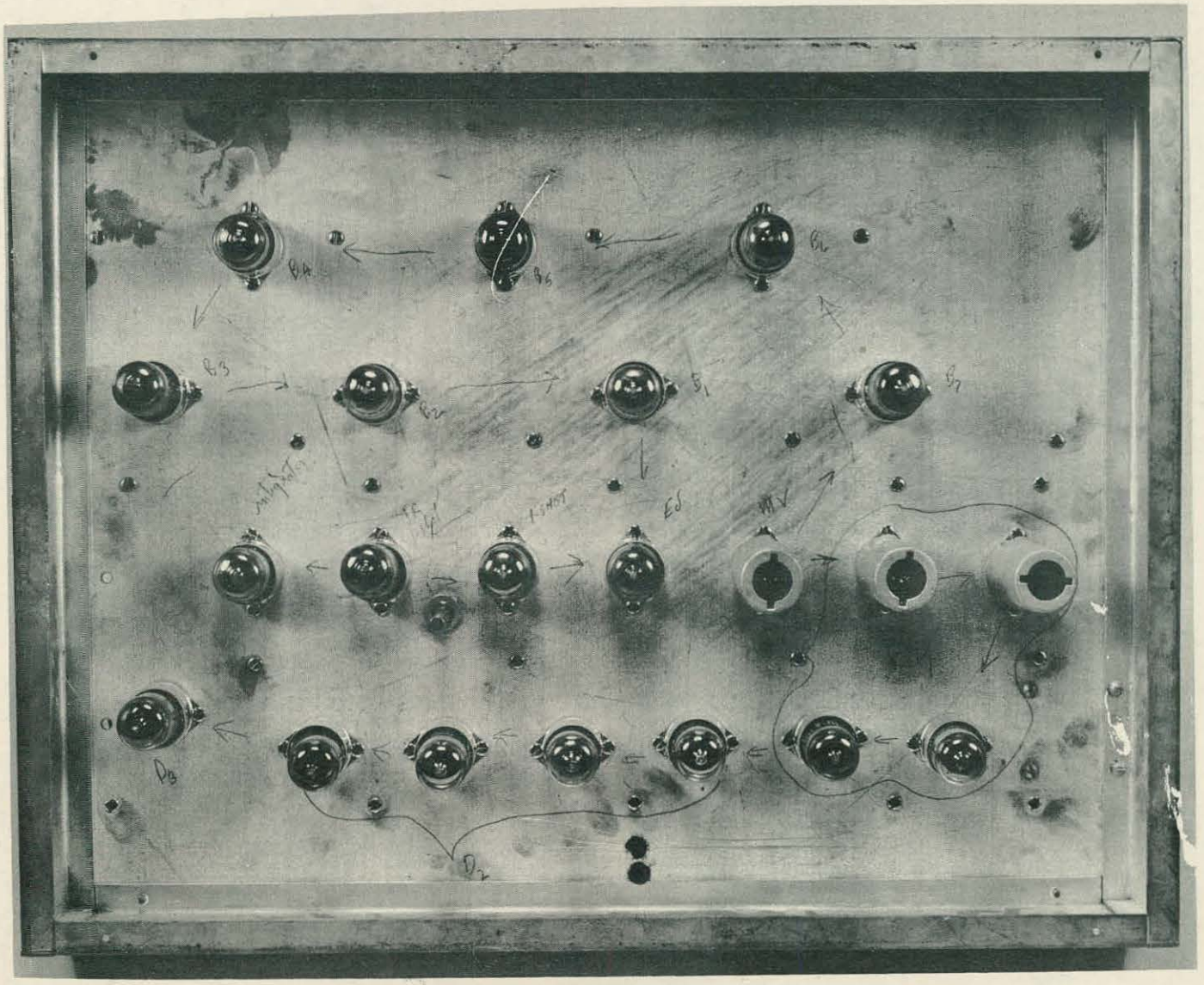


Fig. 1.2 — A Binary-to-Decimal Converter

CHAPTER 2. METHODS FOR BINARY-TO-DECIMAL CONVERSION

The problem of finding the simplest and best circuit to perform even a moderately complicated logical function, such as binary-to-decimal conversion, is usually not simple or straightforward. The logical function can be described precisely in Boolean algebra,[1] but the Boolean expressions can be manipulated in various ways, and many different circuits can be built which will satisfy a single Boolean expression. Nevertheless, it is helpful to express the function in Boolean algebra first, and then to investigate various ways of mechanizing the function.

1. For an explanation of Boolean algebra, see Richards, R. K., Arithmetic Operations in Digital Computers, D. Van Nostrand Company, 1955.

Method 1

This binary-to-decimal converter must accept an input (parallel in time) of a 7-bit binary number N, which will be denoted by

$$N = B_1B_2B_3B_4B_5B_6B_7 \quad ,$$

where B_1 is the most significant bit, having weight 64, and B_7 is the least significant bit, having weight 1. Thus, if

$$N = 1010100 \quad ,$$

where B_1=1, B_2=0, B_3=1, B_4=0, B_5=1, B_6=0, and B_7=0, then N represents the decimal number 84.

Likewise, the output will be denoted by

$$N = D_1D_2D_3 \quad ,$$

where D_1 is the most significant decimal digit and D_3 the least significant decimal digit.

To express the function of the converter, the output must be written as a Boolean function of the input. But Boolean expressions can contain only variables with two possible values, so first the output must be re-expressed. To do this, the double-subscript notation is used for the D's, the first subscript denoting the decade and the second subscript the unit within the decade. Thus the expression

$$N = D_{10}D_{28}D_{34}$$

means that N has the value 084. Note also that, for instance, if D_{34} = 1, then D_{30}=D_{31}=D_{32}=D_{33}=D_{35}=D_{36}=D_{37}=D_{38}=D_{39}=0 necessarily. Furthermore, D_{12}, D_{13}, D_{14}, D_{15}, D_{16}, D_{17}, D_{18}, and D_{19} are always 0, because it is impossible to have an input N greater than 127.

The Boolean equations for the D's as functions of the B's will now be given and simplified as much as possible. D_1 will be expressed first, then D_2, and finally D_3. These equations may be derived by inspection of Table 2.1.

Table 2.1 — BINARY AND DECIMAL NUMBERS

Decimal	Binary (B 1 2 3 4 5 6 7)	Decimal
0	0 0 0 0 0 0 0	0
1	0 0 0 0 0 0 1	1
2	0 0 0 0 0 1 0	2
3	0 0 0 0 0 1 1	3
4	0 0 0 0 1 0 0	4
5	0 0 0 0 1 0 1	5
6	0 0 0 0 1 1 0	6
7	0 0 0 0 1 1 1	7
8	0 0 0 1 0 0 0	8
9	0 0 0 1 0 0 1	9
10	0 0 0 1 0 1 0	10
11	0 0 0 1 0 1 1	11
12	0 0 0 1 1 0 0	12
13	0 0 0 1 1 0 1	13
14	0 0 0 1 1 1 0	14
15	0 0 0 1 1 1 1	15
16	0 0 1 0 0 0 0	16
17	0 0 1 0 0 0 1	17
18	0 0 1 0 0 1 0	18
19	0 0 1 0 0 1 1	19
20	0 0 1 0 1 0 0	20
21	0 0 1 0 1 0 1	21
22	0 0 1 0 1 1 0	22
23	0 0 1 0 1 1 1	23
24	0 0 1 1 0 0 0	24
25	0 0 1 1 0 0 1	25
26	0 0 1 1 0 1 0	26
27	0 0 1 1 0 1 1	27
28	0 0 1 1 1 0 0	28
29	0 0 1 1 1 0 1	29
30	0 0 1 1 1 1 0	30
31	0 0 1 1 1 1 1	31
32	0 1 0 0 0 0 0	32
33	0 1 0 0 0 0 1	33
34	0 1 0 0 0 1 0	34
35	0 1 0 0 0 1 1	35
36	0 1 0 0 1 0 0	36
37	0 1 0 0 1 0 1	37
38	0 1 0 0 1 1 0	38
39	0 1 0 0 1 1 1	39
40	0 1 0 1 0 0 0	40
41	0 1 0 1 0 0 1	41
42	0 1 0 1 0 1 0	42
43	0 1 0 1 0 1 1	43
44	0 1 0 1 1 0 0	44
45	0 1 0 1 1 0 1	45
46	0 1 0 1 1 1 0	46
47	0 1 0 1 1 1 1	47
48	0 1 1 0 0 0 0	48
49	0 1 1 0 0 0 1	49
50	0 1 1 0 0 1 0	50
51	0 1 1 0 0 1 1	51
52	0 1 1 0 1 0 0	52
53	0 1 1 0 1 0 1	53
54	0 1 1 0 1 1 0	54
55	0 1 1 0 1 1 1	55
56	0 1 1 1 0 0 0	56
57	0 1 1 1 0 0 1	57
58	0 1 1 1 0 1 0	58
59	0 1 1 1 0 1 1	59
60	0 1 1 1 1 0 0	60
61	0 1 1 1 1 0 1	61
62	0 1 1 1 1 1 0	62
63	0 1 1 1 1 1 1	63
64	1 0 0 0 0 0 0	64
65	1 0 0 0 0 0 1	65
66	1 0 0 0 0 1 0	66
67	1 0 0 0 0 1 1	67
68	1 0 0 0 1 0 0	68
69	1 0 0 0 1 0 1	69
70	1 0 0 0 1 1 0	70
71	1 0 0 0 1 1 1	71
72	1 0 0 1 0 0 0	72
73	1 0 0 1 0 0 1	73
74	1 0 0 1 0 1 0	74
75	1 0 0 1 0 1 1	75
76	1 0 0 1 1 0 0	76
77	1 0 0 1 1 0 1	77
78	1 0 0 1 1 1 0	78
79	1 0 0 1 1 1 1	79
80	1 0 1 0 0 0 0	80
81	1 0 1 0 0 0 1	81
82	1 0 1 0 0 1 0	82
83	1 0 1 0 0 1 1	83
84	1 0 1 0 1 0 0	84
85	1 0 1 0 1 0 1	85
86	1 0 1 0 1 1 0	86
87	1 0 1 0 1 1 1	87
88	1 0 1 1 0 0 0	88
89	1 0 1 1 0 0 1	89
90	1 0 1 1 0 1 0	90
91	1 0 1 1 0 1 1	91
92	1 0 1 1 1 0 0	92
93	1 0 1 1 1 0 1	93
94	1 0 1 1 1 1 0	94
95	1 0 1 1 1 1 1	95
96	1 1 0 0 0 0 0	96
97	1 1 0 0 0 0 1	97
98	1 1 0 0 0 1 0	98
99	1 1 0 0 0 1 1	99
100	1 1 0 0 1 0 0	100
101	1 1 0 0 1 0 1	101
102	1 1 0 0 1 1 0	102
103	1 1 0 0 1 1 1	103
104	1 1 0 1 0 0 0	104
105	1 1 0 1 0 0 1	105
106	1 1 0 1 0 1 0	106
107	1 1 0 1 0 1 1	107
108	1 1 0 1 1 0 0	108
109	1 1 0 1 1 0 1	109
110	1 1 0 1 1 1 0	110
111	1 1 0 1 1 1 1	111
112	1 1 1 0 0 0 0	112
113	1 1 1 0 0 0 1	113
114	1 1 1 0 0 1 0	114
115	1 1 1 0 0 1 1	115
116	1 1 1 0 1 0 0	116
117	1 1 1 0 1 0 1	117
118	1 1 1 0 1 1 0	118
119	1 1 1 0 1 1 1	119
120	1 1 1 1 0 0 0	120
121	1 1 1 1 0 0 1	121
122	1 1 1 1 0 1 0	122
123	1 1 1 1 0 1 1	123
124	1 1 1 1 1 0 0	124
125	1 1 1 1 1 0 1	125
126	1 1 1 1 1 1 0	126
127	1 1 1 1 1 1 1	127

1. D_1, the "hundreds" digit.

$$D_{10} = \bar{B}_1 + B_1\bar{B}_2 + B_1B_2\bar{B}_3\bar{B}_4\bar{B}_5$$
$$= \bar{B}_1 + \bar{B}_2 + \bar{B}_3\bar{B}_4\bar{B}_5 \qquad (2.1)$$

$$D_{11} = B_1B_2B_3 + B_1B_2\bar{B}_3B_4 + B_1B_2\bar{B}_3\bar{B}_4B_5$$
$$= B_1B_2(B_3 + \bar{B}_3B_4 + \bar{B}_3B_5)$$
$$= B_1B_2(B_3 + B_4 + B_5) = \bar{D}_{10} \qquad (2.2)$$

