THE COLLEGE OF AERONAUTICS

DEPARTMENT OF PRODUCTION ENGINEERING

A short note on useful codes for Fluidic Control Circuits

- by -

E.A. Powell

S U M M A R Y

This note gives a survey of codes particularly useful for computational and analogue to digital circuits. Also included are logic circuits capable of converting the code into natural binary form as at some stage in a control circuit this is normally required.

Contents

A short note on useful codes for Fluidic Control Circuits

Natural Binary Code

Weighting Factors

$$\begin{pmatrix} 2^3 \\ 8 \end{pmatrix} \begin{pmatrix} 2^2 \\ 4 \end{pmatrix} \begin{pmatrix} 2^1 \\ 2 \end{pmatrix} \begin{pmatrix} 2^0 \\ 1 \end{pmatrix}$$

2^3 (8)	2^2 (4)	2^1 (2)	2^0 (1)	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

In a control or computational circuit, the intelligence conveyed by a signal may be conveyed by the flow or pressure level of the signal. If the signal is continuously variable (or is allowed to assume many different discrete values) and its magnitude represents a measurement or a computing quantity, circuit tolerances may be critical and a high degree of performance reliability may be difficult to achieve. However if the signal is only required to exist in one of two widely separated states signal accuracy is non-critical.

This is one of the reasons why the natural binary code has become an international standard form for punched tape command signals. Disadvantages that occur using computational logic circuits are (1) error detection is not possible and (2) when converting analogue signals into a digital form (i.e output from a rotary encoder) ambiguities occur. There are several methods of overcoming these limitations; error detection or correction codes for computational circuits and double nozzle systems or reflected codes for converting analogue motion into digital signals.

Typical ambiguous signals shown in figure 1 could be due to slight misalignment of nozzles, variation in pressures or different time delays.

The reader should realise that each column of the natural binary code has a weighting factor from which its decimal value can be calculated.

Example:

Binary number		Decimal value
1101	$= 8 + 4 + 0 + 1 = 13$	

CODES PARTICULARLY USEFUL FOR COMPUTATIONAL CIRCUITS

EXCESS-3 CODE

A	B	C	D	
0	0	1	1	0 (3)
0	1	0	0	1 (4)
0	1	0	1	2 (5)
0	1	1	0	3 (6)
0	1	1	1	4 (7)
1	0	0	0	5 (8)
1	0	0	1	6 (9)
1	0	1	0	7 (10)
1	0	1	1	8 (11)
1	1	0	0	9 (12)

This is a binary decimal form of code which expresses each decade in digits from binary (3) to binary (12) as shown. We observe first that although it is a nonweighted code, the 'excess-3' code is self-complementing and is thus potentially useful in subtraction. For example, the complement of 0 is 9 i.e. 0011 becomes 1100; the complement of 3 is 6 i.e. 0110 becomes 1001 etc. This is also useful for applications which require a count which increases symmetrically either side of a zero position.

A circuit suitable for converting the excess-3 code to natural binary is shown in figure 2.

PARITY

Augmented Code

2^3	2^2	2^1	2^0	P	
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	1	2
0	0	1	1	0	3
0	1	0	0	1	4
0	1	0	1	0	5
0	1	1	0	0	6
0	0	1	1	1	7
1	0	0	0	1	8
1	0	0	1	0	9

Parity, the simplest method for detecting the presence of a single error, depends on the assumption that the probability of multiple errors is exceedingly small. The basic principle of the parity check is to transmit an extra digit with each code group. This digit is chosen so that the entire group, including the parity check digit, contains either an odd or an even number of either 1's or 0's. There are thus four ways in which the parity check can be applied. The majority of systems use only the even check on the 1's; that is, to each code group a parity is included so that the total transmitted group always contain an even number of 1's. A parity checking circuit is shown in figure 3.

2-OUT-OF-5 CODE

A	B	C	D	E	
1	1	0	0	0	0
0	0	0	1	1	1
0	0	1	0	1	2
0	0	1	1	0	3
0	1	0	0	1	4
0	1	0	1	0	5
0	1	1	0	0	6
1	0	0	0	1	7
1	0	0	1	0	8
1	0	1	0	0	9

The 2-out-of-5 is a parity checking type of code which is able to detect all but one kind of error, this being when the incorrect two outputs are reading 1's. With the exception of '0' the positions A, B, C, D, E are weighted 7 - 4 - 2 - 1-Parity, respectively.

Other error detection codes such as 3-out-of-5 and 2-out-of-7 are all possible and have been extensively used in relay computers.

