Digital Display using an Oscilloscope

A design producing a high-density, low-cost, seven-segment c.r.t. display of numbers from 0 to 9

by C. Attenborough

This article describes a way of displaying large amounts of digital data cheaply. The method used is to draw a row of seven-segment patterns on an oscilloscope, and to brighten appropriate segments to display any number on any of the patterns. Fig. 1 shows the numbers 0 to 9 formed by illuminating segments of the pattern (the numbers and letters on the segments will be explained later). This type of display has an almost constant *total* cost, and thus the cost per digit can be low for large amounts of data. This contrasts with gas-discharge display tubes, which have a constant cost *per* digit. If several numbers are to be displayed simultaneously, they can be presented one above the other. The display lends itself well to displaying data stored in serial-in, serial-out shift registers as it is not necessary to have access to all the bits stored in the register, only the end one. This helps to make the display cheaper, because serial-in serial-out registers are cheaper than those with parallel outputs.

Simplified description of the display

The display has eight digits and it will be made clear how different numbers of digits can be accommodated. Fig. 2 is a simplified block diagram of the display. The clock oscillator times the segment definition counter, which feeds analogue and digital circuitry generating X and Y deflection signals to draw one seven-segment pattern on the c.r.o. screen. With the exception of state 6, a different segment of the pattern is being drawn while the counter is in each state. The numbers on the segments in Fig. 1 are the states of the counter during which each segment is being drawn.

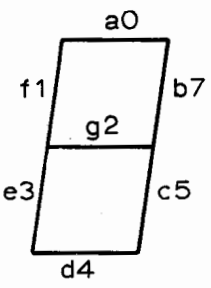
When a pattern has been completed, the segment definition counter goes from state 7 to state 0. This transition does two things; first, it clocks the gross X deflection counter and, secondly, it clocks the shift registers in which the data to be displayed is stored. The gross X deflection counter is connected to a circuit which generates a signal proportional to the state of the counter. This is added to the output of the X deflection circuit. Because the first pattern is drawn while the counter is in state 0 and the second one is drawn while it is in state 1, the second pattern is drawn by the side of the first one. Similarly, subsequent patterns are displaced sideways, until there is a row of them across the screen. There will be as many patterns in the row as the number of states of the gross X deflection counter: when the left-hand pattern is completed, the counter goes from its all-ones state to its 0 state, and the next pattern is drawn in the same place as the first one.

The 1-of-8 decoder, b.c.d. to seven segment decoder, AND gates and OR gate all serve to decide whether or not a bright-up signal is sent to the c.r.o. on which the display is presented. Because the gross X deflection counter and the registers are clocked simultaneously, the information presented to the b.c.d.-to-seven segment decoder changes when the c.r.o. beam moves from one pattern to the next. This ensures that when a new pattern is drawn, the decoder is fed with information corresponding to that digit in the display.

Generating a seven segment pattern

The patterns in Fig. 1 are tilted for better legibility. In designing the deflection circuitry, it is easiest to design for untilted patterns and to add the tilt later. Fig. 3 shows the deflection voltages for drawing one pattern (assuming that positive voltages on the c.r.o's X and Y inputs move the trace to the right and upwards respectively). These voltages are generated at a transistor's collector: it is easier to consider the collector current than the collector voltage. Fig. 4 shows the current waveforms (which are the voltage waveforms inverted) and shows them expressed as a sum of ramps and steps. The ramp and step currents are generated separately and added in the emitter of a common-base stage, the deflection voltage being taken from the stage's collector.

Fig. 5(a) is the X deflection circuit. Transistor Tr_4 is the common-base stage already mentioned, while Tr_3 switches the step current and can be turned off by a logic 'zero' at the end of its base resistor or saturated by a logic '1'. When saturated, its collector current (the step current) is defined by R_1, its collector resistor. Transistors Tr_1 and Tr_2 generate the ramp current and when Tr_1 is saturated, the voltage on C_1 is held at about 3.5V (the saturation voltage of Tr_1 plus the zener voltage). The zener diode ensures that the emitter current of Tr_2 is



a		✓	✓		✓		✓	✓	✓	✓
b	✓	✓	✓	✓			✓	✓	✓	✓
c	✓		✓	✓	✓	✓	✓	✓	✓	✓
d		✓	✓		✓	✓		✓		✓
e		✓				✓		✓		✓
f				✓	✓	✓		✓	✓	✓
g		✓	✓	✓	✓	✓		✓	✓	

Fig. 1. Seven-segment display of numbers from 0 to 9. A tick indicates that a segment is illuminated.

defined, even when Tr_1 is saturated. When Tr_1 is turned off, the voltage across C_1 rises exponentially until the transistor is turned on again. The collector current follows the voltage across C_1, and is added to Tr_3's current in Tr_4.

Fig. 5(b) is the Y deflection circuit, which works in the same way as the X deflection circuit, but has two step current switch transistors as the pattern has three vertical levels and only two horizontal levels. Resistor R_2 is adjusted so that the change in collector current of Tr_2, during a ramp, is equal to the step current. This ensures that the smooth waveforms of Fig. 4 are produced, making the pattern closed without any overlapping of the vertical strokes by the horizontal strokes. Resistor R_4 performs a similar function for the Y deflection circuit.

For ramps and steps to be made at the right times, transistors 1, 3, 5, 7 and 8 must be turned on during certain states of the segment definition counter and off during others. The logic circuitry in Fig. 6 switches the transistors in the correct sequences. Remember the selected output of the 1-of-8 decoder is at logic '0'. The decoder used is actually a 1-of-10 device, two outputs being left unused.

Generating a row of patterns

The row is formed by clocking the gross X deflection counter at the end of each pattern, generating a signal proportional to the state of the counter, and addding it to the X deflection signal. It is convenient to generate a *current* proportional to the counter's state, and to add it into the emitter of Tr_4 in Fig. 5(a). Fig. 7 shows how this is done. Transistors 10, 11, and 12, like Tr_4 in Fig. 5(a), are saturating switches. Because their collector resistors are in a 4-2-1 ratio, the sum of their collector currents is proportional to the state of the counter. For a display, say, 16 digits in a row, the counter would be made up of four bistables ($2^4 = 16$) and there would be four transistors in the digital-to-current converter.

If more than one row of patterns were needed (to display more than one multi-digit number at once, for example), the last bistable of the gross X deflection counter would clock a gross Y deflection counter. This counter would have a number of states equal to the number of rows of patterns in the display and its outputs would be connected to a digital-to-current converter, whose output would be added to the emitter current of Tr_9 in the Y deflection circuit.

Intensity modulation logic

We need to know what segments of the pattern to illuminate to form each number. The 7448 i.c. provides this information. It has four inputs to accept a b.c.d. digit, and seven output lines, labelled *a* to *g* by the

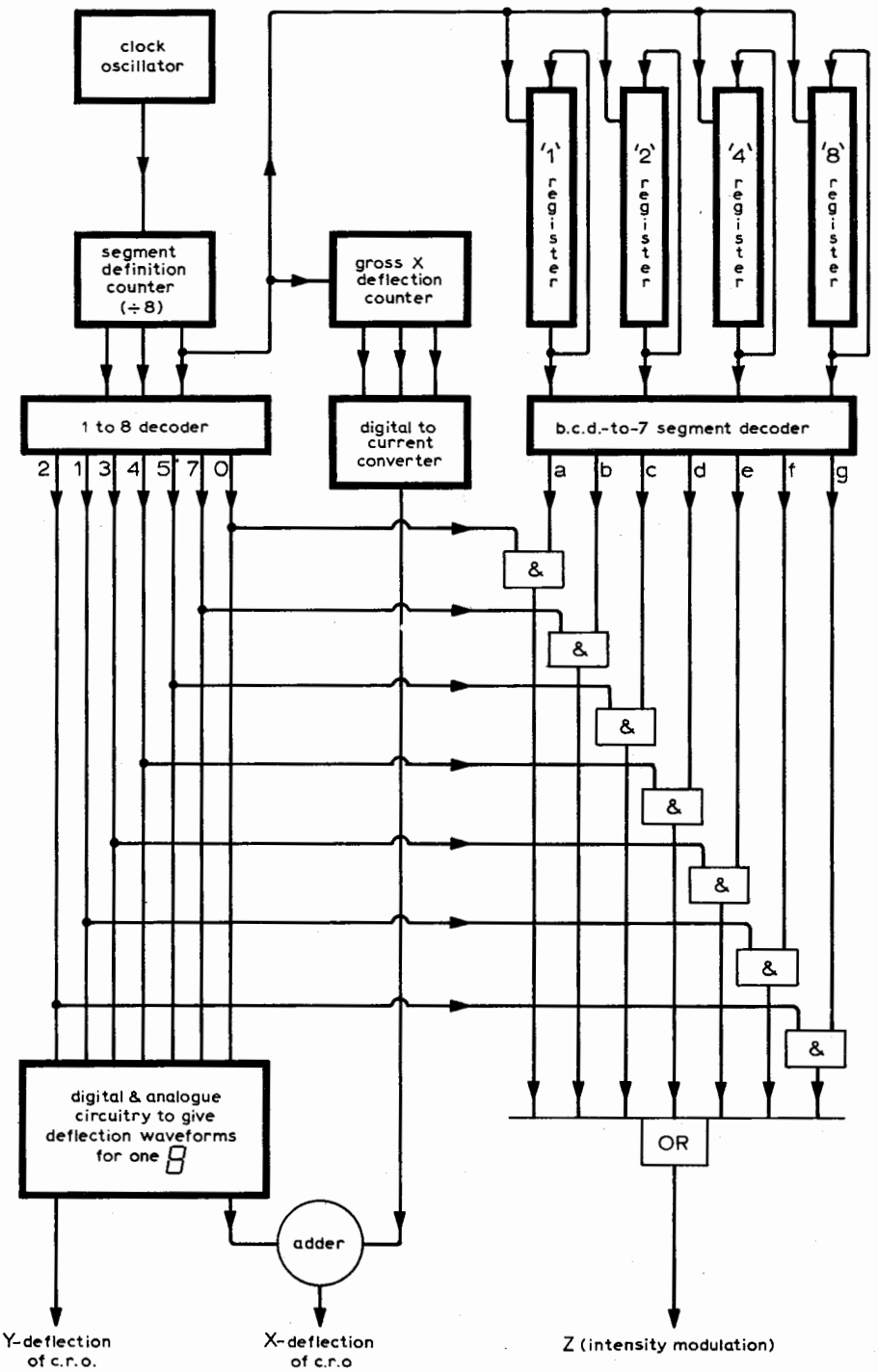


Fig. 2. Simplified block diagram of the display.