2. D_2, the "tens" digit.

$$D_{20} = \bar{B}_1\bar{B}_2\bar{B}_3(\bar{B}_4 + \bar{B}_5\bar{B}_6) + B_1B_2\bar{B}_3(\bar{B}_4B_5 + B_4\bar{B}_5 + B_4B_5\bar{B}_6)$$
$$= \bar{B}_1\bar{B}_2\bar{B}_3(\bar{B}_4 + \bar{B}_5\bar{B}_6) + B_1B_2\bar{B}_3(\bar{B}_4B_5 + B_4\bar{B}_5 + B_4\bar{B}_6) \qquad (2.3)$$

$$D_{21} = \bar{B}_1\bar{B}_2(\bar{B}_3B_4B_5 + \bar{B}_3B_4B_6 + B_3\bar{B}_4\bar{B}_5) + B_1B_2(\bar{B}_3B_4B_5B_6 + B_3\bar{B}_4)$$
$$= \bar{B}_1\bar{B}_2\left[\bar{B}_3B_4(B_5 + B_6) + B_3\bar{B}_4\bar{B}_5\right] + B_1B_2(\bar{B}_3B_4B_5B_6 + B_3\bar{B}_4) \qquad (2.4)$$

$$D_{22} = \bar{B}_1\bar{B}_2B_3\bar{B}_4B_5 + B_3B_4(\bar{B}_5 + B_5\bar{B}_6) + B_1B_2B_3B_4$$
$$= \bar{B}_1\bar{B}_2B_3(\bar{B}_4B_5 + B_4\bar{B}_5 + B_4\bar{B}_6) + B_1B_2B_3B_4 \qquad (2.5)$$

$$D_{23} = \bar{B}_1(\bar{B}_2B_3B_4B_5B_6 + B_2\bar{B}_3\bar{B}_4) \qquad (2.6)$$

$$D_{24} = \bar{B}_1B_2(\bar{B}_3B_4 + B_3\bar{B}_4\bar{B}_5\bar{B}_6) \qquad (2.7)$$

$$D_{25} = \bar{B}_1B_2B_3\bar{B}_4(\bar{B}_5B_6 + B_5) + B_4\bar{B}_5$$
$$= \bar{B}_1B_2B_3\left[\bar{B}_4(B_5 + B_6) + B_4\bar{B}_5\right] \qquad (2.8)$$

$$D_{26} = \bar{B}_1B_2B_3B_4B_5 + B_1\bar{B}_2\bar{B}_3\bar{B}_4(\bar{B}_5 + B_5\bar{B}_6)$$
$$= \bar{B}_1B_2B_3B_4B_5 + B_1\bar{B}_2\bar{B}_3\bar{B}_4(\bar{B}_5 + \bar{B}_6) \qquad (2.9)$$

$$D_{27} = B_1\bar{B}_2\bar{B}_3(B_4 + B_5B_6) \qquad (2.10)$$

$$D_{28} = B_1\bar{B}_2B_3(\bar{B}_4 + \bar{B}_5\bar{B}_6) \qquad (2.11)$$

$$D_{29} = B_1\left[\bar{B}_2B_3B_4(\bar{B}_5B_6+B_5)+B_2\bar{B}_3\bar{B}_4\bar{B}_5\right]$$
$$= B_1\left[\bar{B}_2B_3B_4(B_5+B_6)+B_2\bar{B}_3\bar{B}_4\bar{B}_5\right] \qquad (2.12)$$

Before writing the D_3 equations, it should be noted here that so far the conversion is not too cumbersome, since the twelve equations above can be mechanized with about 141 diodes and 7 triodes. The situation for D_3 is different.

3. D_3, the "units" digit.

$$D_{30} = \bar{B}_7\left[\bar{B}_1\bar{B}_2\bar{B}_3\bar{B}_4\bar{B}_5\bar{B}_6+\bar{B}_1\bar{B}_2\bar{B}_3B_4\bar{B}_5\bar{B}_6+\bar{B}_1\bar{B}_2B_3\bar{B}_4B_5\bar{B}_6\right.$$
$$+\bar{B}_1\bar{B}_2B_3B_4B_5B_6+\bar{B}_1B_2\bar{B}_3B_4\bar{B}_5\bar{B}_6+\bar{B}_1B_2B_3\bar{B}_4\bar{B}_5B_6$$
$$+\bar{B}_1B_2B_3B_4B_5\bar{B}_6+B_1\bar{B}_2\bar{B}_3\bar{B}_4B_5\bar{B}_6+B_1\bar{B}_2B_3\bar{B}_4\bar{B}_5\bar{B}_6$$
$$+B_1\bar{B}_2B_3B_4\bar{B}_5B_6+B_1B_2\bar{B}_3\bar{B}_4B_5\bar{B}_6+B_1B_2\bar{B}_3B_4B_5B_6$$
$$\left.+B_1B_2B_3B_4\bar{B}_5\bar{B}_6\right] \qquad (2.13)$$

$$D_{31} = B_7\left[\text{Same as inside brackets above}\right] \qquad (2.14)$$

$$D_{32} = \bar{B}_7\left[\bar{B}_1\bar{B}_2\bar{B}_3\bar{B}_4\bar{B}_5\bar{B}_6+\bar{B}_1\bar{B}_2\bar{B}_3B_4\bar{B}_5\bar{B}_6+\bar{B}_1\bar{B}_2B_3\bar{B}_4B_5\bar{B}_6\right.$$
$$+\bar{B}_1\bar{B}_2B_3B_4B_5B_6+\bar{B}_1B_2\bar{B}_3B_4\bar{B}_5\bar{B}_6+\bar{B}_1B_2B_3\bar{B}_4B_5B_6$$
$$+\bar{B}_1B_2B_3B_4B_5\bar{B}_6+B_1\bar{B}_2\bar{B}_3\bar{B}_4B_5\bar{B}_6+B_1\bar{B}_2B_3\bar{B}_4\bar{B}_5\bar{B}_6$$
$$+B_1\bar{B}_2B_3B_4\bar{B}_5B_6+B_1B_2\bar{B}_3\bar{B}_4B_5\bar{B}_6+B_1B_2\bar{B}_3B_4B_5B_6$$
$$\left.+B_1B_2B_3B_4\bar{B}_5\bar{B}_6\right] \qquad (2.15)$$

$$D_{33} = B_7\left[\text{Same as inside brackets above}\right] \qquad (2.16)$$

- - - - -

Without writing the rest of the D_3 equations, the results can be evaluated. If all of the D_3 equations are written, there will be a total of 128 terms in all 10 equations (13 terms in each of the equations for $D_{30} - D_{37}$ and 12 in the equations for D_{38} and D_{39}). None of the equations can be reduced, as may be suspected by observing the first four. Therefore, to mechanize the D_3 equations, all 128 terms could be generated in one of the standard "and-tree" matrices, such as the one shown in Fig. 2.1.

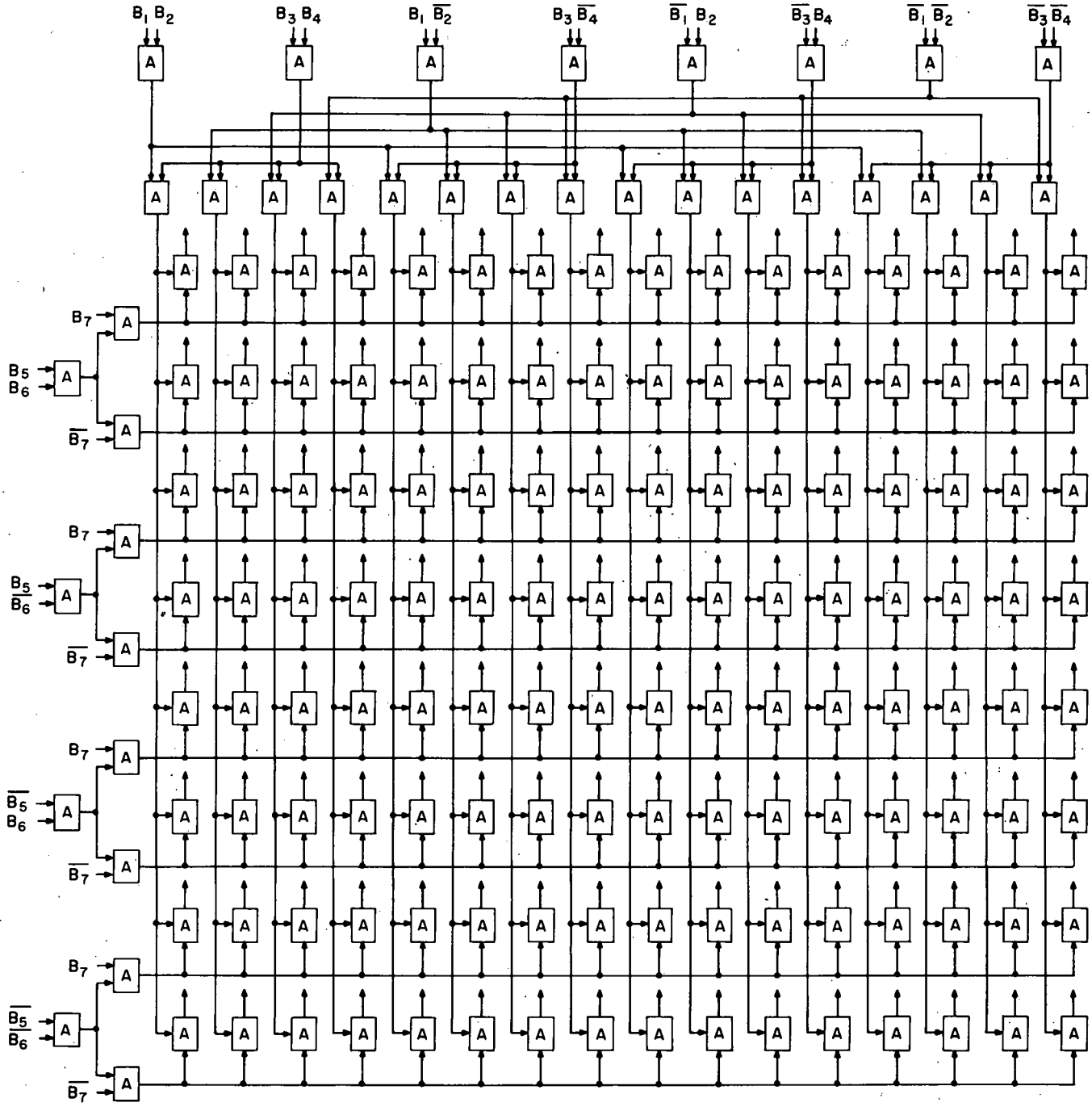
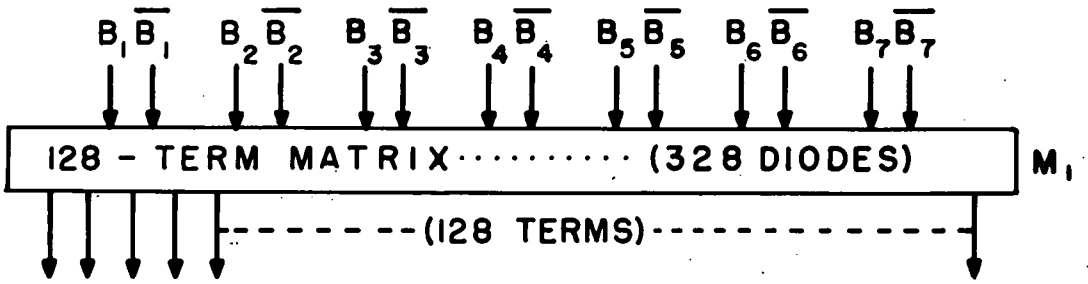


Fig. 2.1 — Tree Matrix Which Will Generate the 128 Terms for the D_3 Equations

040 011

A matrix such as the one in Fig. 2.1 would take about 328 diodes, plus the 7 triodes necessary to give the inverted inputs. Since the outputs of this matrix must be combined further, the matrix is denoted as follows:



The outputs of M_1 can now be combined in "or" circuits to form the ten D_3 outputs. Also, since all of the 128 elemental forms have been generated in M_1, the ten D_2 outputs could be generated using the outputs of M_1. To generate the D_2 outputs would take 128 additional diodes, compared with about 141 diodes to generate separately the D_2 outputs (see Eqs 2.3 through 2.12). It is evidently better to use the M_1 outputs to generate both the D_3 and the D_2 outputs.

The D_1 outputs could of course also be formed using the outputs of M_1, but obviously Eqs 2.1 and 2.2 are much easier to mechanize.

The complete form of this type of binary-decimal converter is now shown in Fig. 2.2. The ten D_2 and the ten D_3 outputs are generated using the M_1 outputs, and D_1 is generated separately.

Altogether, the converter shown in Fig. 2.2 would take 590 diodes and 8 triodes, not counting extra devices needed for clamping, amplification, etc. Although expensive and complex, it could operate rapidly and all 22 decimal outputs would be completely isolated.