The logic function derived to design the circuit shown in figure 4 are:-

$$2^3 = A.\bar{B}.E$$

$$2^2 = \bar{A}.B + A.E$$

$$2^1 = \bar{A}.C + A.E$$

$$2^0 = \bar{A}.D.\bar{E} + \bar{C}.\bar{B}.E + A.C$$

An error detection circuit is shown in Figure 5.

HAMMERING'S SINGLE ERROR CORRECTING CODE

POSITION

k	k	m	k	m	m	m	
1	2	3	4	5	6	7	
0	0	0	0	0	0	0	0
1	1	0	1	0	0	1	1
0	1	0	1	0	1	0	2
1	0	0	0	0	1	1	3
1	0	0	1	1	0	0	4
0	1	0	0	1	0	1	5
1	1	0	0	1	1	0	6
0	0	0	1	1	1	1	7
1	1	1	0	0	0	0	8
0	0	1	1	0	0	1	9
1	0	1	1	0	1	0	10
0	1	1	0	0	1	1	11
0	1	1	1	1	0	0	12
1	0	1	0	1	0	1	13
0	0	1	0	1	1	0	14
1	1	1	1	1	1	1	15

Hammering developed a number of codes to detect and correct errors.
His code to detect and correct a single error in a four bit binary output
requires seven input readings. The code is shown above where 'm' denotes
a digit carrying information and 'k' denotes a checking digit.

To detect and correct a single error, even parity checks are made
on positions 4, 5, 6, 7; 2, 3, 6, 7, and 1, 3, 5, 7. A parity failure in
any one or more of the combinations will indicate the position of the error.

Let X, Y and Z indicate the outputs of the three parity checks. A
'0' bit output indicating correct even parity and a '1' bit output a parity
failure.

EXAMPLE

	POSITION						
(1)	(2)	(3)	(4)	(5)	(6)	(7)	
0	0	1	1	0	0	1	Correct output for 9
0	0	1	1	1	0	1	Output containing error
			1	1	0	1	4-5-6-7 parity failure $\therefore$ X = 1
	0	1			0	1	2-3-6-7 parity correct $\therefore$ Y = 0
0		1		1		1	1-3-5-7 parity failure $\therefore$ Z = 1

The output from the parity check is in natural binary form.

	X	Y	Z	POSITION
Parity check outputs	1	0	1	5

An error is shown to be in position 5 and therefore the input to position 5 is complemented. A circuit to perform this operation is shown in figure 6.

The addition of an eighth position to the code enables a single error correcting, double error detecting code to be formed. Hammering's code can also be used for double error detecting and correcting, single error correction and triple error detection, or quadruple error detection.

CODES PARTICULARLY USEFUL FOR ANALOGUE TO DIGITAL CONVERSIONS.

PETHERICK CODE

A	B	C	D	
0	1	0	1	0
0	0	0	1	1
0	0	1	1	2
0	0	1	0	3
0	1	1	0	4
1	1	1	0	5
1	0	1	0	6
1	0	1	1	7
1	0	0	1	8
1	1	0	1	9

This code is particularly useful where analogue to digital conversions and computations must be made. Only one bit changes as the code progresses from one decimal value to the next, thus the possibility of ambiguous signals due to the misalignment of a nozzle is eliminated. The code shown in positions B, C and D for values of '0' to '4' is reflected for values '5' to '9', thus to complement any number the bit in row 'A' only need be altered. This ability is potentially useful when subtraction is required.

The logic functions derived to design the circuit shown in figure 7 are:

$$2^3 = A.\bar{C}$$

$$2^2 = C.(A+B)$$

$$2^1 = \bar{B}.C$$

$$2^0 = A.B + A.C.D + \bar{A}.\bar{B}.\bar{D} + \bar{A}.\bar{B}.\bar{C}$$

GRAY CODE

A	B	C	D	
0	0	0	0	0
0	0	0	1	1
0	0	1	1	2
0	0	1	0	3
0	1	1	0	4
0	1	1	1	5
0	1	0	1	6
0	1	0	0	7
1	1	0	0	8
1	1	0	1	9
1	1	1	1	10
1	1	1	0	11
1	0	1	0	12
1	0	1	1	13
1	0	0	1	14
1	0	0	0	15

This is a code designed to overcome the ambiguity problems encountered when using such devices as rotary encoders. The main advantage of this code is its very simple Gray to natural binary conversion logic but has the disadvantage the code recycles on 15 and not 9 as is required by many systems.

Weighting values of 15, 7, 3 and 1 can be given to position A, B, C and D respectively. A simple rule to determine the equivalent decimal value is to subtract, add, subtract, add each successive '1' bit as it appears from left to right.