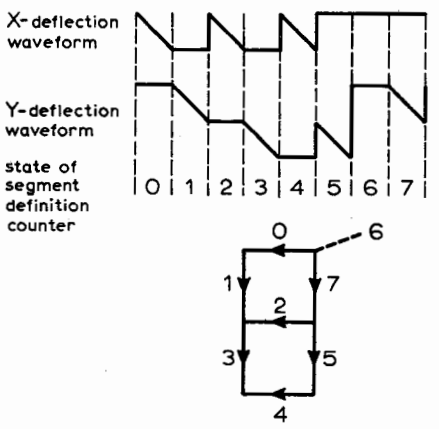


Fig. 3. Deflection waveforms for one seven-segment display.

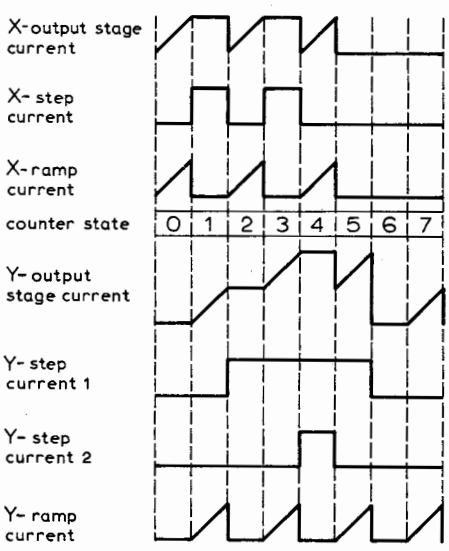


Fig. 4. Resolution of output stage currents into steps and ramps.

manufacturer, one for each segment of the pattern. The letters on the segments in Fig. 1 are those used to identify the outputs of the decoder. When a segment is to be illuminated, the corresponding output line goes to logic '1'.

The seven-segment decoder tells us which segments must be illuminated: the outputs of the 1-of-8 decoder tell us that state of the segment definition counter, and thus which segment is being drawn at any instant. Suppose a b.c.d. nine is fed into the seven-segment decoder; outputs *a*, *b*, *c*, *f* and *g* will go to logic '1'. To present a nine on the c.r.o. screen, a 'bright-up' pulse must be generated during states 0, 1, 2, 5 and 7 of the segment definition counter. Fig. 8 shows the logic circuitry needed to generate the bright-up waveform for any number: the outputs of the 1-of-8 decoder are inverted because the selected output is low.

Fig. 8 shows a practical version of the intensity modulation drive: it avoids the possible AND and OR gates using instead NAND gates, which are the most easily available type in t.t.l. The intensity modulation signal may be a.c. coupled inside the display c.r.o., so a steady level of direct voltage cannot be used as the bright-up signal. Instead, when a segment must be illuminated, a train of pulses is used. When the output of the 8-input gate is high, a train of pulses goes from the clock oscillator through a 2-input gate to the output amplifier: when it is low, the train of pulses is not passed by the 2-input gate. The segment definition counter input signal is obtained by dividing the clock oscillator frequency by 16. The effect on the display is that each illuminated segment is made up of 16 bright dots. This is hardly noticeable, and certainly not objectionable.[1]

Some c.r.os have intensity modulation applied to the c.r.t. cathode, others have it applied to the grid. The former need a negative signal to increase brightness, the latter need a positive signal, therefore the inverter shown dotted in Fig. 8 should be included for grid modulation and omitted for cathode modulation. Output transistors raise the level of the intensity modulation signal and feed it to the display c.r.o. from a low impedance.

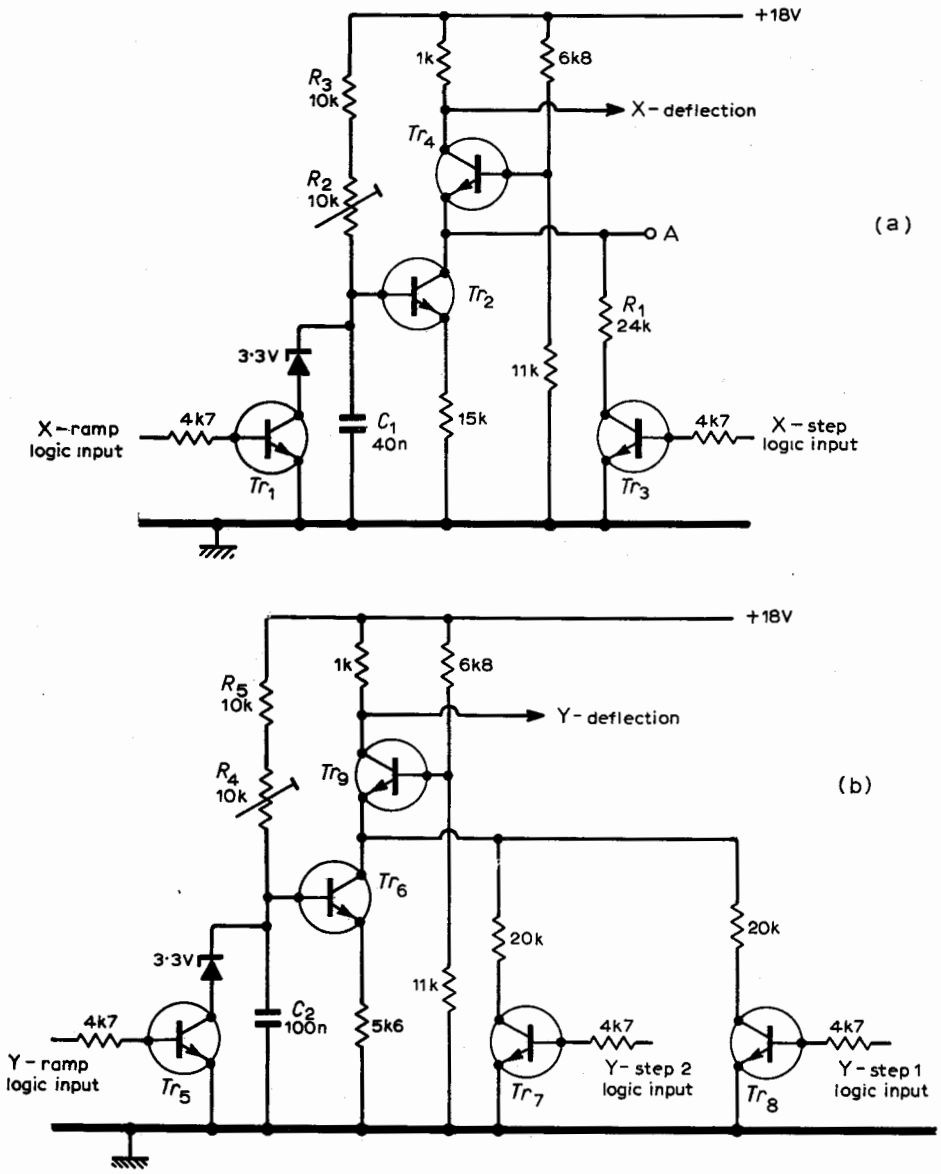


Fig. 5. (a) shows the X-deflection circuit for one seven-segment figure and (b) the Y-deflection circuit.

Storage register circuitry

Fig. 9 shows the essentials of the data storage circuitry. The top register in the diagram holds the '1' bits of all the stored b.c.d. digits, the one below holds the '2' bits, the next one the '4' bits and the bottom one the '8' bits. When the registers are clocked, the data moves to the right and a new b.c.d. digit is presented to the seven segment decoder. The outputs of the registers are connected to their inputs to re-enter the data. The c.r.t. beam must move from one character to the next at the same time as new data is presented to the decoder. As the gross X deflection counter changes state on the negative-going edge of the clock pulse and the data in the registers shift on the positive edge, an inverter must be connected between the clock terminals of the counter and the registers.

How is information put into the registers initially? This depends on the device which generates the information and so no one answer can be given to cover all requirements. Fig. 10 shows the method used in the prototype. Transistors 15 and 16 are the clock oscillator, Tr_{17} is a buffer amplifier. Pins 11 and 12 are the inputs of the 7491 shift registers. Both must be at logic '1' to enter a '1' in the register. If one of these pins is held low and the register is clocked at least eight times (the registers being eight bits long), all its bits will go to logic '0'. Switch S_1 in Fig. 10, normally open, is connected to one input of all the registers and is closed to clear the registers. Having done this, we must stop the counters to prevent the registers being clocked. This is done by opening S_2 which holds the counters in their all-zeros state. Information is entered by using S_3: there is an S_3 for each register.