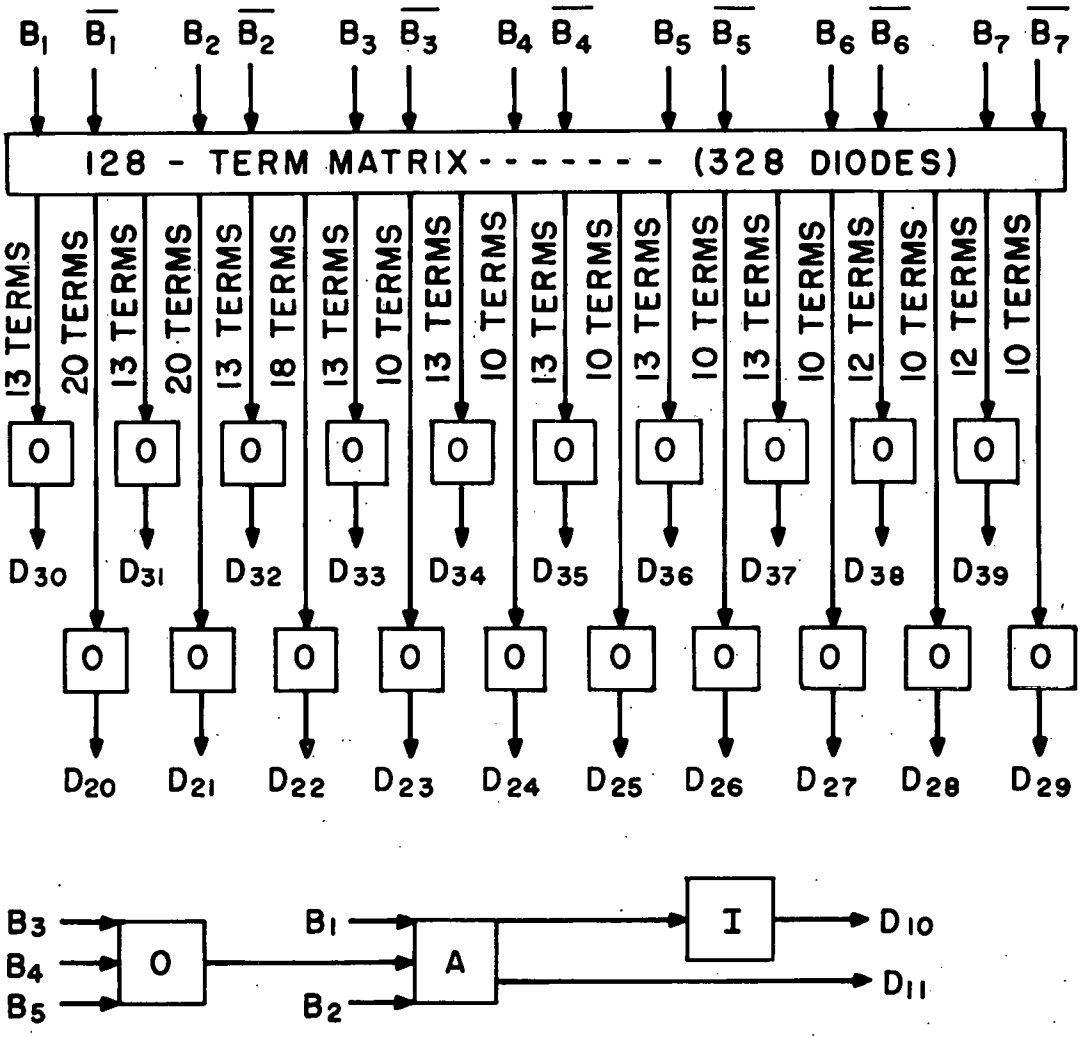


Fig. 2.2 — One Type of Binary-to-Decimal Converter

Method 2

Two special criteria for the binary-decimal converter needed by Sandia Corporation suggest other approaches to the problem. First, the converter will be used as an indicating device with the output displayed on lights. Secondly, the necessary conversion rate here is quite low, being 25 conversions/sec. The first criterion suggests that the output of the converter might be in some decimal code, rather than on 22 separate lines. This is essentially the difference between Method 2 and Method 1. The second criterion will be disregarded for the time being.

A large number of decimal codes is available.[2] All of the simplest codes use four binary bits to represent a single decimal digit, four bits being the minimum number for representing a digit having ten possible values. In different codes, each of the four bits has a different weight assigned to it, or no weight at all. Examples of some different four-bit codes are given in Table 2.2. The 8421 code is a straight binary representation of the decimal digit. All other four-bit codes simply use ten unique combinations of ones and zeros to represent the ten decimal digits.

Table 2.2 — FOUR-BIT DECIMAL CODES

Decimal Number	8421 Code $b_1b_2b_3b_4$	6311 Code $b_1b_2b_3b_4$	Nonweighted (Excess-3) Code $b_1b_2b_3b_4$
0	0 0 0 0	0 0 0 0	0 0 1 1
1	0 0 0 1	0 0 0 1	0 1 0 0
2	0 0 1 0	0 0 1 1	0 1 0 1
3	0 0 1 1	0 1 0 0	0 1 1 0
4	0 1 0 0	0 1 0 1	0 1 1 1
5	0 1 0 1	0 1 1 1	1 0 0 0
6	0 1 1 0	1 0 0 0	1 0 0 1
7	0 1 1 1	1 0 0 1	1 0 1 0
8	1 0 0 0	1 0 1 1	1 0 1 1
9	1 0 0 1	1 1 0 0	1 1 0 0

2. Ibid.

If the converter output consisted of decimal digits in some four-bit code, straight decimal indication could be achieved with the use of neon lights and resistors in a permutating network. A permutating network for the 8421 code is shown in Fig. 2.3. In this example, the numbered neon lights in the center will fire at 70 volts, and any "b" input is either 60 or 140 volts depending on whether it is 0 or 1. The inverse bits, or "$\bar{b}$"'s, are available here also, since each bit is assumed to be stored in a flip-flop stage, where either plate voltage may be used. Note that a given lamp will not light unless the proper b_4 bus is at +140 volts and all inputs to the other side of the lamp are at 60 volts. Referring to Table 2.2, it can be seen how the decimal indication of the 8421 code is achieved in Fig. 2.3. The permutation of any other decimal code is just as simple and would require about the same number of components.

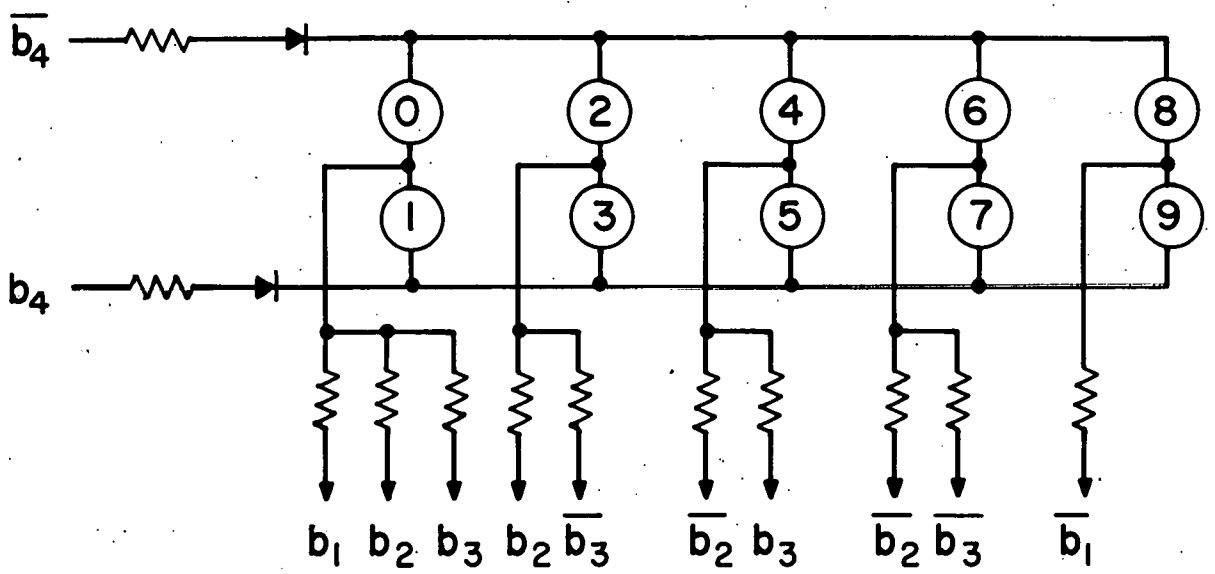


Fig. 2.3 — Permutating Network for 8421 Code

Thus, if it were easier to convert from binary to some decimal code than to straight decimal as in Method 1, a saving in components could probably be achieved, since the permutating network is quite simple.

Of the many four-bit codes, the excess-3 code was chosen for use as an example here, not because it has any unique properties, but because a binary-to-excess-3 converter has already

been built with relays.[3] In looking at other four-bit codes, it appears that conversion from binary to other codes is no simpler than conversion to excess-3. However, no proof of this fact is known to the author, and the existence of an easier-to-convert-to code is possible.

The excess-3 code is shown in Table 2.2. Since three decimal digits must be represented by this code at the converter output, nine bits must be generated. The output will be of the form

$$e_1 \quad e_2e_3e_4e_5 \quad e_6e_7e_8e_9 \quad ,$$

where e_1 represents the hundreds digit (can only be 0 or 1), $e_2e_3e_4e_5$ represents the tens digit, and $e_6e_7e_8e_9$ represents the units digit. A permutating network for the tens digit is shown in Fig. 2.4. Operation is similar to the 8421 code permutating network already described.

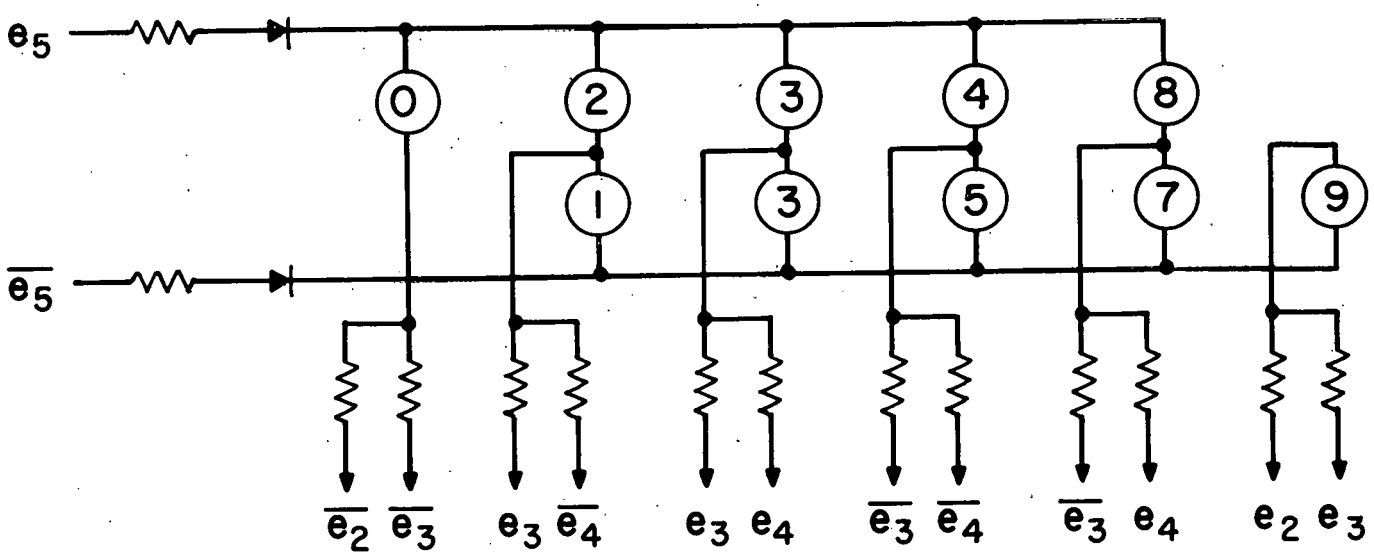


Fig. 2.4 — Permutating Network for Excess-3 Code (Tens Digit)

For conversion from binary to excess-3, the Boolean equations may be examined for possible means of mechanization. This has been done, and it has been found[4] that the simplest way to convert is to generate all 12 decades on separate lines, and then generate most of the e's

3. Robnett, A. V. and Vulgan, E. J., Small-Scale Automatic Field-Data Reduction System, Sandia Corporation TM-236-55-51, October 3, 1955.