EXAMPLE 1

	A	B	C	D	
	(15)	(7)	(3)	(1)	
Decimal 6	0	1	0	1	7 - 1 = 6

EXAMPLE 2

	A	B	C	D	
Decimal 13	1	0	1	1	15 - 3 + 1 = 13

The code can be expressed by the following equations to enable a Gray to natural binary convertor to be designed:

$$2^3 = A$$
$$2^2 = \bar{A}.B + A.\bar{B}$$
$$2^1 = \bar{A}.\bar{B}.C + \bar{A}.B.\bar{C} + A.\bar{B}.\bar{C} + A.B.C.$$
$$2^0 = \bar{A}.\bar{B}.\bar{C}.D + \bar{A}.\bar{B}.C.\bar{D} + \bar{A}.B.C.D + \bar{A}.B.\bar{C}.\bar{D}$$
$$+ A.B.\bar{C}.D + A.B.C.\bar{D} + A.\bar{B}.C.D + A.\bar{B}.\bar{C}.\bar{D}.$$

which simplifies to the following

$$2^3 = A$$
$$2^2 = (\bar{2^3}) B + (2^3) \bar{B}$$
$$2^1 = (\bar{2^2}) C + (2^2) \bar{C}$$
$$2^0 = (\bar{2^1}) D = (2^1) \bar{D}$$

From these equations the circuit shown in figure 8 was designed.

MODIFIED GRAY CODE

A	B	C	D	
0	0	0	0	0
0	0	0	1	1
0	0	1	1	2
0	0	1	0	3
0	1	1	0	4
0	1	1	1	5
0	1	0	1	6
0	1	0	0	7
1	1	0	0	8
1	0	0	0	9

The advantage of the simple decoding circuit has been retained with added bonus of being useful where the 9 to 0 transition is important.

If the code needs to be converted into natural binary form, the following equations can be used to design a suitable circuit (figure 9).

$$2^3 = A$$
$$2^2 = B.\bar{A}$$
$$2^1 = C..(\bar{2^2}) + \bar{C}.(2^2)$$
$$2^0 = D.(\bar{2^1}) + \bar{D}(2^1) + A.\bar{B}.$$

LIBAW AND CRAIG CODE

A	B	C	D	E	
0	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	1	2
0	0	1	1	1	3
0	1	1	1	1	4
1	1	1	1	1	5
1	1	1	1	0	6
1	1	1	0	0	7
1	1	0	0	0	8
1	0	0	0	0	9

Although the Gray type of codes are easier to decode into natural binary form, difficulties can be encountered in the manufacture of the encoder strip or drum. This type of code was one developed to reduce these problems. A coded-pattern-nozzle arrangement (as shown in figure 10) can be manufactured with far greater ease than those required by previously mentioned codes.

The following equations can be used to design the circuit shown in figure 11.

$$2^3 = A.\bar{C}$$
$$2^2 = B.C$$
$$2^1 = \bar{B}.D + C.\bar{E}$$
$$2^0 = \bar{D}.C + \bar{D}.E + A.\bar{B} + \bar{B}.C + A.E$$

U-SCAN

This system consists of a natural binary coded strip and two parallel rows of nozzles positioned one digit apart (figure 12). The row selected for reading is determined by the state of the least significant nozzle. When this nozzle is reading '0' the leading nozzles are read, but when '1' the lagging nozzles are read. The only nozzle read while changing its state is the one detecting the least significant digit, therefore the probability of ambiguous signals during transition periods is removed. A circuit capable of performing the required operations is shown in figure 13.

V-SCAN

The nozzle arrangement for a V-Scan of a natural binary code is shown in figure 14. The distance between each pair of nozzles is equal to the length of the previous digit. The selection of the nozzle to be read is made digit by digit. If the signal from the least significant digit is '0', the lagging nozzle of the next significant digit is chosen. The method of selection for each successive nozzle is identical and the process is repeated until the most significant digit is chosen. Figure 15 shows a 'V' scan to binary readout circuit.

References

1. D. Slepian
'A Class of Binary Signalling Alphabets'
Bell System Technical Journal.
Vol. 35, January 1956, pp. 203-234.

2. R.W. Hammering
'Error Detecting and Error Correcting Codes'.
Bell System Technical Journal, Vol. 29, April 1950, pp. 147-160.

3. S. Ramanathan, I. Aviv and R.E. Bidgood
'The Design of a Pure Shaft Encoder'.
2nd Cranfield Fluidics Conference.
January 1967.

4. S.H. Caldwell
'Switching Circuits and Logical Design'.
Wiley.

5. A.K. Susskind
'Notes on Analog-Digital Conversion Techniques'.
Technology Press and Wiley.

6. E.A. Powell
'An Investigation into the Design of a Fluidic Linear Displacement Transducer'.
College of Aeronautics Thesis October 1967.