While information is being displayed, all poles of S_3 are opened; the inverted output of the register is then inverted again by the two-input NAND gate and fed back to the input. When the registers have been cleared, the inverted output goes to logic '1', the gate's input goes to logic '0' and this is fed into the register. When S_3 is opened, the gate's output goes to logic '1' and this is likewise fed to the register's input. The appropriate poles of S_3 are closed to enter a digital 9 (1001 in b.c.d.)—logic '1' must be put into the eight and one registers, and logic '0' must be put into the one and two registers. This can be done by closing the S_3s of the eight and one registers, and leaving open those of the one and two registers.

Now the required information is at the input of the registers they must be clocked so that the next digit of the number to be displayed may be entered. A single clock pulse can be applied to the registers by pressing S_4 once. This is a biased change-over switch, connected to the inputs of a latch circuit using two input NAND gates to avoid multiple pulses being produced by switch bounce. When S_4 is pressed, the output of the upper gate of the latch goes from logic '1' to logic '0'. The reverse transition occurs when the switch is released, and this shifts the data in the register.

The least significant digit of the number to be displayed must be entered first: when all eight of the digits have been entered, the least significant digit will be at the right hand end of the registers. The procedure

for entering information is summarized below:

(1) close S_1 to clear registers: open S_1
(2) open S_2 to stop counters
(3) set the S_3s to enter the least significant digit (close S_3 to enter a '1', open S_3 to enter a '0')
(4) press S_4 once
(5) repeat (3) and (4) for each digit
(6) open all S_3s
(7) close S_2 to allow counters to run.

If several multi-digit numbers are to be displayed, then additional logic will be needed to connect the appropriate registers to the inputs of the b.c.d.-to-seven segment decoder, so that each number is displayed in the correct row of patterns (assuming that several rows of patterns are drawn, one for each number).

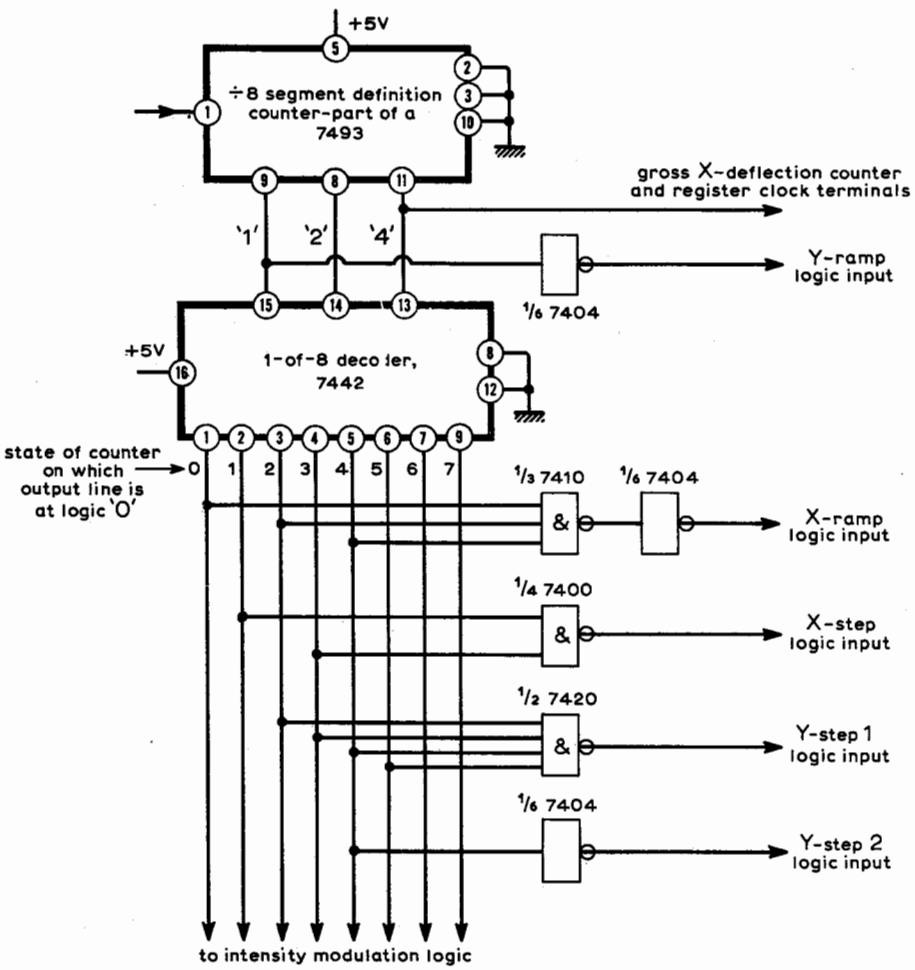


Fig. 6. Circuitry to provide logic inputs to analogue deflection circuitry.

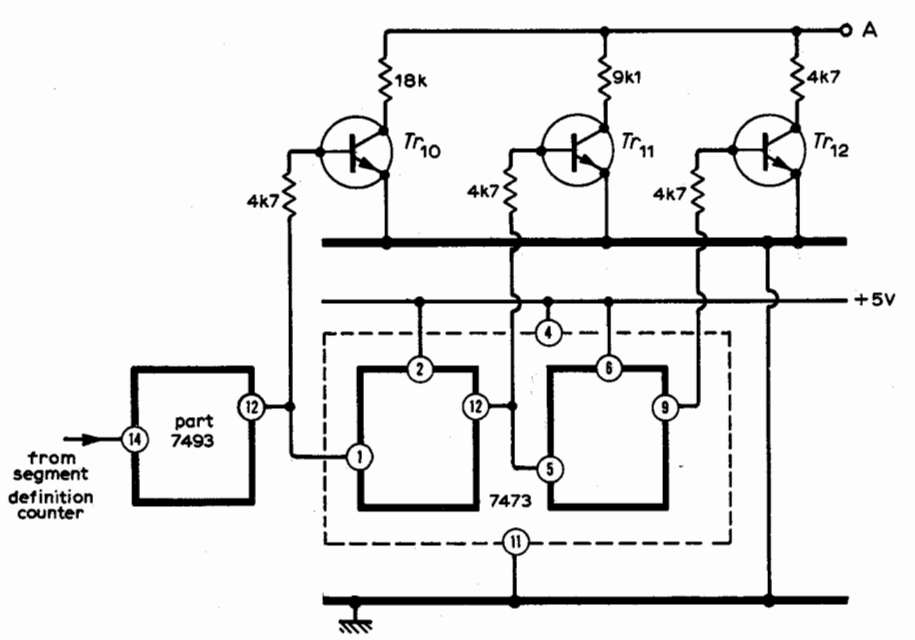


Fig. 7. Gross X deflection counter and digital-to-current converter.

Tilting the patterns

Fig. 11 shows how the deflection circuits can be modified to give tilted patterns as in Fig. 1. As the beam moves towards the bottom of the screen, the current taken by the X deflection circuit increases. This increases the voltage dropped across the 100Ω resistor in Fig. 11, and so the voltage from the output of the Y deflection circuit to earth decreases, moving the beam to the left, and thus tilting the patterns. This simple circuit has one minor disadvantage: not only does the Y-deflection affect the X-deflection, as described, but the X-deflection affects the Y-deflection, making the left-hand end of a row of patterns lower than the right-hand end. In practice, this is hardly noticeable with a full screen-width of patterns: it can be made entirely unnoticeable by adjusting the 'trace rotate' control of the display c.r.o.

Calculations

Clock oscillator. The clock oscillator must run at a high enough frequency to avoid flickering of the display. This means that the screen full of information must be scanned at least 50 times per second. In the prototype, the clock oscillator frequency is the number of rows scanned per second (50) times the division ratio of the gross X deflection counter (8) times the division ratio of the segment definition counter (8) times the division ratio of the counter connected between the clock oscillator and the input of the segment definition counter (16), i.e. about 50kHz. Choosing 1kΩ as the collector resistor of an astable multivibrator, the cross-coupling capacitor is found to be 15nF.

X deflection circuit. The c.r.o. used with the prototype has a full-screen X deflection sensitivity of 5V; thus the width of each pattern is 400mV, and the separation between the patterns and between the end patterns and the edge of the screen, is 200mV. We choose Tr_4's collector resistor to be 1kΩ; this simplifies some arithmetic, since 1mA drops 1V across 1kΩ. During the right-hand strokes of the right-hand pattern, Tr_4 passes 200μA (assuming that the c.r.o. beam is at the right-hand edge of the screen when Tr_4 is cut off, and Tr_1 is saturated). The voltage across C_1 is about 3.5V; since Tr_2's collector current is equal to the current through Tr_4, R_1 can be calculated as 14kΩ, 15kΩ being used in practice.