4. Ibid.

using these decade lines. Letting Δ_1, Δ_2, - - - Δ_{12} be the twelve decades, and letting $B_1B_2B_3B_4B_5B_6B_7$ be the seven-bit binary input, the Boolean expressions for the e's are as follows:

$$e_1 = B_1B_2 (B_3 + B_4 + B_5) \tag{2.17}$$

$$e_2 = \Delta_5 + \Delta_6 + \Delta_7 + \Delta_8 + \Delta_9 \tag{2.18}$$

$$e_3 = \Delta_1 + \Delta_2 + \Delta_3 + \Delta_4 + \Delta_9 + \Delta_{11} + \Delta_{12} \tag{2.19}$$

$$e_4 = \Delta_0 + \Delta_3 + \Delta_4 + \Delta_7 + \Delta_8 + \Delta_{10} \tag{2.20}$$

$$e_5 = \Delta_0 + \Delta_2 + \Delta_4 + \Delta_6 + \Delta_8 + \Delta_{10} + \Delta_{12} \tag{2.21}$$

$$\begin{aligned} e_6 = {} & e_7\bar{e}_8\bar{e}_9 \left[B_4 (\Delta_0 + \Delta_2 + \Delta_5 + \Delta_7 + \Delta_8 + \Delta_{10}) + \bar{B}_4 (\Delta_1 + \right. \\ & \left. \Delta_3 + \Delta_4 + \Delta_6 + \Delta_9 + \Delta_{11} + \Delta_{12})\right] + \\ & \bar{e}_7 \left[\bar{e}_8 + \bar{e}_9 + B_4 (\Delta_0 + \Delta_2 + \Delta_5 + \Delta_7 + \Delta_8 + \Delta_{10}) + \bar{B}_4 (\Delta_1 + \right. \\ & \left. \Delta_3 + \Delta_4 + \Delta_6 + \Delta_9 + \Delta_{11} + \Delta_{12})\right] \end{aligned} \tag{2.22}$$

$$\begin{aligned} e_7 = {} & (\Delta_0 + \Delta_4 + \Delta_8 + \Delta_{12}) \left[B_5\bar{B}_6\bar{B}_7 + \bar{B}_5 (B_6 + B_7)\right] + \\ & (\Delta_2 + \Delta_6 + \Delta_{10}) \left[\bar{B}_5\bar{B}_6\bar{B}_7 + B_5 (B_6 + B_7)\right] + \\ & (\Delta_1 + \Delta_5 + \Delta_9) \left[B_5 (\bar{B}_6 + \bar{B}_7) + \bar{B}_5B_6B_7\right] + \\ & (\Delta_3 + \Delta_7 + \Delta_{11}) \left[\bar{B}_5 (\bar{B}_6 + \bar{B}_7) + B_5B_6B_7\right] \end{aligned} \tag{2.23}$$

$$\begin{aligned} e_8 = {} & (\Delta_1 + \Delta_3 + \Delta_5 + \Delta_7 + \Delta_9 + \Delta_{11}) (B_6\bar{B}_7 + \bar{B}_6B_7) + \\ & (\Delta_0 + \Delta_2 + \Delta_4 + \Delta_6 + \Delta_8 + \Delta_{10} + \Delta_{12}) (B_6B_7 + \bar{B}_6\bar{B}_7) \end{aligned} \tag{2.24}$$

$$e_9 = \bar{B}_7 \tag{2.25}$$

The separate generation of Δ_1 through Δ_{12} is of course also necessary, and so the following equations must also be mechanized:

$$\Delta_0 = \bar{B}_1\bar{B}_2\bar{B}_3 (\bar{B}_4 + \bar{B}_5\bar{B}_6) \tag{2.26}$$

040 017

$$\Delta_1 = \bar{B}_1\bar{B}_2\left[\bar{B}_3B_4\ (B_5 + B_6) + B_3\bar{B}_4\bar{B}_5\right] \tag{2.27}$$

$$\Delta_2 = \bar{B}_1\bar{B}_2B_3\left[\bar{B}_4B_5 + B_4\ (\bar{B}_5 + \bar{B}_6)\right] \tag{2.28}$$

$$\Delta_3 = \bar{B}_1\ (\bar{B}_2B_3B_4B_5B_6 + B_2\bar{B}_3\bar{B}_4) \tag{2.29}$$

$$\Delta_4 = \bar{B}_1B_2\ (\bar{B}_3B_4 + B_3\bar{B}_4\bar{B}_5\bar{B}_6) \tag{2.30}$$

$$\Delta_5 = \bar{B}_1B_2B_3\left[\bar{B}_4\ (B_5 + B_6) + B_4\bar{B}_5\right] \tag{2.31}$$

$$\Delta_6 = \bar{B}_1B_2B_3B_4B_5 + B_1\bar{B}_2\bar{B}_3\bar{B}_4\ (\bar{B}_5 + \bar{B}_6) \tag{2.32}$$

$$\Delta_7 = B_1\bar{B}_2\bar{B}_3\ (B_5B_6 + B_4) \tag{2.33}$$

$$\Delta_8 = B_1\bar{B}_2B_3\ (\bar{B}_4 + \bar{B}_5\bar{B}_6) \tag{2.34}$$

$$\Delta_9 = B_1\left[\bar{B}_2B_3B_4\ (B_5 + B_6) + B_2\bar{B}_3\bar{B}_4\bar{B}_5\right] \tag{2.35}$$

$$\Delta_{10} = B_1B_2\bar{B}_3\left[\bar{B}_4B_5 + B_4\ (\bar{B}_5 + \bar{B}_6)\right] \tag{2.36}$$

$$\Delta_{11} = B_1B_2\ (\bar{B}_3B_4B_5B_6 + B_3\bar{B}_4) \tag{2.37}$$

$$\Delta_{12} = B_1B_2B_3B_4 \tag{2.38}$$

A converter using the mechanization of these e- and Δ - equations has been built[5] with relays and diodes, and it uses 15 multicontact relays and 36 diodes. Because of the relays, its maximum conversion rate is 25 conversions/sec. Using only diodes, the same converter would require about 250 diodes, plus 7 triodes for inverting the input; again not counting extra devices for clamping, amplification, etc.

Method 2, then, although still requiring a large number of components. is simpler than Method 1 and would operate as rapidly as desired.

5. Ibid.

040 018

Method 3

The third and final method for mechanizing the binary-decimal converter is suggested by the second special criterion mentioned at the beginning of the discussion of Method 2; that is, that the necessary conversion rate of this particular converter is quite slow, being only 25 or more conversions/sec. This slow conversion rate suggests that the binary input might actually be counted by some scaling device after it enters the converter.

Using a scale-of-128 binary scaler with 7 scaling stages, each of the 7 binary input bits could alter each of the scaling stages in some prescribed manner from some known reset condition. Then the scaler could be triggered by an external device and set to some known stable condition. The number of trigger pulses required to set the scaler to this known condition would depend upon the manner in which the scaler had been altered from its reset condition; i.e., the number of trigger pulses would be a function of the seven-bit binary input. If the number of trigger pulses were counted, a decimal indication of the input could be obtained.

A block diagram of this idea is shown in Fig. 2.5. The binary scaler, consisting of the 7 coupled binary stages at the top of Fig. 2.5, is reset to 1111111. Then the binary input ($B_1B_2B_3B_4B_5B_6B_7$) is allowed to enter the scaling stages. If a particular B input is 1, it will change its corresponding scaler stage to the zero condition; otherwise the stage will remain set at 1. Thus each of the B's is effectively inverted when it is stored. Inverting all of the bits of a binary number in this manner is the same as creating its one's complement, which is

$$2^n - N - 1 \quad ,$$

where N is the binary number and n is the number of bits in N, or 7 in this case. The number

$$127 - N \quad ,$$

then, is stored in the binary scaler.

When the binary scaler was reset to 1111111, the decade scaler at the bottom of Fig. 2.5 was reset to 199. The decade scaler altogether is a scale-of-200 scaler, so that one pulse from the oscillator will make it read 000.

040 019

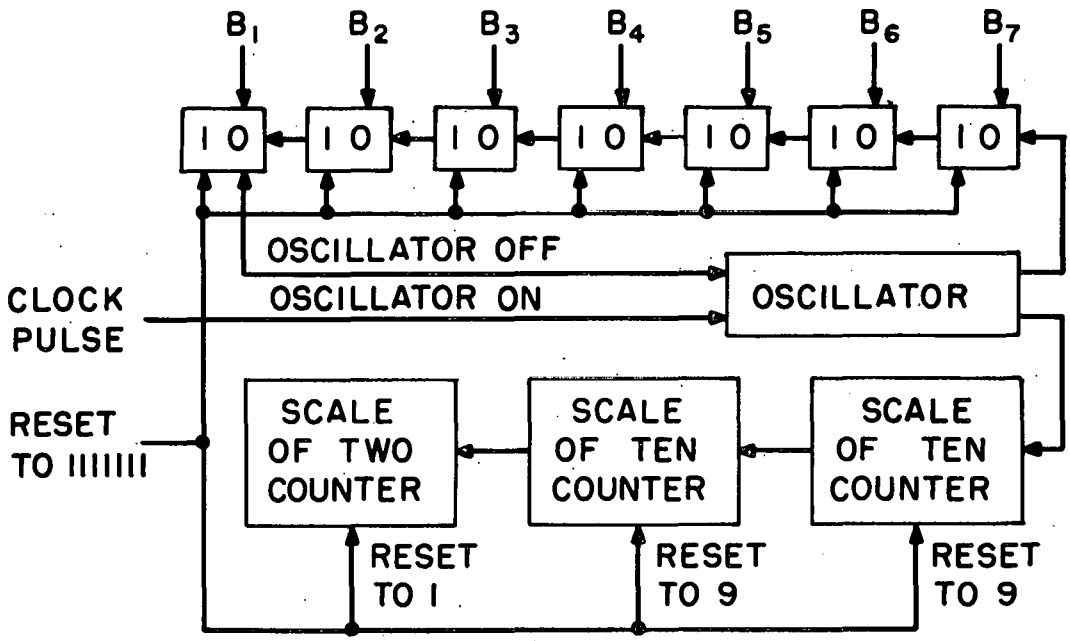


Fig. 2.5 — A Binary-to-Decimal Converter Using Method 3

After the above-mentioned reset and storage operations have taken place, a clock pulse starts the oscillator running. The pulses out of the oscillator enter both scalers. The first pulse from the oscillator increases the number stored in the binary scaler by one, and also changes the decade scaler from a reading of 199 to a reading of 000. Thus, at the instant that the decade scaler reads 000, the number in the binary scaler is:

$$(127 - N) + 1$$

$$= 128 - N \quad .$$

The oscillator continues to trigger both scalers until an output from the last binary stage shuts it off. At this point, the binary scaler is actually storing the number 128, since all of its stages are in the "0" condition. The decade scaler now reads the difference between 128 and (128 - N), or N, and the binary-to-decimal conversion has been completed.

This relatively simple method of conversion seems to represent a considerable saving in components over Methods 1 and 2. For this reason, it was chosen as the best method for the mechanization of the converter.

040 020

Although Method 3 seems to be somewhat unique, it is not presented as an absolute or "best" method of binary-decimal conversion. It is, however, the best method that could be found for this particular application. The next chapter describes in detail the circuitry and operation of a binary-decimal converter using Method 3.

CHAPTER 3. CIRCUITRY AND OPERATION OF THE BINARY-TO-DECIMAL CONVERTER

A basic block diagram of the converter was given in the third part of the preceding chapter, and the theory of operation was discussed. It was found that an oscillator could be made to put out a number of pulses equal to the input binary number plus one, and that these pulses could be counted by a decade scaler which had effectively been reset to minus one. The decade scaler would then give a decimal indication of the input.

A complete operational block diagram is shown in Fig. 3.1. The dotted lines around the individual blocks give reference to the figures that show the individual circuit schematics. In the scalers, where individual stages are identical, the circuit of only one stage will be shown. A complete schematic appears at the end of this chapter.

Referring to Fig. 3.1, the binary scaler consisting of seven binary blocks may be seen at the top of the figure, and the decade scaler at the bottom. Assuming that the binary scaler has been reset to 1111111 and the decade scaler to 199, the sequence of operation goes as follows:

1. Binary number $B_1B_2B_3B_4B_5B_6B_7$ enters scaler stages. If any $B_i = 1$, then $i^{\underline{th}}$ stage is flipped to "0." "And Drive" is not on, so no interstage coupling is possible at this time.
2. With inverted binary bits now stored, clock pulse enters "Timing One-Shot."
3. Timing One-Shot immediately pulses "MV Switch," turning on "MV" (free-running multivibrator). And Drive is energized simultaneously, and interstage coupling in binary scaler is now possible.
4. MV free-runs and sends pulses to inputs of both scalers.
5. MV continues to run until binary scaler reaches 128, or 0000000, and gives an output pulse, causing MV Switch to turn MV off. "Decade Scaler" now reads decimal equivalent of input.