CODE

OIIO = 6

OIII = 7

IOX = 8

IOCI = 9

READING

IIII = 15

OOOO = O

AND GATE

3 INPUT

OR / NOR GATE

STANDARD BOOLEAN ALGEBRA SYMBOLS

A = A

$\bar{A}$ = NOT A

$+$ = CR

$\cdot$ = AND
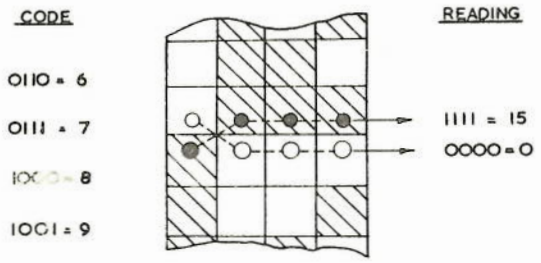
AMBIGUITY IN READING NATURAL BINARY CODE   FIG I

A   B   C   D

2^0

2^1

2^2

2^3
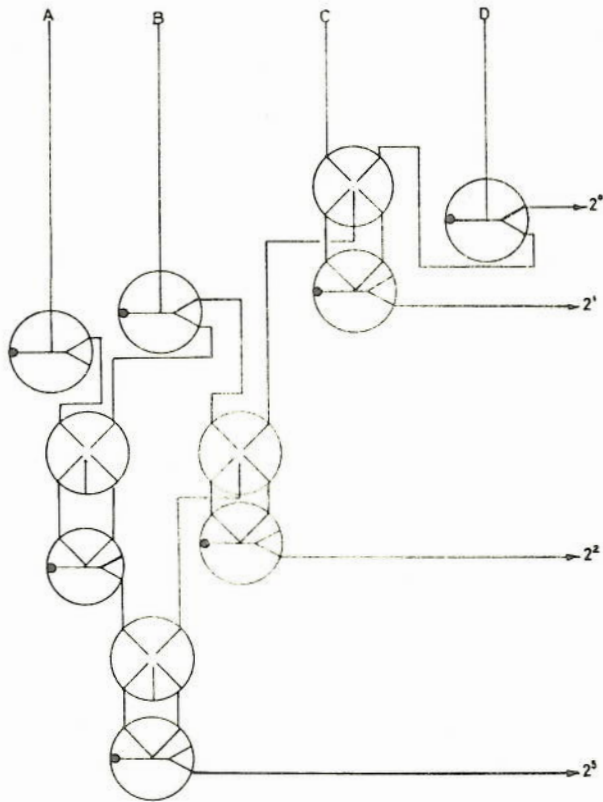
EXCESS 3 TO NATURAL BINARY CONVERTOR   FIG 2

A   B   C   D   P

EVEN NUMBERS

ODD NUMBERS
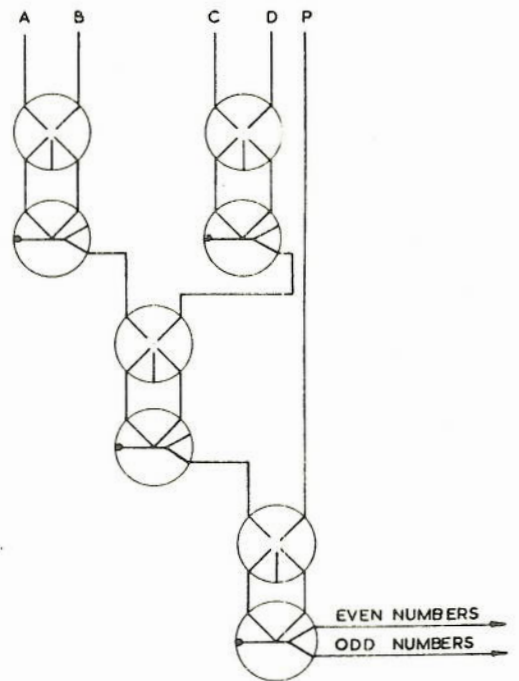
PARITY CHECKING CIRCUIT   FIG 3
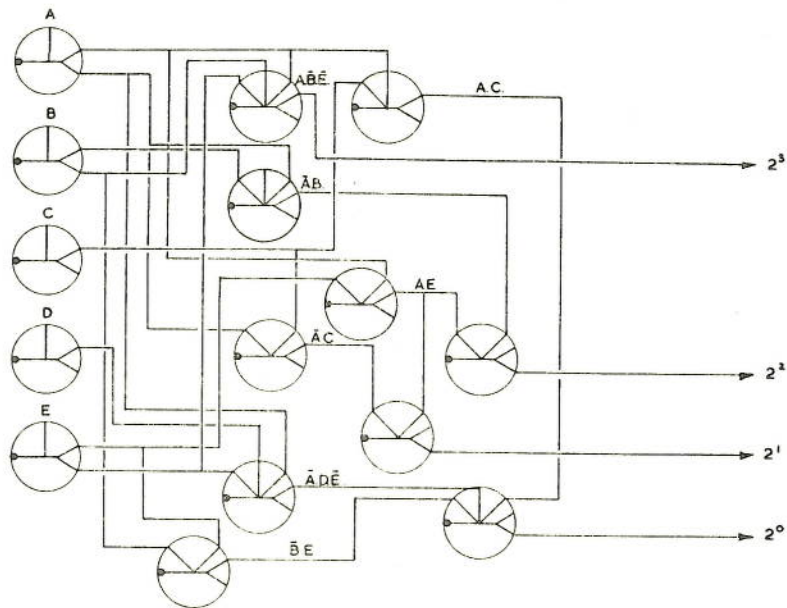
2 OUT OF 5 TO NATURAL BINARY CONVERTOR