When the horizontal strokes of the right-hand figure are being drawn, the voltage across C_1 increases until, at the end of the stroke, Tr_4 is passing 600μA and its collector resistor is dropping 600mV. Thus the voltage across C_1 at the end of the stroke can be calculated as about 9.7 (600μA through 15Ω plus the emitter-base voltage of Tr_2). The time during which the stroke is being drawn is known (16 times the clock period, i.e. about 320μs), so the time constant of C_1, R_2 and R_3 can be calculated. It is given by $T/\ln[(V-V_a)/(V-V_T)]$ where T is the capacitor's charging time (320μs), V is the supply voltage (18V), V_0 is the initial

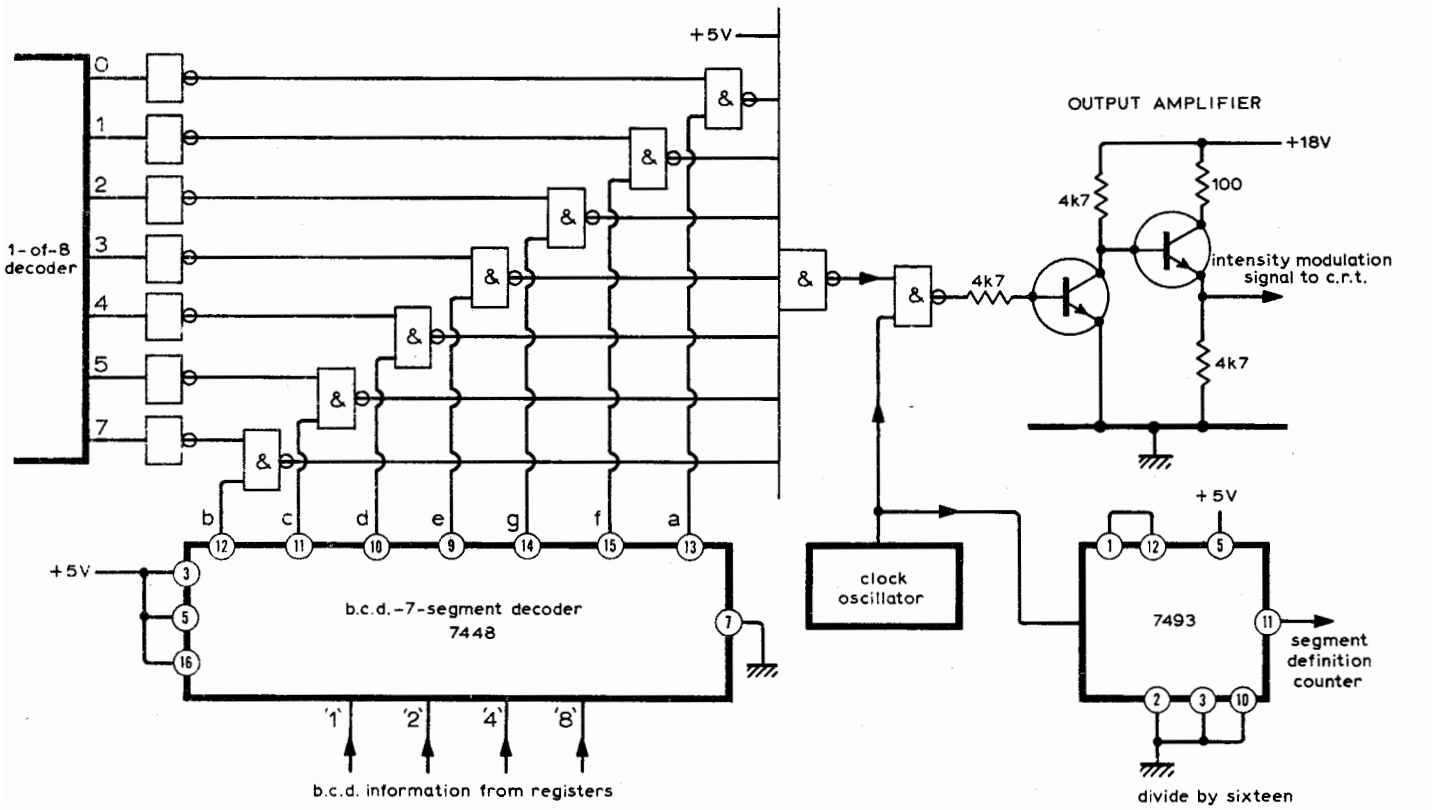


Fig. 8. Practical form of intensity modulation logic using NAND gates with gated a.c. drive to intensity modulator/output amplifier.

capacitor voltage (3.5V) and V_T is the capacitor voltage after time T (9.7V). The time constant can be calculated as 570μs and so R_2 and R_3 are 10kΩ, and C_1 is 0.04μF.

The base voltage of Tr_4 must be above 9.7V so that Tr_2 does not saturate even when C_1 is at its maximum voltage: it must be below 13V so that Tr_4 does not saturate even when 5V are dropped across its collector resistor (that is, when the left-hand pattern is being drawn). The values shown in Fig. 5(a) hold the base of Tr_4 at about 11V. Resistor R_1 can now be calculated. When Tr_3 saturates, the current through R_1 moves the beam of the c.r.o. from one side of a pattern to the other. This current must be 400μA to drop 400mV across the collector resistor of Tr_4. Since R_1 has about 10V across it, its value may be calculated as 25kΩ. The nearest preferred value, 24kΩ, may be used.

Next the values of the resistors in Fig. 7 can be calculated. When the gross X deflection counter goes from state '0' to state '1', the c.r.o. beam moves from the left-hand pattern to the next one along. For this to happen, Tr_{10} must put 600μA into Tr_4, as there is 600mV between any point on a pattern and the same point on the adjacent pattern. The collector resistor of Tr_{10} has about 10V across it, and thus its value is about 17kΩ; 18kΩ is used in practice. For the patterns to be equally spaced, the sum of the collector currents of Tr_{10}, Tr_{11} and Tr_{12} must be proportional to the state of the gross X deflection counter. This means that the collector resistor of Tr_{11} must be half that of Tr_{10}. Fig. 7 shows the nearest preferred values used.

Y deflection circuit. Assume there is a voltage drop of 500mV across the collector resistor of Tr_9 when the c.r.o. beam is at the top of a pattern. Under these conditions, Tr_5 is saturated, and thus there is about 2.8V across the emitter resistor of Tr_6. Since Tr_6 is passing 500μA, this resistor can be calculated as 5.6kΩ. When the central bar of a pattern is being drawn, one volt is dropped across Tr_9's collector resistor so there must be about 6.3V across C_2 at the end of its charging period. Using the formula already given, the time constant of C_2, R_4 and R_5 can be calculated to be 1.5ms: R_4 and R_5 is 10kΩ, and C_2 is 0.1μF. Transistors Tr_7 and Tr_8 must both pass 500μA when saturated (to move the beam vertically through the pattern) and since they have about 10V across their collector resistors, these must be 20kΩ

The transistors must be silicon types with breakdown voltages greater than 18V. BC108s were used in the prototype. The supply to every few logic circuits should be decoupled with a 0.01μF disc ceramic capacitor.

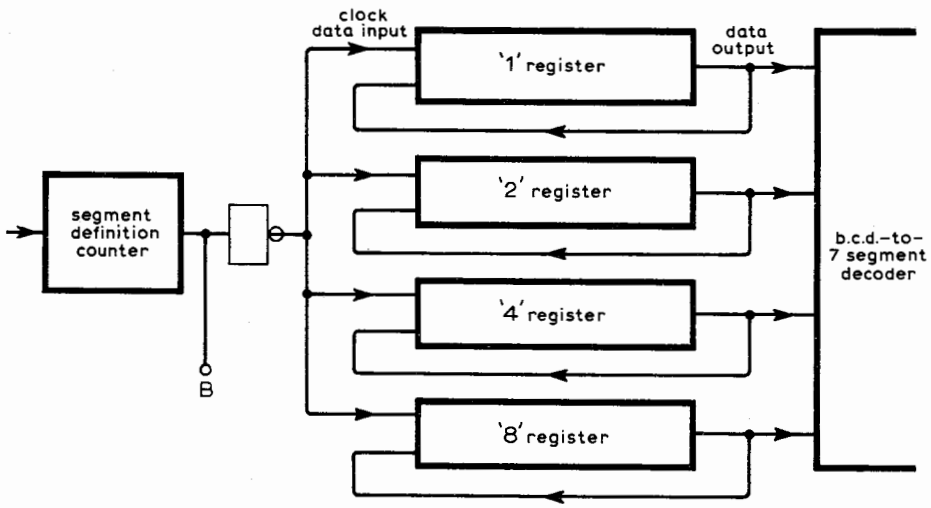


Fig. 9. Circuitry of the data storage registers.

Centring the figure one

As the figure one is made up of the two right-hand vertical segments of the pattern, it is not centred in the space between the two adjacent characters. This can be avoided by detecting a figure 1 at the end of the registers, and putting a current of the right amplitude into the emitter of Tr_4, to move the figure into the centre of the space. Fig. 12 shows how this may be done.

Possible developments

It is possible to display both letters and numbers on a 5-by-7 matrix of dots, the numbers being better shaped than those