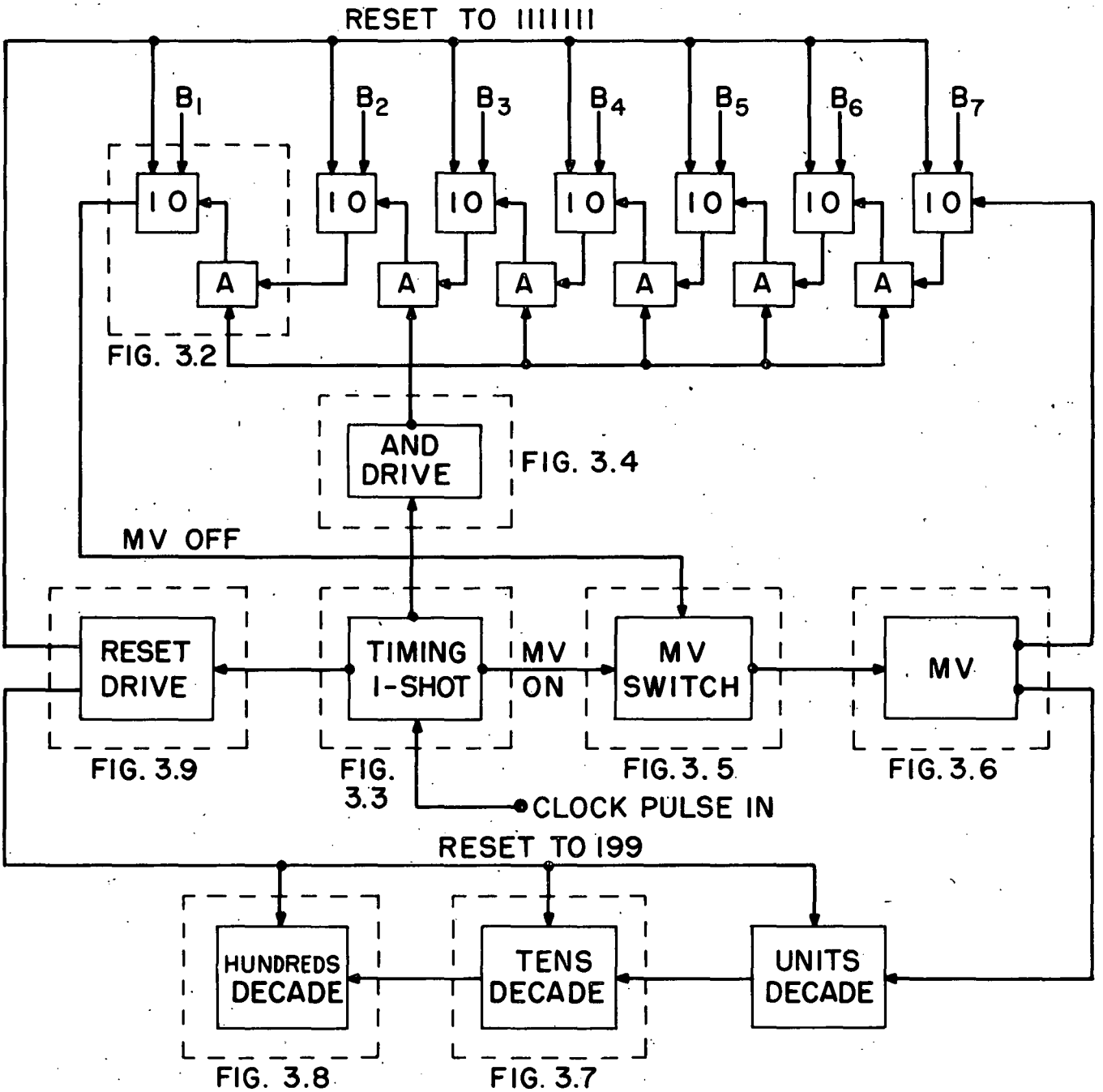


Fig. 3.1 — Operational Block Diagram of the Binary-Decimal Converter

040 023

6. Converter remains inoperative, with decade scaler reading correct output, until Timing One-Shot relaxes and returns to stable state.
7. When Timing One-Shot relaxes, "Reset Drive" is operated and scalers are reset. And Drive is simultaneously de-energized, cutting off interstage coupling.
8. Next binary number enters scaling stages, and cycle starts again.

Since the converter output is presented on the neon lights of the decade scaler, it is essential to have the scaler reading the correct output most of the time, in order that the observer may see the correct lights lit brightly. The decade scaler is either reset to 199, or it is being pulsed by the multivibrator, or it is reading the correct output. From this consideration, two things are essential for proper operation:

1. Reset must occur at the last possible moment before the binary number enters the converter.
2. The free-running multivibrator must run at a high enough frequency so that up to 128 pulses can be put out in a small fraction of the time between binary inputs.

If these two conditions are met, then the proper lights on the decade scaler will appear to burn brightly and continuously. They are easily met in this particular application, since the binary input frequency can be made to be as low as 25 numbers/sec. The multivibrator runs at 27.8 kc, and will put out 128 pulses in about 4.6 milliseconds. Thus, if the conversion rate is 25 conversions/sec, or one conversion every 40 milliseconds, multivibrator time is negligibly short. The time that the converter spends in the reset condition is determined somewhat by the shape of the binary input pulses, but it too is negligibly short. Due to the persistence of the eye, the proper neon lights appear to glow continuously.

The circuitry of each block in Fig. 3.1 is generally quite simple and straightforward. The binary scaler stages will be described first, and then the Timing One-Shot, the And Drive, the MV Switch, the MV, the Decade Scaler and the Reset Drive will be described in order.

040 024

Binary Scaler Stage with Coupling

The Binary Stage is shown in Fig. 3.2. It is almost exactly like a single Berkeley scaler stage,[6] and is a typical example of the standard flip-flop circuit, or Eccles-Jordan circuit, with some modifications. Each of the seven stages is the same, except that the first stage does not have the IN48 diode or the input from And Drive.

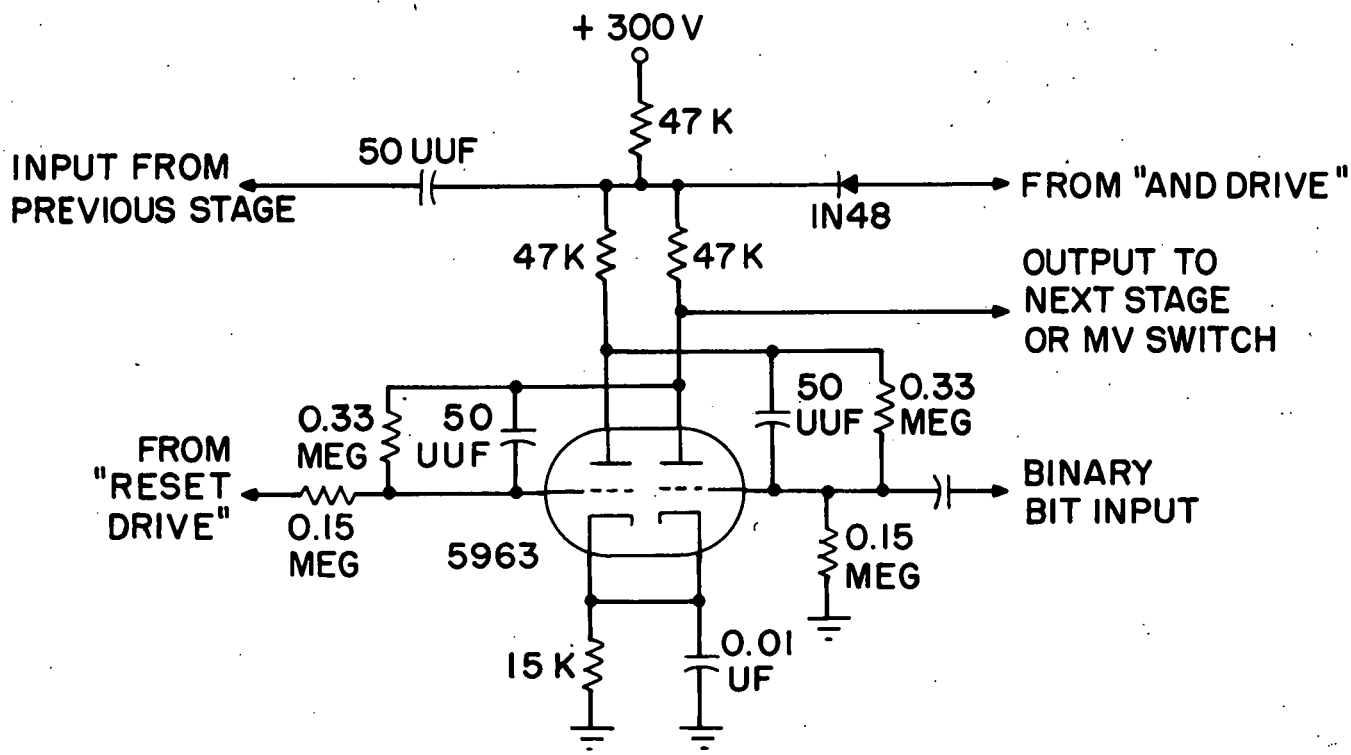


Fig. 3.2 — Binary Scaler Stage (Except First Stage)

When a positive pulse arrives from Reset Drive, the left grid is driven positive and the left side conducts. A positive "Binary Bit" pulse on the right grid would cause the right side to conduct and the left side to cut off, or if there were no Binary Bit input, the left side would continue to conduct.

Both positive and negative pulses are coupled in from the previous stage; however, the circuit can be triggered only with negative pulses. The negative pulses will only trigger the

6. Berkeley Scientific Division of Beckman Instruments, Inc., "Instruction Manual - Berkeley Decimal Counting Units - Models 700A and 705A."

040 025

circuit if the And Drive input is less than 180 volts. If the And Drive input is more than 180 volts, the input pulse will see a lower impedance through the diode, and will be too small to trigger the stage.

The stage uses a 5963 twin-triode, which is specially designed for this type of on-off operation. It has a resolution time of about 10 microseconds, so that input frequencies up to about 100 kc may be used.

Matching the input binary lines to the 7 binary stages is important for proper operation of the converter. In the circuit used here, almost complete isolation of the scaler stage from any input source impedance is necessary. If some loading is anticipated at the binary bit inputs, component changes may be necessary in the binary stages in order to obtain stable operation and proper interstage triggering.

The Timing One-Shot

The Timing One-Shot, shown in Fig. 3.3, is a typical one-shot, or unistable multivibrator. The left side, with its grid tied to B+, is normally conducting. When a negative clock pulse enters, the left side cuts off and the right side conducts for a length of time determined primarily by the 1-meg resistor, the 0.03-mfd capacitor, and the setting of the 5-meg pot. The one-shot then returns to its stable condition.

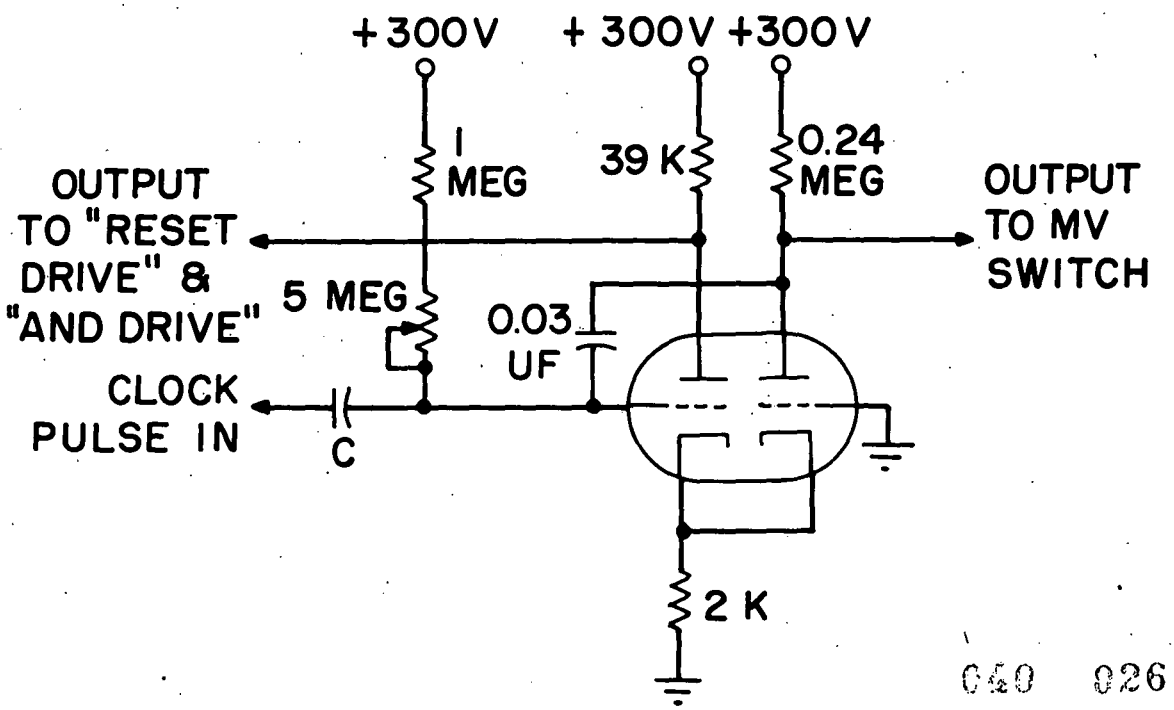


Fig. 3.3 — The Timing One-Shot

A clock pulse in causes the left plate to rise from 30 volts to 300 volts, and this positive-going voltage is coupled out to the And Drive circuit. At the same time, a negative pulse is coupled from the right plate to turn on the MV Switch. Later, when the one-shot returns to its stable state, the negative-going left plate turns off the And Drive, and simultaneously sends a negative pulse out to the Reset Drive. It is obvious that the clock pulse must be a negative pulse which arrives just <u>after</u> the binary inputs (see description of Fig. 3.1).