FIG 4
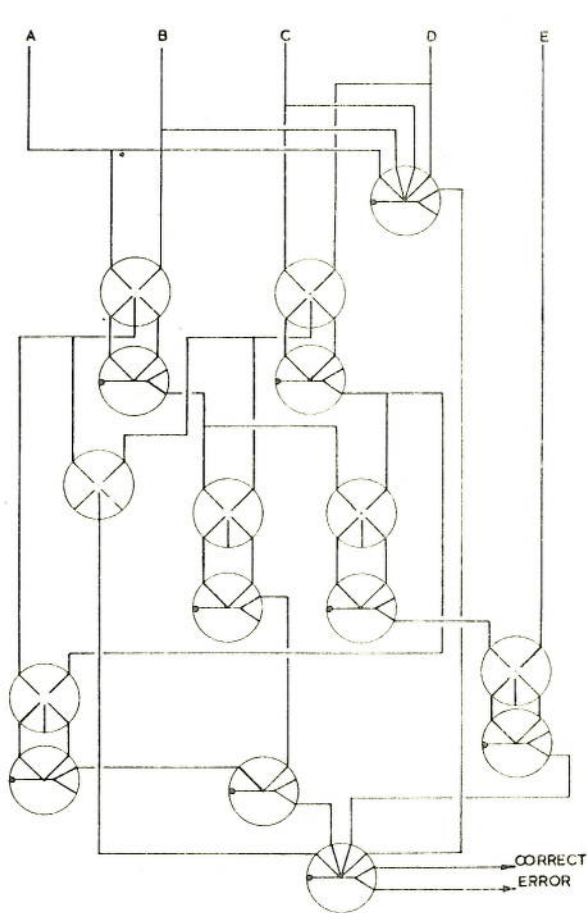
2 OUT OF 5 ERROR DETECTION CIRCUIT   FIG 5
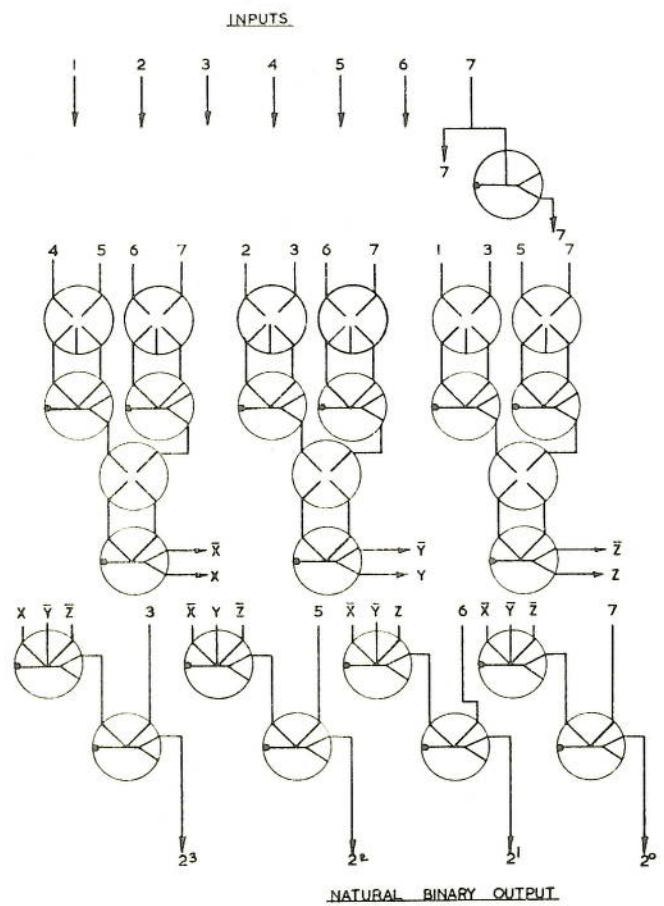
SINGLE ERROR CORRECTION CIRCUIT   FIG 6