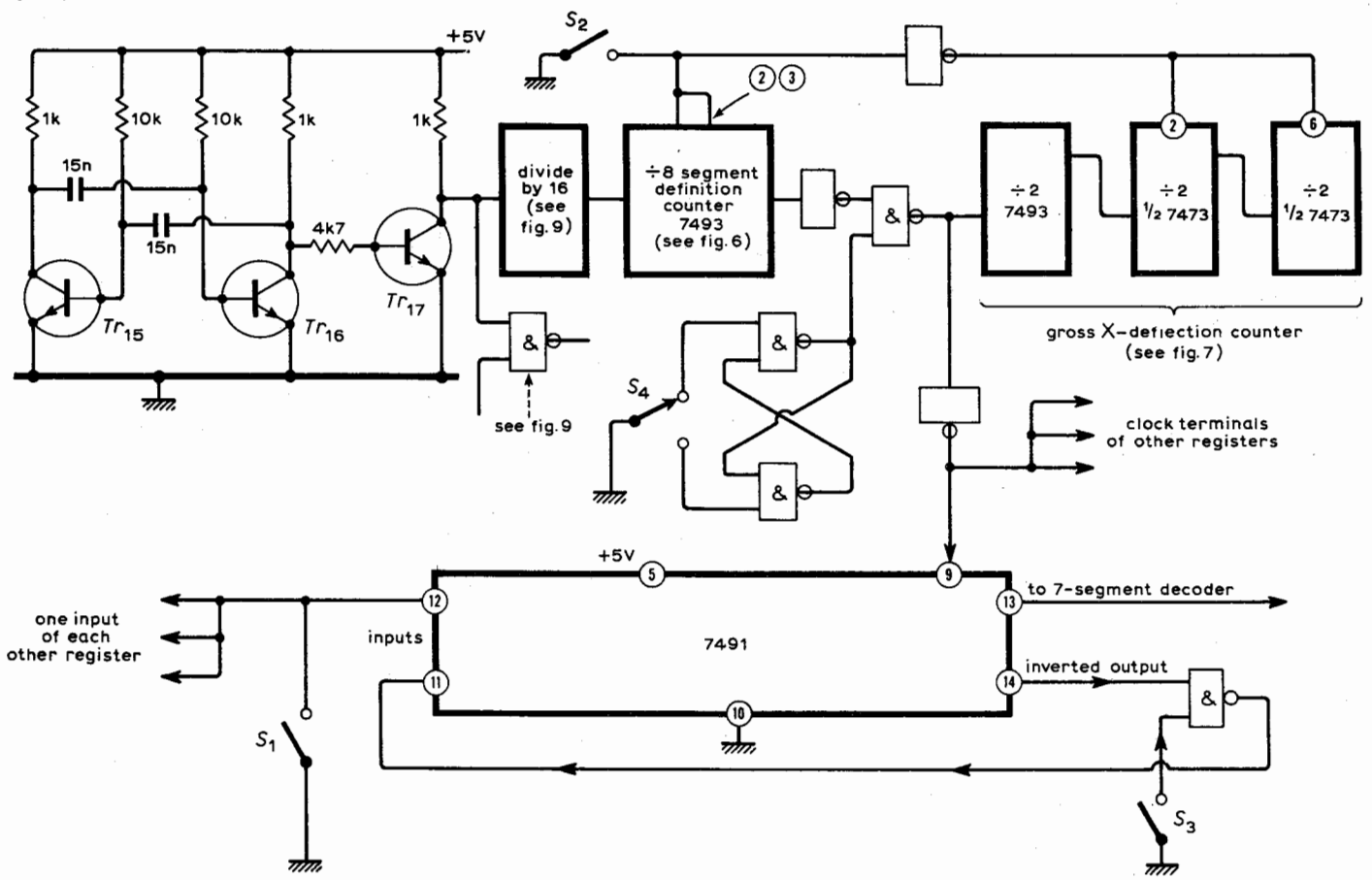
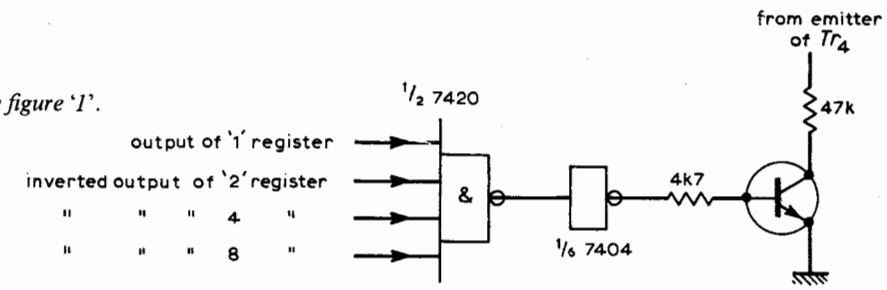


Fig. 10. Data entry in the prototype. Only one register is shown as the other three are similar. The segment definition counter and first 'divide by 2' in the gross X-deflection counter are in the same 7493 and thus clear simultaneously.

Fig. 12. Centring the figure '1'.



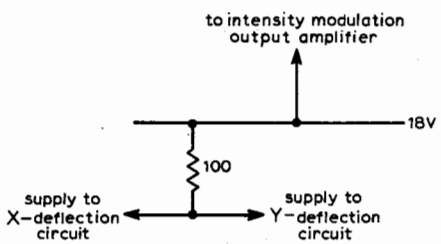


Fig. 11. Tilting the patterns.

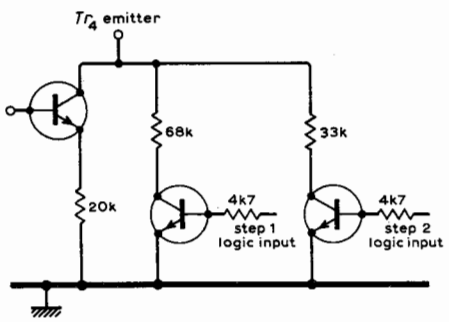


Fig. 13. X-deflection circuit for the decimal point version.

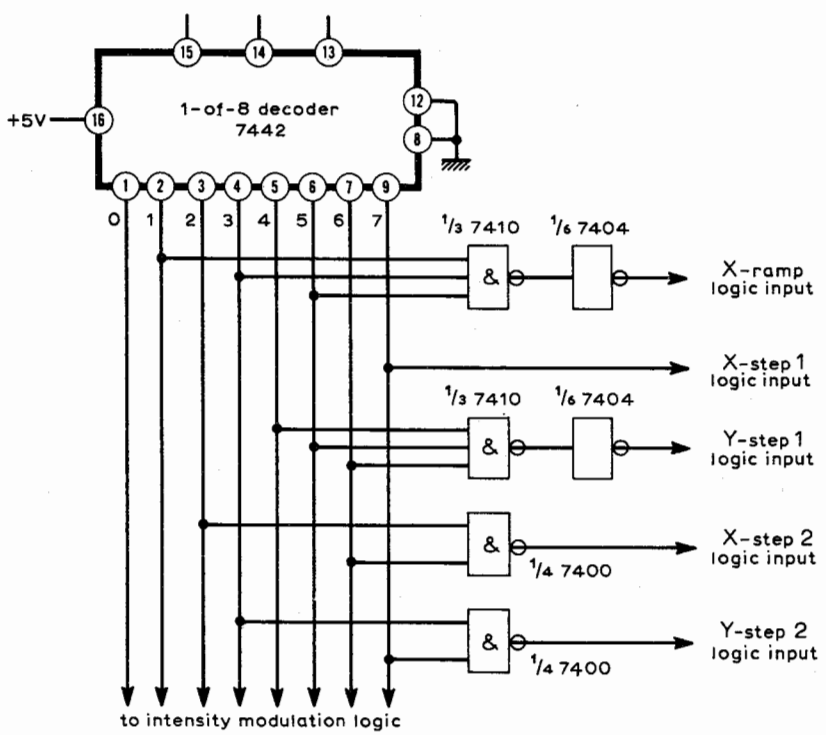


Fig. 14. Logic to control the analogue deflection circuitry in providing the decimal point.

formed on a seven-segment pattern. The blanking logic needed to form the characters on the dot matrix seems prohibitively complex to implement with t.t.l. If the m.o.s. read-only memories used to implement the logic become available to the amateur, the dot-matrix display will become attractive.

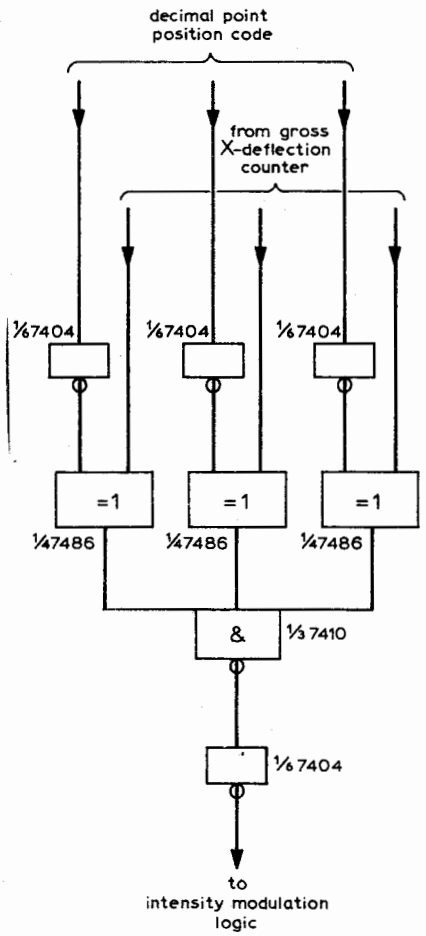


Fig. 15. Decimal point control logic comparator.

The design of the prototype was prompted by an interest in calculating machines: the bit-parallel, digit-serial arrangement of the data to be displayed fits in well with the bit-parallel, digit-serial arrangement of some of the commonly used b.c.d. adder/subtractors, but other register layouts can be used if necessary. For example, some adder/subtractors need bit-serial, digit-serial layouts: these can be accommodated by running the data into a four-bit register until it contains one b.c.d. digit, memorizing the register outputs in four latches, and presenting the output of the latches to the inputs of the b.c.d.-to-seven segment decoder. When the next digit has entered the four-bit register, it is entered into the latches, and so on for all the digits.

Appendix—Adding a decimal point

The addition of a decimal point to the display requires two things: that a decimal point is added to each basic seven-segment pattern and that additional logic circuitry is provided to turn on the decimal point which is to be used.

In the system already described, the c.r.o. beam is stationary at the top right-hand edge of the pattern during state 6 of the segment definition counter. A different route of the c.r.o. beam round the pattern can give a stationary spot at the bottom right-hand corner. An added transistor in the X-deflection circuit can then move this stationary point to the right, giving a decimal point to the right of the lower edge of the figure. The following table of segment letters and segment definition counter states gives a stationary point in the correct position.

Fig. 13 is the modified X deflection circuit with an added switch transistor to move the c.r.o. beam to the right during state 7 of the segment definition counter, i.e. during the drawing of the decimal point. Fig. 14 is the modified logic circuitry giving digital inputs

State of segment definition counter	Segment drawn
0	*c*
1	*g*
2	*e*
3	*d*
4	*b*
5	*a*
6	*f*
7	decimal point

to the X and Y deflection circuitry (while the Y deflection circuit is unaltered, the transistors are switched in a different sequence).