The length of time between the clock pulse and the negative pulse out to Reset Drive is determined by the setting of the 5-meg pot. The 5-meg pot, then, may be used to adjust the timing of reset, so that it occurs just before the binary input pulses arrive at the scaler stages. The duration between clock pulse and reset is variable from 8 ms to 53 ms. The 1-meg resistor is added in series with the 5-meg pot to insure a minimum duration, long enough for the MV to put out 128 pulses. The input capacitor, C, has a value which depends on the size and shape of the clock pulse.

A typical waveform appearing at the left plate is shown below.

<u>And Drive</u>

The And Drive circuit is simply a triode amplifier, inverter and impedance matcher which is connected to the six interstage binary coupling lines through six diodes. With no signal in on the grid, the triode is held cut off by the cathode biasing arrangement shown in Fig. 3.4.

After the storage operation has taken place in the binary stages, the positive-going plate of the Timing One-Shot is coupled to the grid of And Drive through the 0.05-mfd capacitor. The plate of And Drive then goes negative, and binary interstage coupling is then possible. The long-time constant in the And Drive grid circuit causes the plate to remain low until a negative voltage is coupled in from the Timing One-Shot, cutting off the And Drive tube and thus again making binary interstage coupling impossible.

040 027

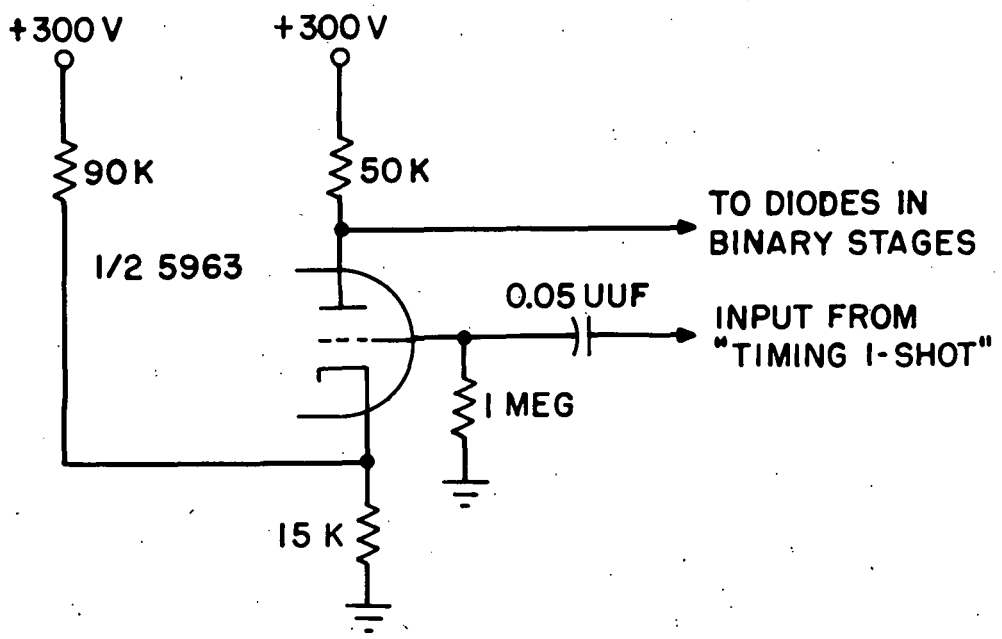


Fig. 3.4 — And Drive Circuit

The And Drive 50-k plate resistance is low enough so that a low shunting impedance is seen by the binary outputs whenever the And Drive tube is not conducting.

MV Switch

The MV Switch is a single flip-flop stage which acts as an on-off switch for the MV (Fig. 3.5). When a negative pulse is coupled in from the Timing One-Shot the right side cuts off, the right plate goes positive, and the MV is turned on. When a negative pulse is coupled in from the last Binary Stage, the circuit is flipped back and the MV is turned off. The manner in which the MV is switched on and off is described in the next section.

The MV

The MV stage is a standard plate-coupled free-running multivibrator circuit, designed to run at about 27.8 kc (Fig. 3.6). The circuit is symmetric and free-running only if the right side of the MV Switch circuit is cut off. If the right side of the MV Switch is conducting, then the left side of the MV is driven into cutoff by virtue of the direct coupling between the stages. Fast and positive switching of the MV is accomplished with this arrangement.

040 028

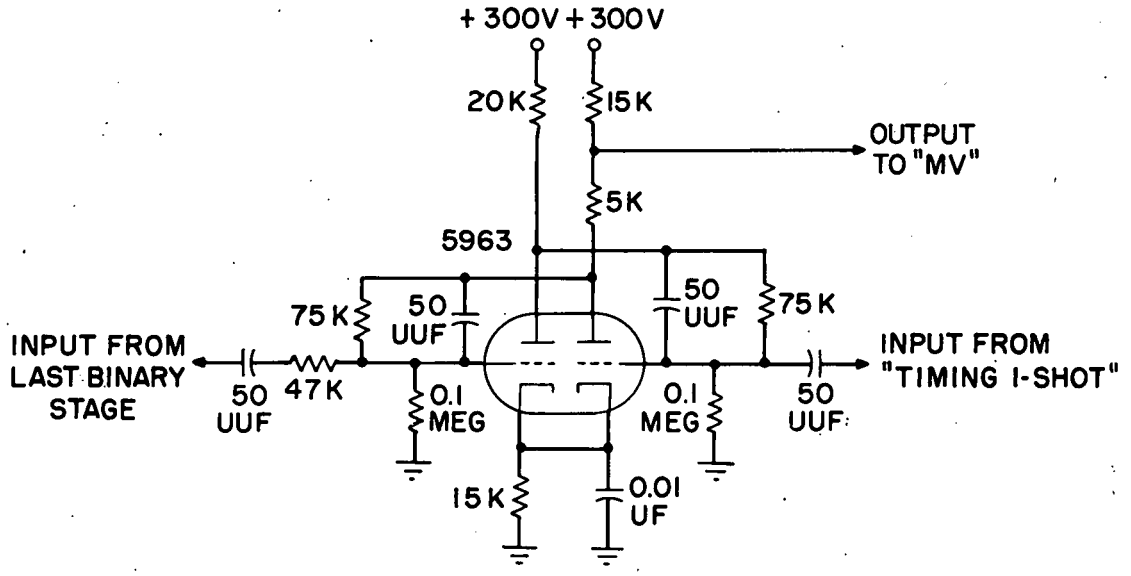


Fig. 3.5 — MV Switch Circuit

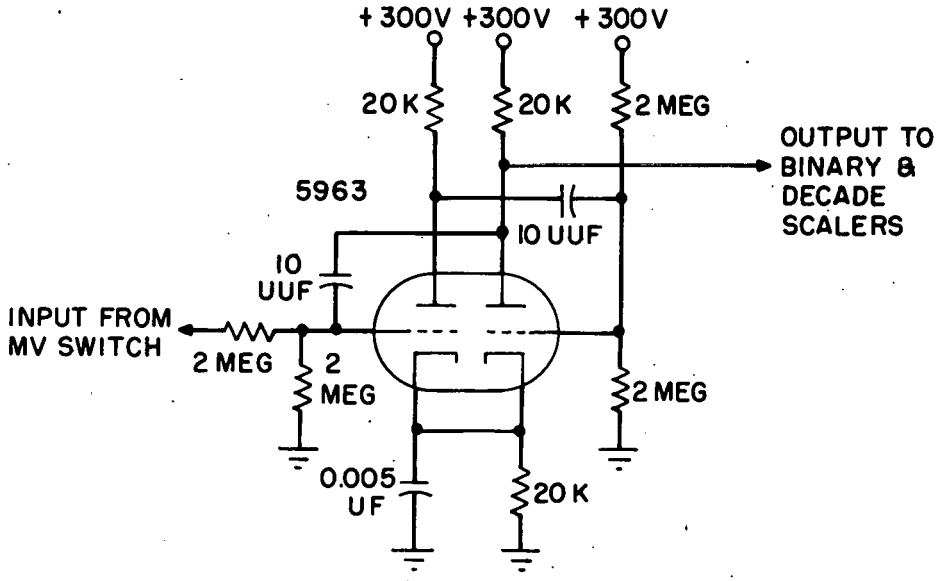


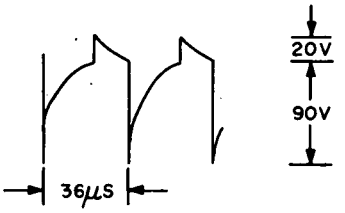
Fig. 3.6 — MV Circuit

One point is important in the operation of the MV: when it is switched on, the left side conducts first, and so a positive pulse is first coupled out to the scalers, and then a negative pulse, a positive pulse, a negative pulse, and so on. Since only negative pulses will trigger the scalers, a delay of

$$\frac{1}{2}\left(\frac{1}{f}\right) = \frac{1}{2}\left(\frac{1}{27.8 \times 10^3}\right) = 18.0\ \mu\text{sec}$$

040 029

is created between the time that the MV Switch operates and the time that the first trigger pulse goes to the scalers. This delay allows the And Drive circuit time to completely turn on interstage coupling in the binary scaler. The free-running MV waveform is shown below.



M.V.
(EITHER PLATE)

The Decade Scaler

The units and tens decades are alike, and Fig. 3.7 shows the circuit of either decade. The hundreds decade is shown in Fig. 3.8. When the three decades are cascaded, a scale-of-200 decade scaler is the result. Since the maximum number of input pulses to the scaler is 128, the scale of 200 is adequate.

The units and tens decades are almost exact copies of the Berkeley decade unit,[7] except for the reset connections, the capacitors in the permutating networks, and other minor changes. The Berkeley unit is designed for mechanical reset, while electronic reset was necessary here. Also, the Berkeley units are reset to zero, while the units and tens decades here are each reset to nine. These decades are standard split-feedback scale-of-ten circuits, with capacitive coupling and 4221 code permutating networks.[8] The 0.25-mfd capacitors in Figs. 3.7 and 3.8 hold the neon lights off during reset, so that they will not flicker while the converter is running.

The Hundreds circuit, shown in Fig. 3.8, is a single binary stage. It is exactly like binary stages 2, 3 or 4 in either the units or the tens circuit. The Ne-51 neon lamp will light when the stage remains in the "one" state, with the right side conducting. This stage is reset to one.