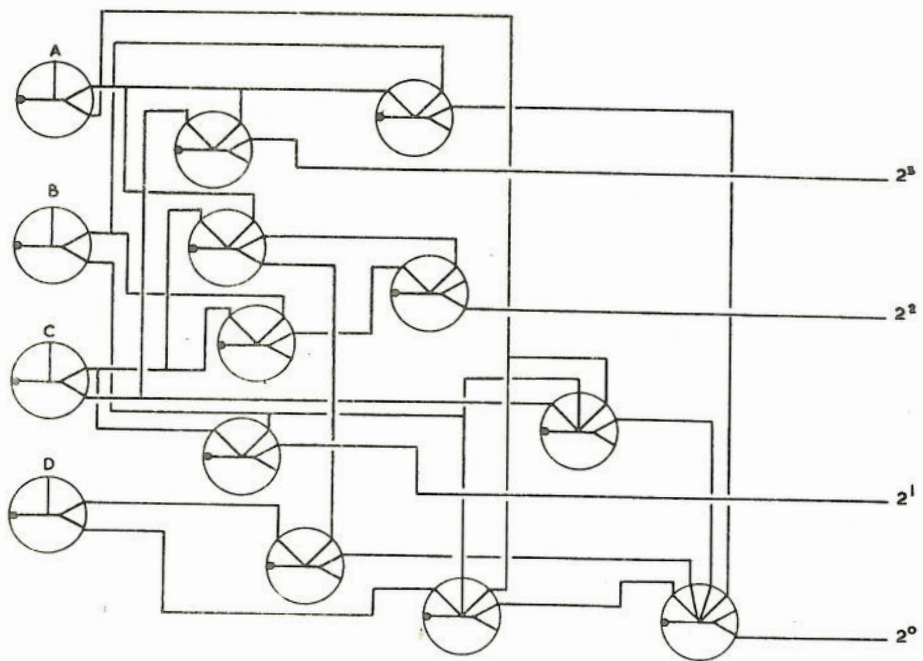
PETHERICK TO BINARY DECODING CIRCUIT   FIG 7
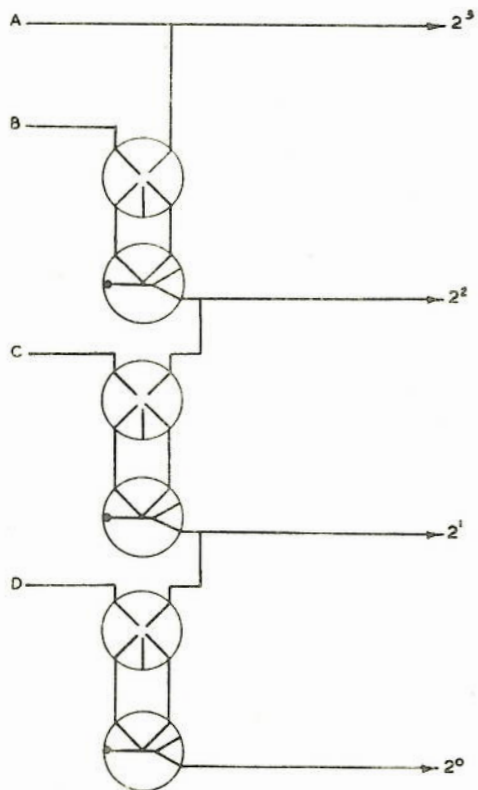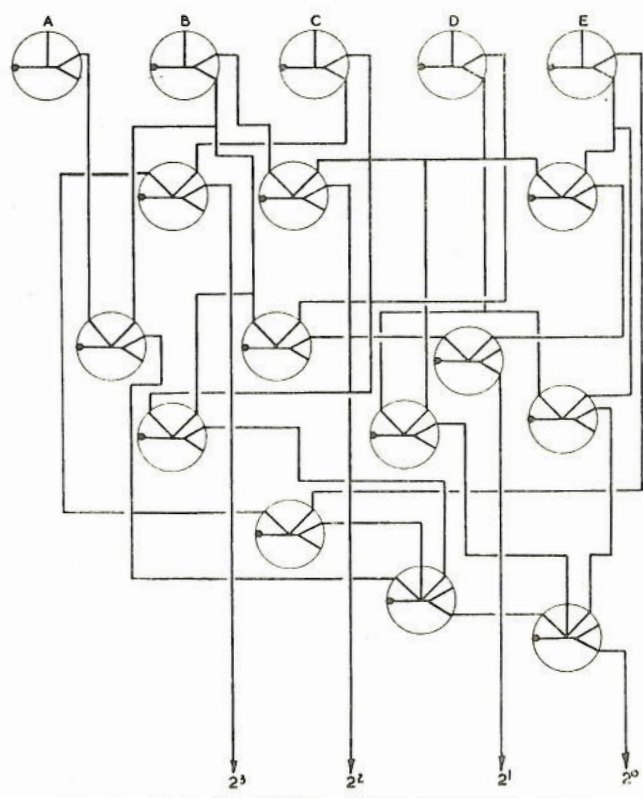