Additional intensity modulation logic is needed to control the decimal point. This is done by comparing a binary word with outputs of the bistables in the gross X deflection counter. When the word is identical with the outputs, the decimal point is illuminated. This is done by sending a 'bright-up' pulse to the oscilloscope during state 7 of the segment definition counter. Because the gross X deflection counter is in state 000 while the extreme right-hand pattern is drawn and in state 001 during the next pattern and so on, the extreme right decimal point will be turned on by the binary word 000, the next one by the binary word 001 and so on.

Fig. 15 is the circuit used to compare the binary word with the gross X deflection counter output. The circuit shown accepts a 3-bit word, sufficient for a display with $2^3 = 8$ figures. Larger numbers of figures can be accommodated by adding an inverter, an exclusive-OR gate and an input to the NAND gate for each bit of the binary word.

Fig. 16 is the modified intensity modulation circuit. It has an input from the decimal point control comparator and also differs from Fig. 8 in the relationship between the segment being drawn and the state of the segment definition counter which has been changed to allow the stationary dot state to occur at the bottom right-hand corner of the pattern.

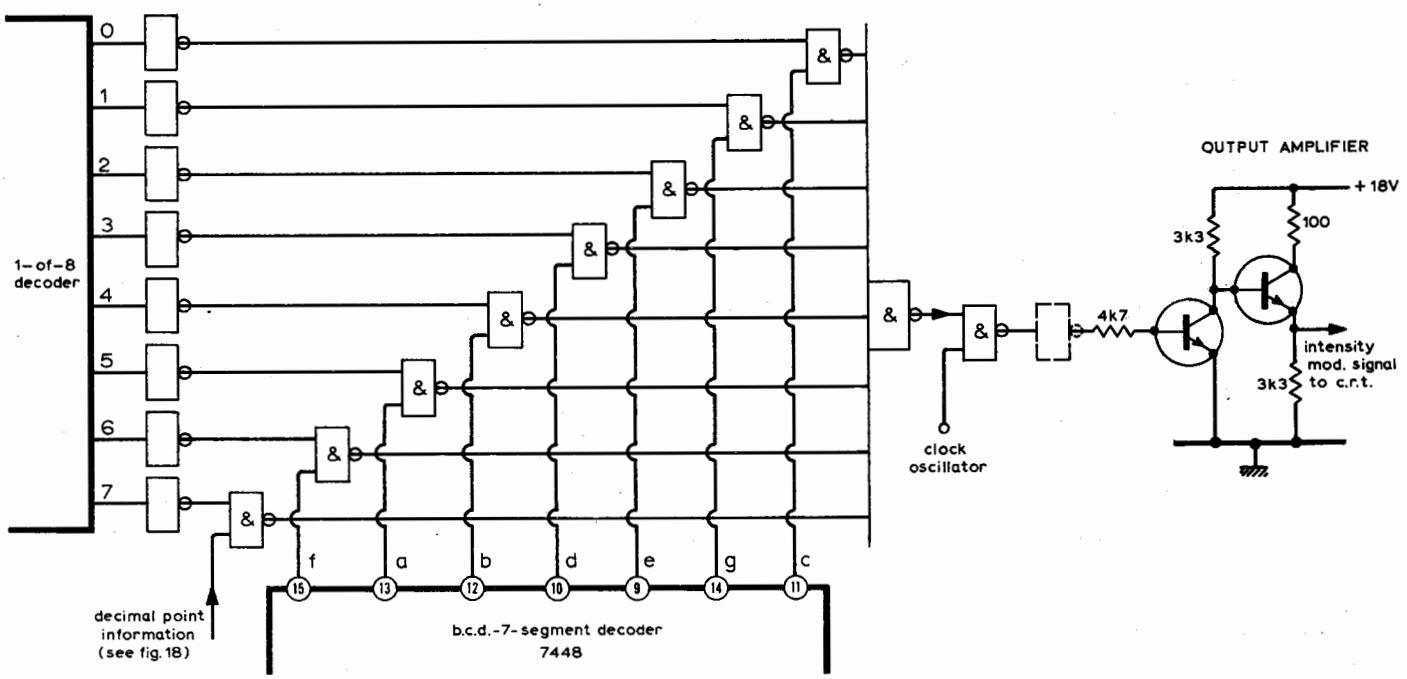
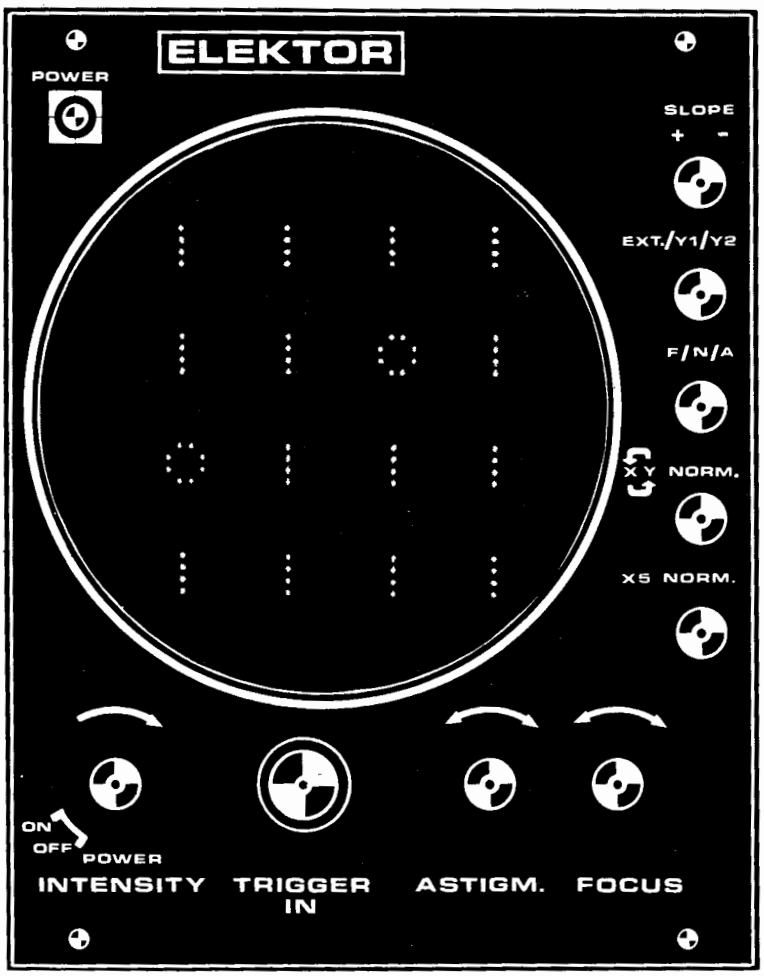


Fig. 16. Intensity modulation logic for the decimal point version.

M.G. Fishel

digisplay



This logic tester displays the states of sixteen binary signals (either '0' or '1') simultaneously, in the convenient form of a 4 x 4 matrix on an oscilloscope screen. The facility for automatic Karnaugh mapping is of particular interest, although some other uses of the tester are described.

The ways of testing binary logic signals are many and various with the involved test equipment being more or less complicated. This article describes a circuit, dubbed the 'Digisplay', which tests the states of sixteen binary signals and displays the results on an oscilloscope screen in the particularly convenient form of a four-by-four matrix of '0' and '1' characters. The Digisplay, which uses conventional TTL throughout, may be used in a variety of ways, vis:

(1) to test integrated circuits;
(2) to display logic states in the form Karnaugh maps;
(3) as a programmed 16-bit pulse generator.

An internal clock pulse generator of fixed frequency is used in the circuit, but a facility is provided for connecting an external clock should a variable frequency be required.

The circuits which generate the characters '0' and '1' and place them in matrix form on the screen are described first and then the various uses of the Digisplay are explained.

Block Diagram

The Digisplay is shown in block diagram form in figure 1. In order to obtain a display on the screen of '0' and '1' characters arranged in a matrix, the circuit performs two main functions. The first is to produce horizontal and vertical scanning information at the oscilloscope terminals to enable a '0' or a '1' to be written on the screen; the second is to superimpose on these signals further scanning information to position the character within the matrix square required. The '0' character is written as a circle of eight dots and the '1' is derived from the '0' by supressing the horizontal scanning signal.

The character generating signals are derived from clock pulses which are fed into a divide-by-eight counter. The three outputs, L, M and N are re-encoded into the signals E, F, G and H of which E and F are fed to a digital-to-analogue (D/A) converter whose output supplies the vertical scanning signal to the oscilloscope, and G and H are fed to a second D/A converter which supplies the horizontal scanning signals. It can therefore be seen that each state of the divide-by-eight counter corresponds to one of the eight dots in the character. (The details of the signal coding are given later.)

The matrix generating signals are derived from a divide-by-sixteen counter which is driven by the N output from the divide-by-eight counter. Thus each state of this divide-by-sixteen counter corresponds to one of the squares in the matrix. The four outputs from the counter, W, X, Y and Z, are also fed to the two D/A converters as shown, where they are superimposed on the voltages obtained from E, F and G, H such that each character is positioned in its correct square in the matrix.

The outputs W, X, Y and Z are also used to produce the test signals A, B, C and D. These signals, which are used to drive the circuit under test, are coded in such a way that Karnaugh maps may be realised on the display, (see 'Karnaugh Mapping'). For each state of the drive signals ABCD the multiplexor produces the result $\overline{S}$ (the inverse of the selected input signal S) which is used to inhibit the horizontal scanning signal whenever a '1' needs to be displayed.

Each part of the Digisplay circuit is now described in detail.