7. Ibid.
8. For description of operation, see Ibid.

040 030

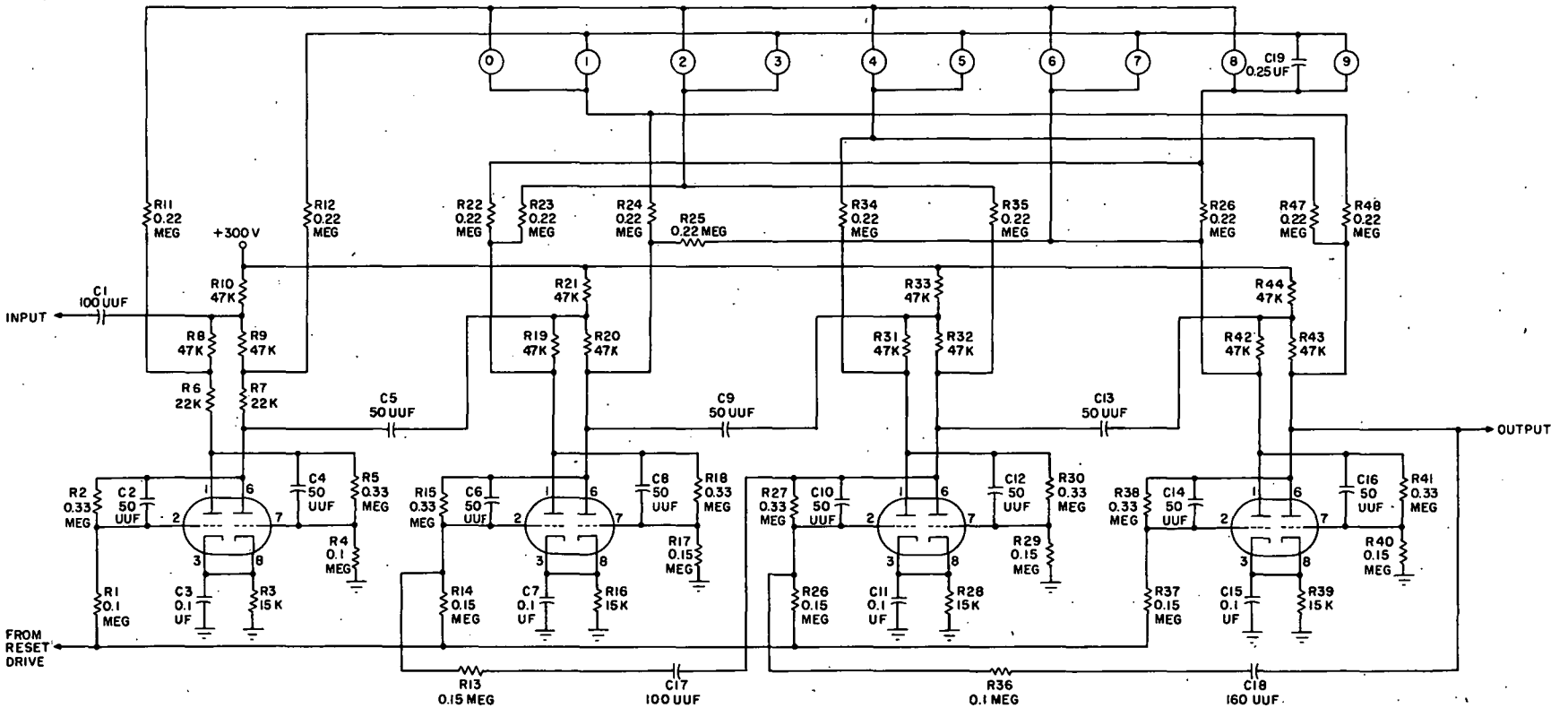


Fig. 3.7 — Units and Tens Decades

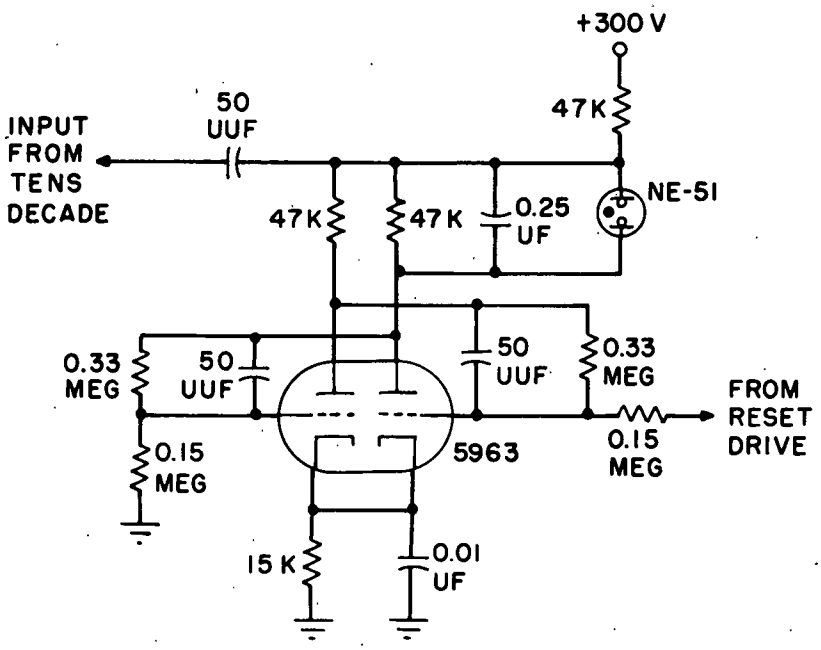


Fig. 3.8 — Hundreds Decade

Reset Drive

The Reset Drive circuit, which sends a positive pulse to the proper grids of all scaler stages, is shown in Fig. 3.9. Due to the short time constant in the input grid circuit, the pulse from the left plate of the Timing One-Shot appears at the input grid in differentiated form, which is a positive pulse followed by a negative pulse. Since the input triode is normally conducting, only the negative input pulse has an appreciable effect on the plate voltage.

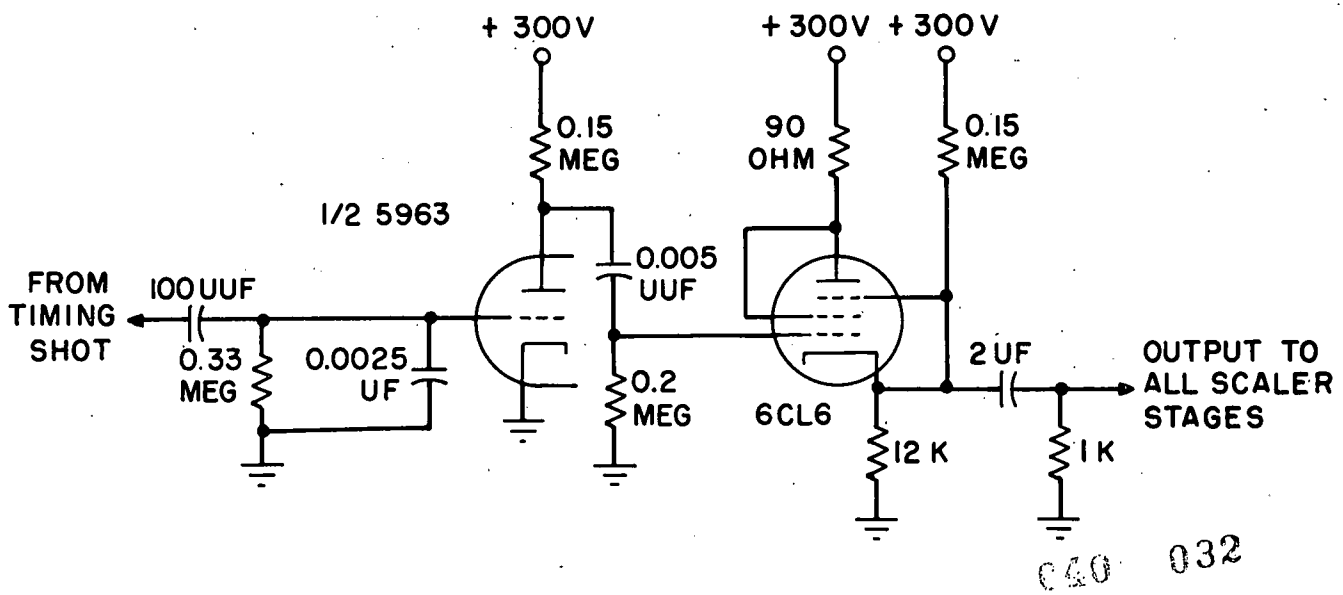
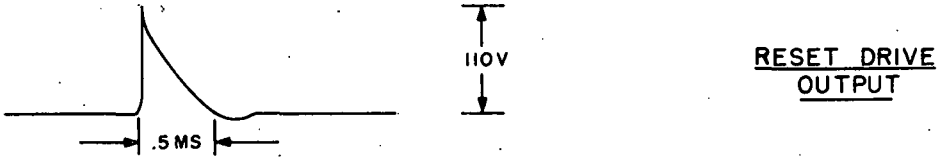


Fig. 3.9 — Reset Drive Circuit

The negative input pulse causes the triode plate to rise momentarily, and this positive pulse is coupled to the control grid of the pentode, which is held close to cutoff with cathode biasing. The pentode acts as a cathode follower and sends a 110-volt positive pulse out through the 2-mfd capacitor to all scaler stages. This pulse is shown below.

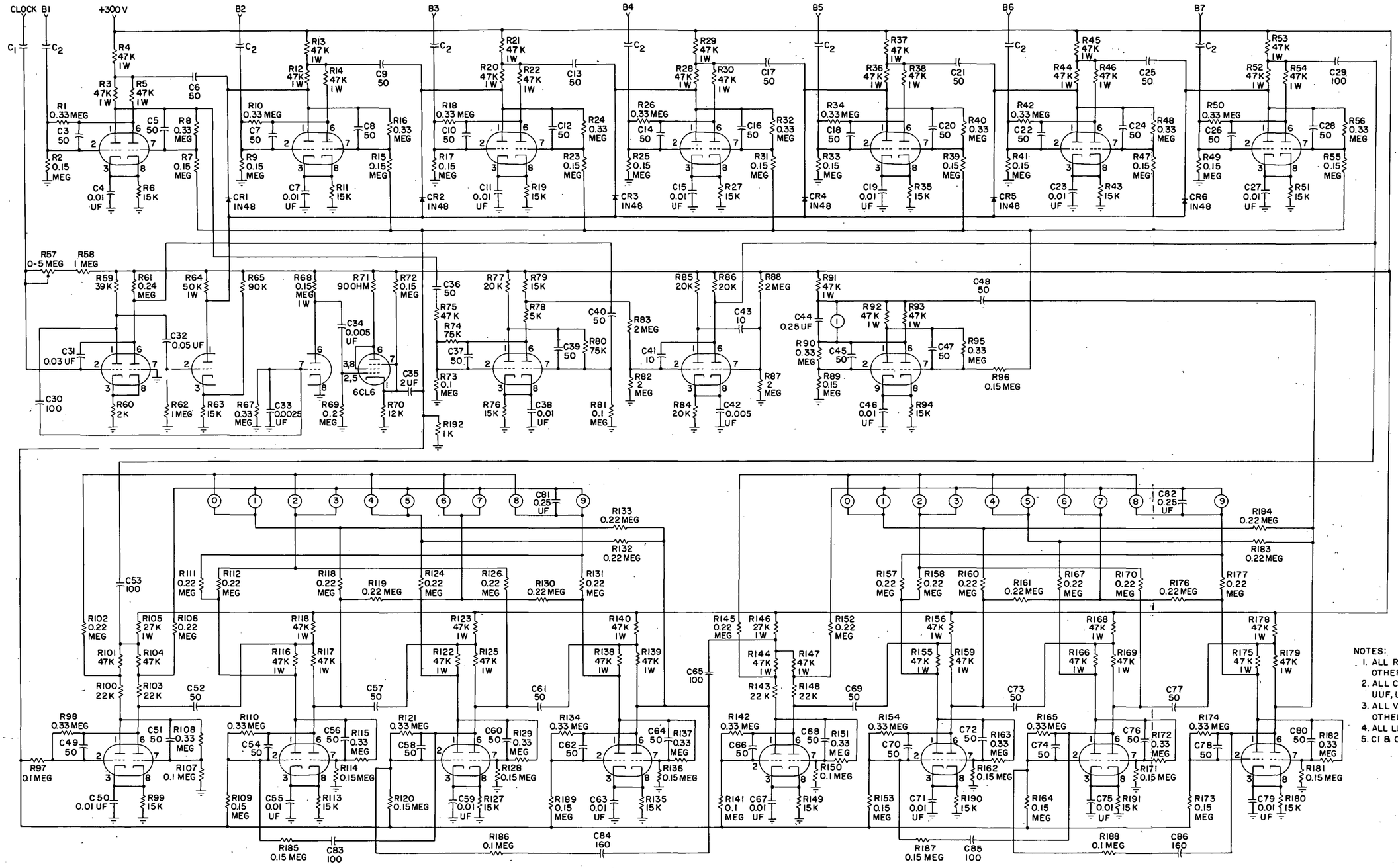


This completes the discussion of the individual circuits in the converter. A complete schematic is given in Fig. 3.10. Power requirements for the converter are as follows:

Plate Supply: 300 v dc at 80 ma unregulated

Filaments: 6.3 v ac at 6.65 a.

The binary input pulses (positive pulse = 1, no pulse = 0) and the input clock pulse (negative pulse) are matched to the converter by varying the input coupling capacitors. In the case of the binary stages, it is essential that the inputs do not present too low an impedance as seen by the input grids; otherwise the binary scaler will not count properly. The clock pulse must arrive <u>just after</u> the binary input pulses.