GRAY TO BINARY READOUT CIRCUIT   FIG 8
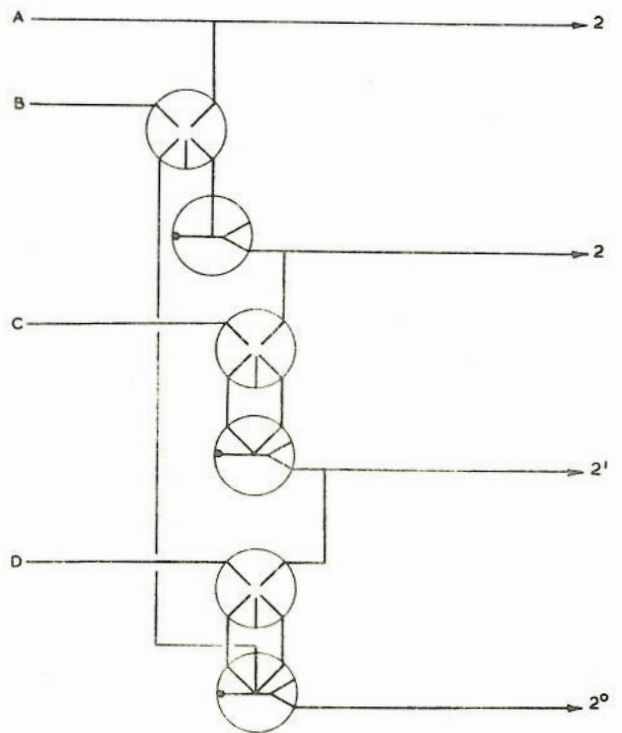
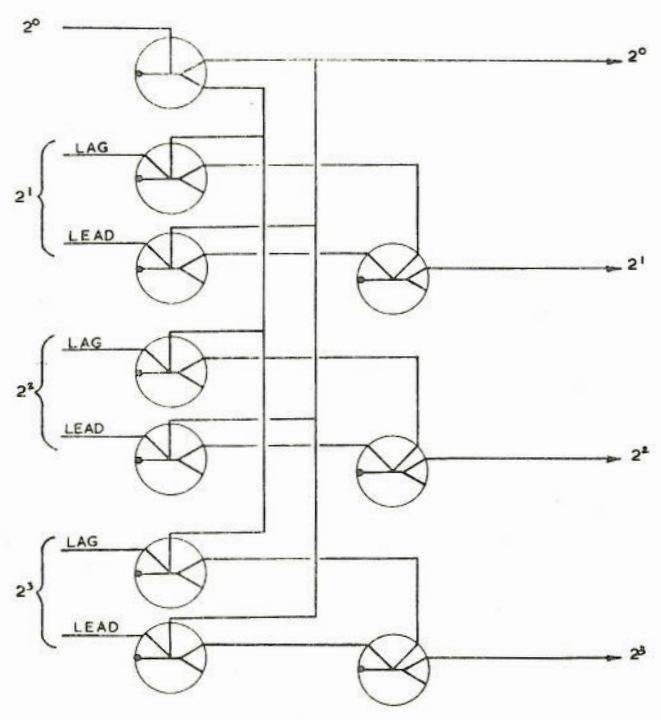
A MODIFIED GRAY TO NATURAL BINARY CONVERTOR   FIG 9
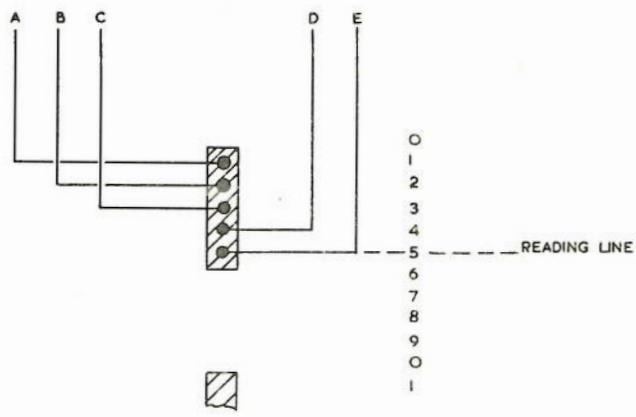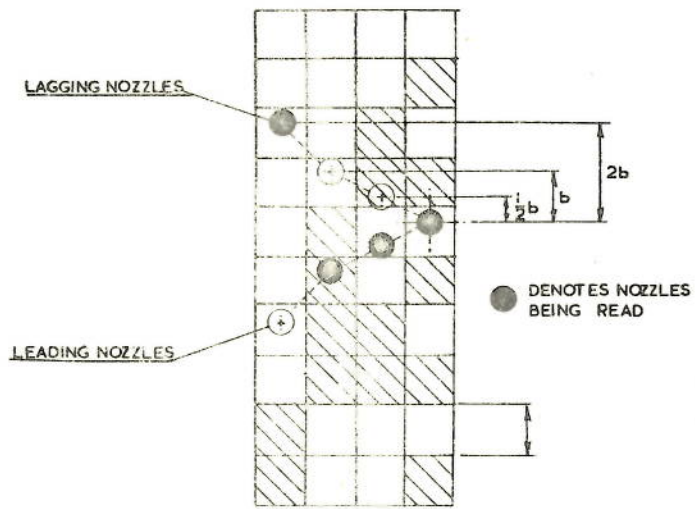
READING TYPE CODER USING LIBAW AND CRAIG CODE. FIG 10
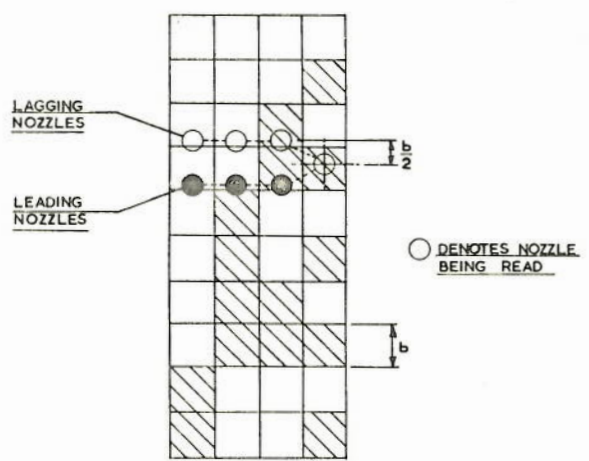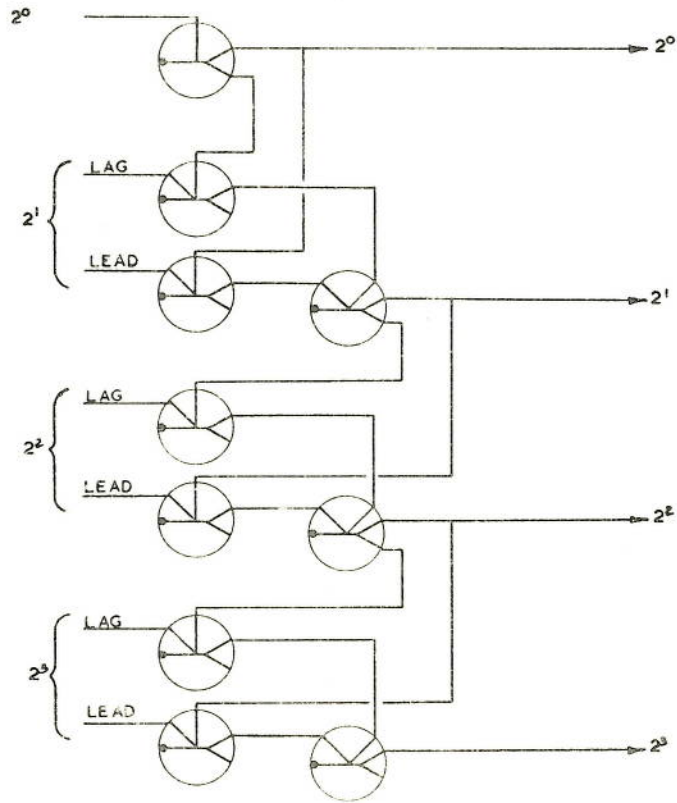

SCHEME OF NOZZLE ARRANGEMENT FOR 'U' SCAN FIG 12

LIBAW AND CRAIG TO NATURAL BINARY CONVERTOR FIG 11

'U' SCAN BINARY READOUT CIRCUIT FIG 13

LAGGING NOZZLES

LEADING NOZZLES

DENOTES NOZZLES BEING READ

$2b$

b

$\frac{1}{2}b$

SCHEME OF NOZZLE ARRANGEMENT FOR V SCAN   FIG 14

2^0

2^0

2^1 { LAG
LEAD }

2^1

2^2 { LAG
LEAD }

2^2

2^3 { LAG
LEAD }

2^3

V SCAN BINARY READOUT CIRCUIT   FIG 15