Tracing the Characters

Figure 2 shows the path along which the tracing beam steps in order to produce the characters. In each dot position (numbered 0-7) the tracing beam pauses for the duration of one clock pulse. The figure also shows how the '1' is derived from the '0' with the horizontal movement inhibited. (It is true that this will cause the '1' to be slightly brighter than the '0', but in practice this is barely perceptible and can be ignored.)

The apparent misalignment of the two characters within a square is corrected as described later. Each character is traced out more than a hundred times each second, so that a '0' appears as a circular arrangement of eight dots and a '1' as a vertical line of four dots. The horizontal and vertical co-ordinates of the eight positions, which correspond to the scanning information which must be supplied by the two D/A converters, are given in the tables of figure 2.

The circuit which controls the movement

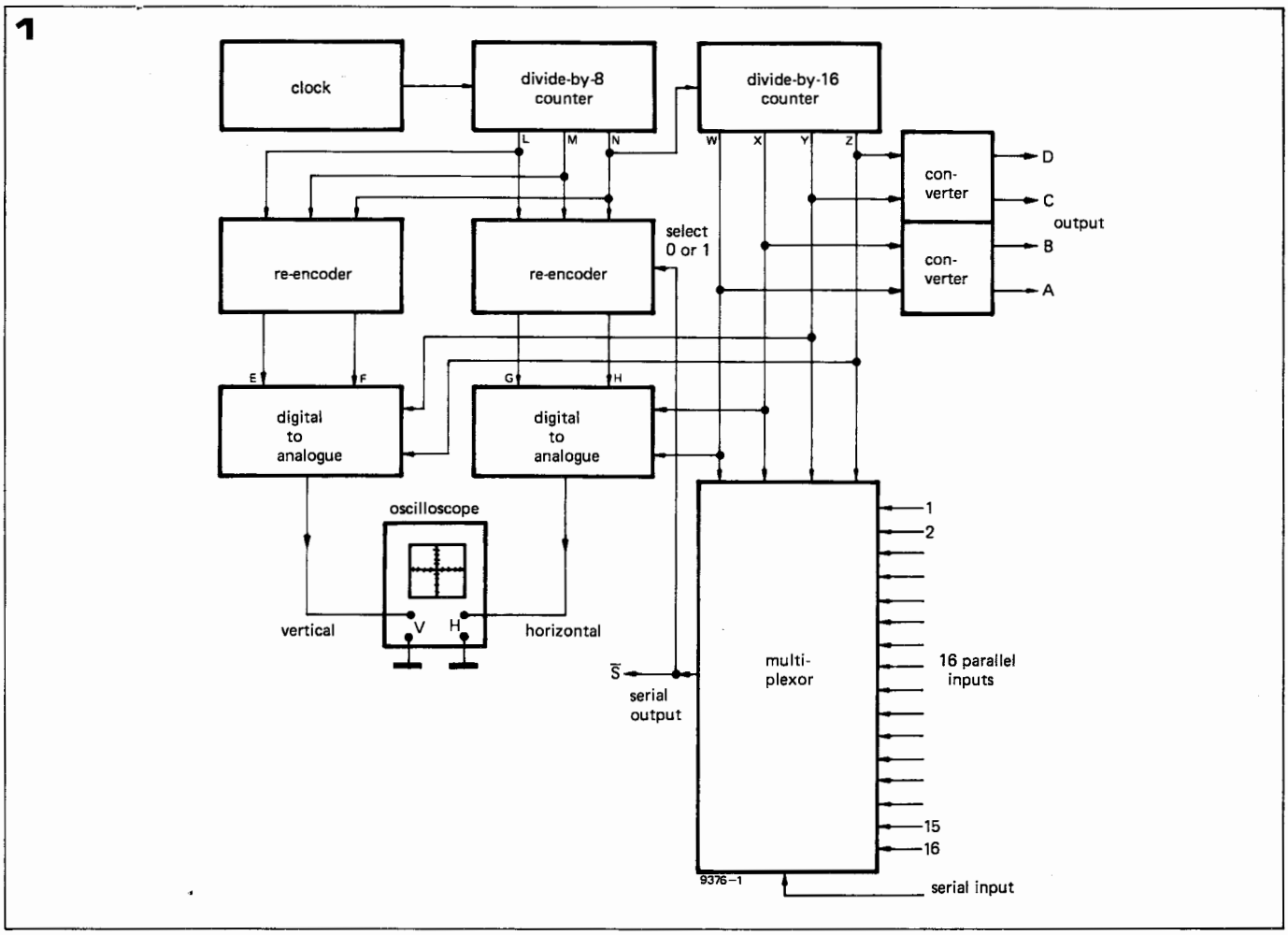


Figure 1. Block Diagram. Clock pulses are re-encoded to drive logic circuit under test and to produce characters '0' and '1' on the screen. Circuit results are decoded to select either '0' or '1'.

of the tracing beam is shown in figure 3 and operates as follows: a simple multivibrator comprising two NAND gates, generating a signal of approximately 20 kHz, feeds clock pulses into a divide-by-eight counter consisting of three flip-flops in an IC7493 (the fourth flip-flop is left unused). The counter outputs L, M and N are used as inputs to a re-encoding circuit as previously described. The re-encoding which takes place is shown in the table of figure 4 and corresponds to the following equations:

$E = L$

$F = \overline{LMN} + \overline{L}MN + L\overline{M}N + LM\overline{N}$

$G = \overline{\overline{M} \cdot \overline{S}} = M + S$

$H = \overline{\overline{N} \cdot \overline{S}} = N + S$

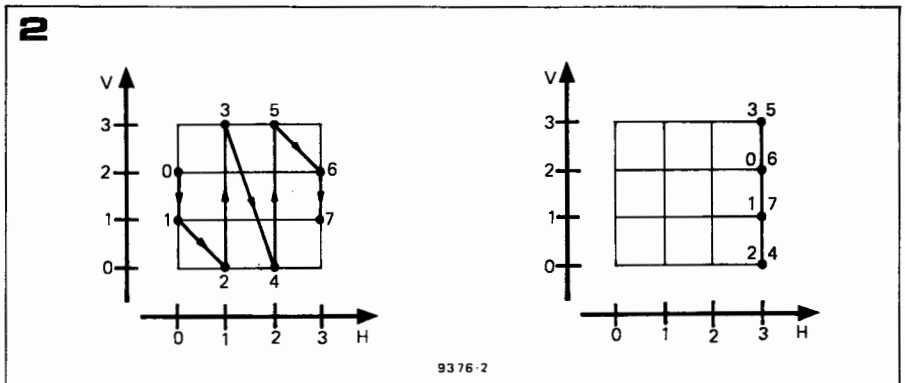
where $\overline{S}$ is the multiplexor output. The re-encoding circuit shown in figure 5 consists of ICs 7400, 7410 and 7420. The two remaining NAND gates in the IC 7400 are used in the clock pulse generator (see figure 3). The $\overline{S}$ signal inhibits G and H in order to trace a '1', i.e. when $\overline{S}$ is '0'.

Plotting the Matrix

The path the tracing beam takes through the matrix while drawing sixteen characters is shown in figure 6. The path is achieved by adding horizontal and vertical staircase voltages to the character scanning signals. Figure 7 shows the circuit, which consists of a divide-by-sixteen counter IC 7493 whose outputs W and X are input to two gates in an IC 7486 and outputs Y and Z are input to two gates in an IC 7404. The outputs from these gates are used respectively to produce the horizontal scan after each character and the vertical scan after each row of four characters.

Some types of oscilloscope show a deflection to the left for a positive voltage at the horizontal input terminal,

Figure 2. Path followed on oscilloscope screen by character tracing beam. Tables show horizontal and vertical scanning co-ordinates.



CHARACTER 0

DOT NUMBER	HOR. CO-ORD.	VERT. CO-ORD.
0	0	2
1	0	1
2	1	0
3	1	3
4	2	0
5	2	3
6	3	2
7	3	1

CHARACTER 1

DOT NUMBER	HOR. CO-ORD.	VERT. CO-ORD.
0	3	2
1	3	1
2	3	0
3	3	3
4	3	0
5	3	3
6	3	2
7	3	1

which would present a mirror-image matrix. This may be corrected in the Digisplay simply by strapping the input I of the two EXOR gates to earth, as shown.

Digital-to-Analogue Conversion

The D/A converters used in the Digisplay are of the simple resistor type as high accuracy is not required. The two circuits are shown in figure 8. The currents through the resistors are added algebraically so that a potential difference appears across the summing resistors R_{hor} and R_{vert}. These voltages are applied directly to the horizontal and vertical scanning input terminals of the oscilloscope, as shown.
Resistor R_S is present to add a small voltage to the horizontal scan whenever $\overline{S}$ is '1' i.e. whenever a '0' is required on the screen. This displaces the written '0' so that it is aligned with the '1' (see figure 9).

Input/Output Correlation

The circuit which correlates each of the Digisplay inputs (E0-E15) with one square of the matrix, is shown in figure 10. The WXYZ signals which control the plotting of the matrix, also drive the multiplexor IC 74150 so that the $\overline{S}$ signal is in synchronism with the matrix position. The multiplexor also has an over-riding serial strobe input.