NOTES:
1. ALL RESISTORS ARE 1/2 WATT, ±5%, UNLESS OTHERWISE NOTED.
2. ALL CAPACITORS ARE 400 W.V., ±10%, IN UUF, UNLESS OTHERWISE NOTED.
3. ALL VACUUM TUBES ARE 5963'S UNLESS OTHERWISE NOTED.
4. ALL LIGHTS ARE NE-51.
5. C1 & C2 VARY WITH INPUT PULSE SIZES.

040 - 034

FIG. 3.10 -- SCHEMATIC, BINARY-TO-DECIMAL CONVERTER

CHAPTER 4. CONCLUSION

There are several different basic designs for a binary-to-decimal converter. The design used here seems to have the advantage of being a very simple design, but it cannot be proven to be the simplest possible.

The maximum conversion rate of this converter is limited because the converter actually "counts" the binary number. Other designs do not have this limitation.

The converter described here meets the requirements of the Sandia Corporation data handling system for which it was designed, and it should be useful in other data handling and computing systems as an indicating device.

040-035

APPENDIX A

THE GRAY CODE AND GRAY-TO-BINARY CONVERSION

The data-handling system that the binary-to-decimal converter has been designed for actually employs the inverted binary, or Gray code, and a Gray-to-decimal converter is what is actually required. The seven-bit Gray code for decimal numbers 1 through 127 is shown in Table A.1.

The Gray code is a nonweighted code. Inspection of this code indicates that Gray-to-decimal conversion would be more difficult than binary-to-decimal conversion. The Boolean equations for Gray-decimal are as complex or more complex than the binary-decimal equations. Furthermore, since the Gray code is nonweighted, no Gray-decimal converter employing Method 3 is possible.

Gray-to-binary conversion, however, is quite simple. Letting $G_1G_2G_3$ --- G_7 be the Gray-coded number and $B_1B_2B_3$ --- B_7 be the binary number, where B_1 and G_1 are the most significant bits, the Boolean relationships are

$$B_1 = G_1$$

and

$$B_n = B_{n-1}\,\overline{G_n} + \overline{B_{n-1}}\,G_n \quad \Big| \quad n = 2, 3, 4, 5, 6, 7$$

These equations may be easily verified by inspecting Table A.1. They are comparatively easy to mechanize. A block diagram equivalent of the above equations is shown in Fig. A.1. In order to minimize the number of inverters (triodes), the B_n equations are manipulated as follows:

Table A.1 — THE SEVEN-BIT GRAY CODE

Decimal	Gray (G 1 2 3 4 5 6 7)	Decimal
0	0 0 0 0 0 0 0	0
1	0 0 0 0 0 0 1	1
2	0 0 0 0 0 1 1	2
3	0 0 0 0 0 1 0	3
4	0 0 0 0 1 1 0	4
5	0 0 0 0 1 1 1	5
6	0 0 0 0 1 0 1	6
7	0 0 0 0 1 0 0	7
8	0 0 0 1 1 0 0	8
9	0 0 0 1 1 0 1	9
10	0 0 0 1 1 1 1	10
11	0 0 0 1 1 1 0	11
12	0 0 0 1 0 1 0	12
13	0 0 0 1 0 1 1	13
14	0 0 0 1 0 0 1	14
15	0 0 0 1 0 0 0	15
16	0 0 1 1 0 0 0	16
17	0 0 1 1 0 0 1	17
18	0 0 1 1 0 1 1	18
19	0 0 1 1 0 1 0	19
20	0 0 1 1 1 1 0	20
21	0 0 1 1 1 1 1	21
22	0 0 1 1 1 0 1	22
23	0 0 1 1 1 0 0	23
24	0 0 1 0 1 0 0	24
25	0 0 1 0 1 0 1	25
26	0 0 1 0 1 1 1	26
27	0 0 1 0 1 1 0	27
28	0 0 1 0 0 1 0	28
29	0 0 1 0 0 1 1	29
30	0 0 1 0 0 0 1	30
31	0 0 1 0 0 0 0	31
32	0 1 1 0 0 0 0	32
33	0 1 1 0 0 0 1	33
34	0 1 1 0 0 1 1	34
35	0 1 1 0 0 1 0	35
36	0 1 1 0 1 1 0	36
37	0 1 1 0 1 1 1	37
38	0 1 1 0 1 0 1	38
39	0 1 1 0 1 0 0	39
40	0 1 1 1 1 0 0	40
41	0 1 1 1 1 0 1	41
42	0 1 1 1 1 1 1	42
43	0 1 1 1 1 1 0	43
44	0 1 1 1 0 1 0	44
45	0 1 1 1 0 1 1	45
46	0 1 1 1 0 0 1	46
47	0 1 1 1 0 0 0	47
48	0 1 0 1 0 0 0	48
49	0 1 0 1 0 0 1	49
50	0 1 0 1 0 1 1	50
51	0 1 0 1 0 1 0	51
52	0 1 0 1 1 1 0	52
53	0 1 0 1 1 1 1	53
54	0 1 0 1 1 0 1	54
55	0 1 0 1 1 0 0	55
56	0 1 0 0 1 0 0	56
57	0 1 0 0 1 0 1	57
58	0 1 0 0 1 1 1	58
59	0 1 0 0 1 1 0	59
60	0 1 0 0 0 1 0	60
61	0 1 0 0 0 1 1	61
62	0 1 0 0 0 0 1	62
63	0 1 0 0 0 0 0	63
64	1 1 0 0 0 0 0	64
65	1 1 0 0 0 0 1	65
66	1 1 0 0 0 1 1	66
67	1 1 0 0 0 1 0	67
68	1 1 0 0 1 1 0	68
69	1 1 0 0 1 1 1	69
70	1 1 0 0 1 0 1	70
71	1 1 0 0 1 0 0	71
72	1 1 0 1 1 0 0	72
73	1 1 0 1 1 0 1	73
74	1 1 0 1 1 1 1	74
75	1 1 0 1 1 1 0	75
76	1 1 0 1 0 1 0	76
77	1 1 0 1 0 1 1	77
78	1 1 0 1 0 0 1	78
79	1 1 0 1 0 0 0	79
80	1 1 1 1 0 0 0	80
81	1 1 1 1 0 0 1	81
82	1 1 1 1 0 1 1	82
83	1 1 1 1 0 1 0	83
84	1 1 1 1 1 1 0	84
85	1 1 1 1 1 1 1	85
86	1 1 1 1 1 0 1	86
87	1 1 1 1 1 0 0	87
88	1 1 1 0 1 0 0	88
89	1 1 1 0 1 0 1	89
90	1 1 1 0 1 1 1	90
91	1 1 1 0 1 1 0	91
92	1 1 1 0 0 1 0	92
93	1 1 1 0 0 1 1	93
94	1 1 1 0 0 0 1	94
95	1 1 1 0 0 0 0	95
96	1 0 1 0 0 0 0	96
97	1 0 1 0 0 0 1	97
98	1 0 1 0 0 1 1	98
99	1 0 1 0 0 1 0	99
100	1 0 1 0 1 1 0	100
101	1 0 1 0 1 1 1	101
102	1 0 1 0 1 0 1	102
103	1 0 1 0 1 0 0	103
104	1 0 1 1 1 0 0	104
105	1 0 1 1 1 0 1	105
106	1 0 1 1 1 1 1	106
107	1 0 1 1 1 1 0	107
108	1 0 1 1 0 1 0	108
109	1 0 1 1 0 1 1	109
110	1 0 1 1 0 0 1	110
111	1 0 1 1 0 0 0	111
112	1 0 0 1 0 0 0	112
113	1 0 0 1 0 0 1	113
114	1 0 0 1 0 1 1	114
115	1 0 0 1 0 1 0	115
116	1 0 0 1 1 1 0	116
117	1 0 0 1 1 1 1	117
118	1 0 0 1 1 0 1	118
119	1 0 0 1 1 0 0	119
120	1 0 0 0 1 0 0	120
121	1 0 0 0 1 0 1	121
122	1 0 0 0 1 1 1	122
123	1 0 0 0 1 1 0	123
124	1 0 0 0 0 1 0	124
125	1 0 0 0 0 1 1	125
126	1 0 0 0 0 0 1	126
127	1 0 0 0 0 0 0	127

C49 : 037

$$B_n = B_{n-1}\overline{G_n} + \overline{B}_{n-1}G_n$$

$$= \overline{\overline{B_{n-1}G_n} + \overline{B}_{n-1}\overline{G_n}}$$

$$= \overline{(B_{n-1}G_n)}\,\overline{(\overline{B}_{n-1}\overline{G_n})}$$

$$= \overline{B_{n-1}G_n}\,(B_{n-1} + G_n)$$

The final expression above is mechanized in Fig. A.1 below.

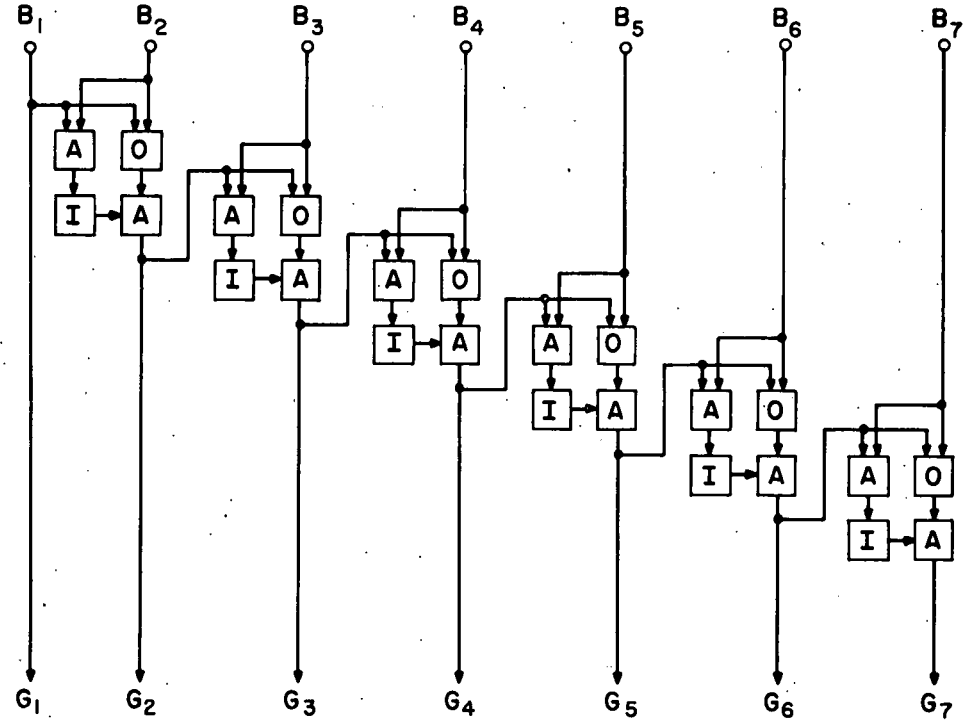


Fig. A.1 — Gray-to-Binary Converter

BIBLIOGRAPHY

1. Richards, R. K., Arithmetic Operations in Digital Computers, D. Van Nostrand Co., Inc., 1955.

2. Buchholz, Werner, "The System Design of the IBM 701 Computer," Proceedings of the IRE, October 1953, Vol. 41, No. 10, p. 1262.

3. Huskey, H. D., Thorensen, R., and others, "The SWAC-Design Features and Operating Experience," Proceedings of the IRE, October 1953, Vol. 41, No. 10, p. 1294.

4. Erickson, R. S., "The Logistics Computer," Proceedings of the IRE, October 1953, Vol. 41, No. 10, p. 1325.

5. Burks, A. W. and Wright, J. B., "Theory of Logical Nets," Proceedings of the IRE, October 1953, Vol. 41, No. 10, p. 1357.

6. White, Garland S., "Coded Decimal Number Systems for Digital Computers," Proceedings of the IRE, October 1953, Vol. 41, No. 10, p. 1450.

7. Robnett, A. V. and Vulgan, E. J., Small-Scale Automatic Field-Data Reduction System, Sandia Corporation TM-236-55-51, October 3, 1955.

8. Jones, R. D., "A High Speed Digital Recording System," Seventh Region IRE Technical Conference, April 29, 1955.

9. Ferner, R. O., Jones, R. D., and Palmer, D. G., High-Speed Digital-Recording System, Sandia Corporation SC-3497(TR).

10. Shaw, R. F., "Payroll Accounting with Elecom 120 Computer," Proceedings of the Western Computer Conference, 1953, published by the IRE, Inc., June 1953.

11. Servomechanisms Laboratory, Massachusetts Institute of Technology, "Project Whirlwind," Summary Report No. 23, Second Quarter, 1950.

12. Jacobs, D. H., "Design Features of the JAINCOMP-C and JAINCOMP-D Electronic Digital Computers," 1954 Convention Record of the IRE, Part 4.

13. Berkeley Division of Beckman Instruments, Inc., "Instruction Manual - Berkeley Decimal Counting Units - Models 700A and 705A."