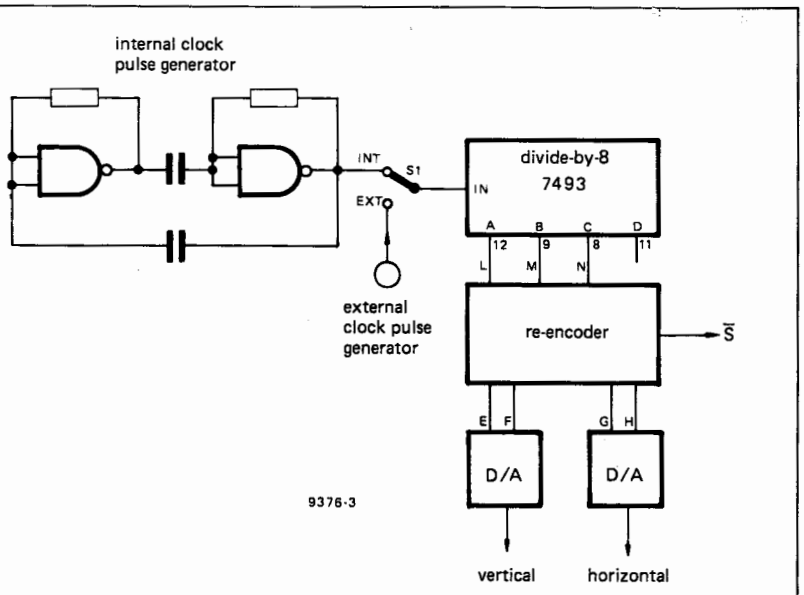
Karnaugh Mapping

The signals ABCD which are used to drive the circuit under test are coded to enable Karnaugh maps to be displayed. The layout of a Karnaugh map for inputs A, B, C and D is given in figure 11. It can be seen that two adjacent matrix squares (vertically as well as horizontally) never differ by more than one element. The conversion used in the Digisplay is given in the table of figure 12, where the circuit which performs the conversion is also shown.

The Complete Circuit

The complete circuit diagram is given in figure 13. The internal (N1, N2), or

3



4

character	IC7493				re-encoder output		
S	N	M	L	H	G	F	E
0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	1
0	0	1	0	0	1	0	0
0	0	1	1	0	1	1	1
0	1	0	0	1	0	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	1	1	0
0	1	1	1	1	1	0	1
1	0	0	0	1	1	1	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	1	1
1	1	0	0	1	1	0	0
1	1	0	1	1	1	1	1
1	1	1	0	1	1	1	0
1	1	1	1	1	1	0	1

Figure 3. Two NAND gates in tandem, forming multivibrator clock pulse generator driving character tracing circuits.

Figure 4. Table of binary signals from re-encoder to trace either '0' or '1'.

Figure 5. Re-encoder circuit producing signals as shown in table of figure 4.

Figure 6. Path taken by beam to plot matrix.

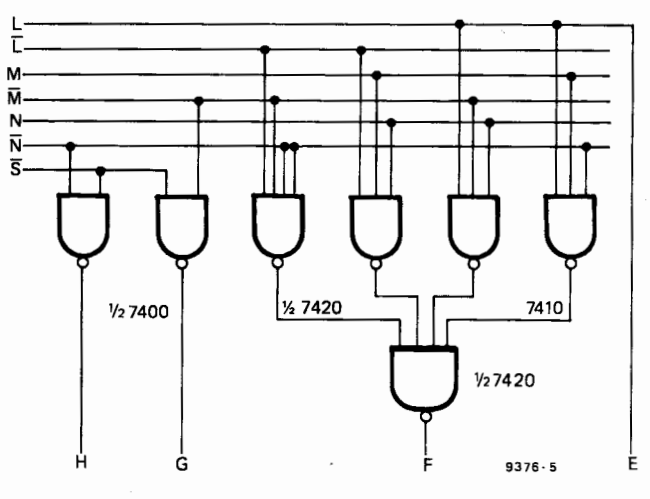
Figure 7. Matrix-plotting circuit, also showing image inversion.

Figure 8. Digital-to-analogue converters consisting of resistor networks.

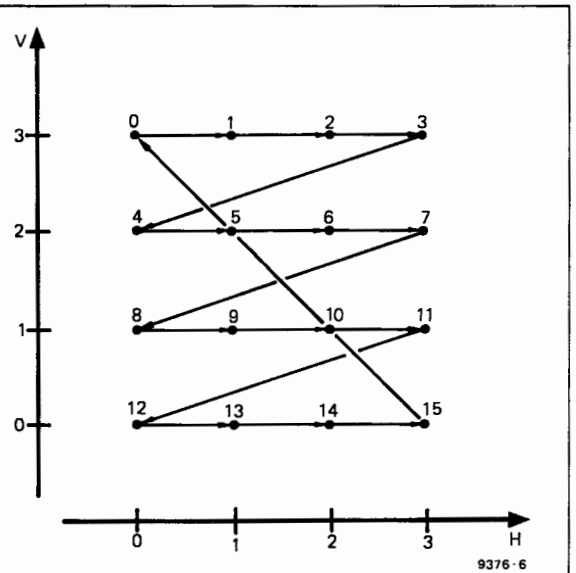
Figure 9. Character alignment.

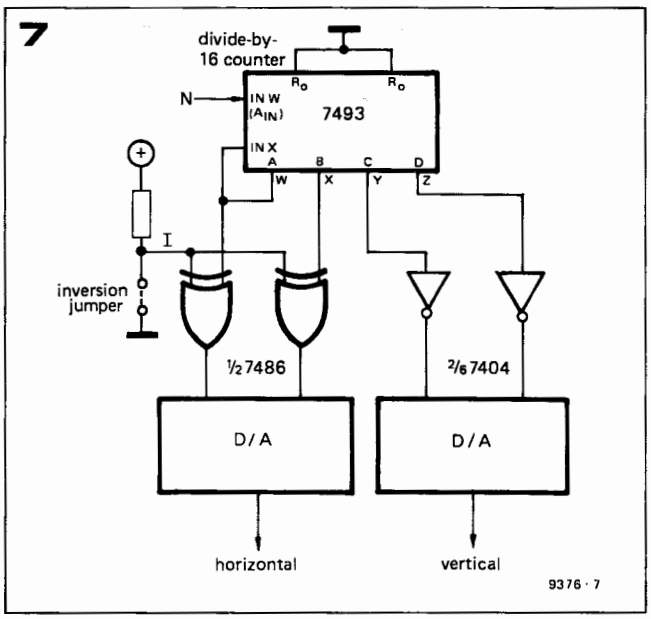
Figure 10. Multiplexor IC 74150 circuit diagram and pin connections.

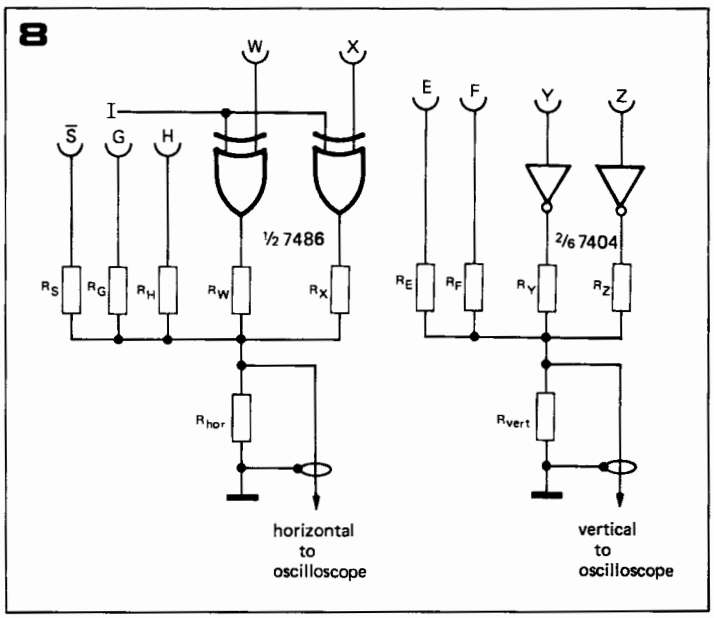
5



6







the external, clock pulse generator drives the counters via the INT/EXT switch and the diode clippers. The divide-by-eight counter consisting of three of the four flip-flops of IC6 supplies the pulses L, M and N which are re-encoded as previously described by N3-N9 and N14-N16 to drive the two D/A converters supplying the horizontal and vertical character tracing voltages.

Signal $\overline{S}$ is input to N3 and N4 to determine whether a '0' is drawn or the '1' derived from it. It also supplies the small additional voltage required to align the '0' with a '1'.

The divide-by-sixteen counter IC7 is driven by the N pulse from IC6, and it supplies pulses W and X for the horizontal matrix plotting and pulses Y and Z for the vertical, via N10, N11 and N17, N18 respectively. N10 and N11 are also used to correct image inversion where necessary.

After D/A conversion as described the matrix plotting staircase voltages are added to the character generating voltages with the resulting voltages appearing across R_{hor} and R_{vert}.

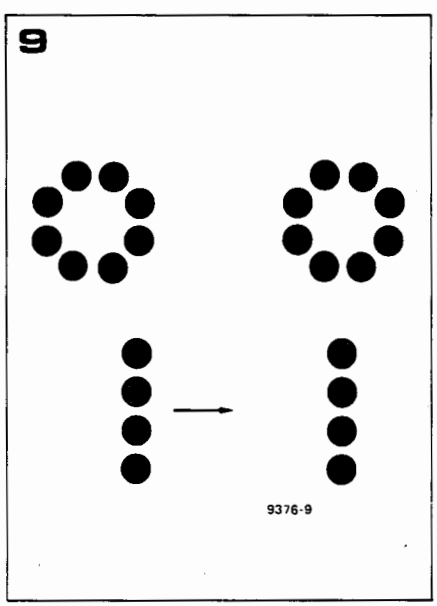
The signals W, X, Y and Z are also used to drive the multiplexor, so that synchronism is achieved between the test signals A, B, C and D and the inputs to the Digisplay. The EXOR gates N12 and N13 perform the conversion of WXYZ into the sequence ABCD required for Karnaugh mapping.

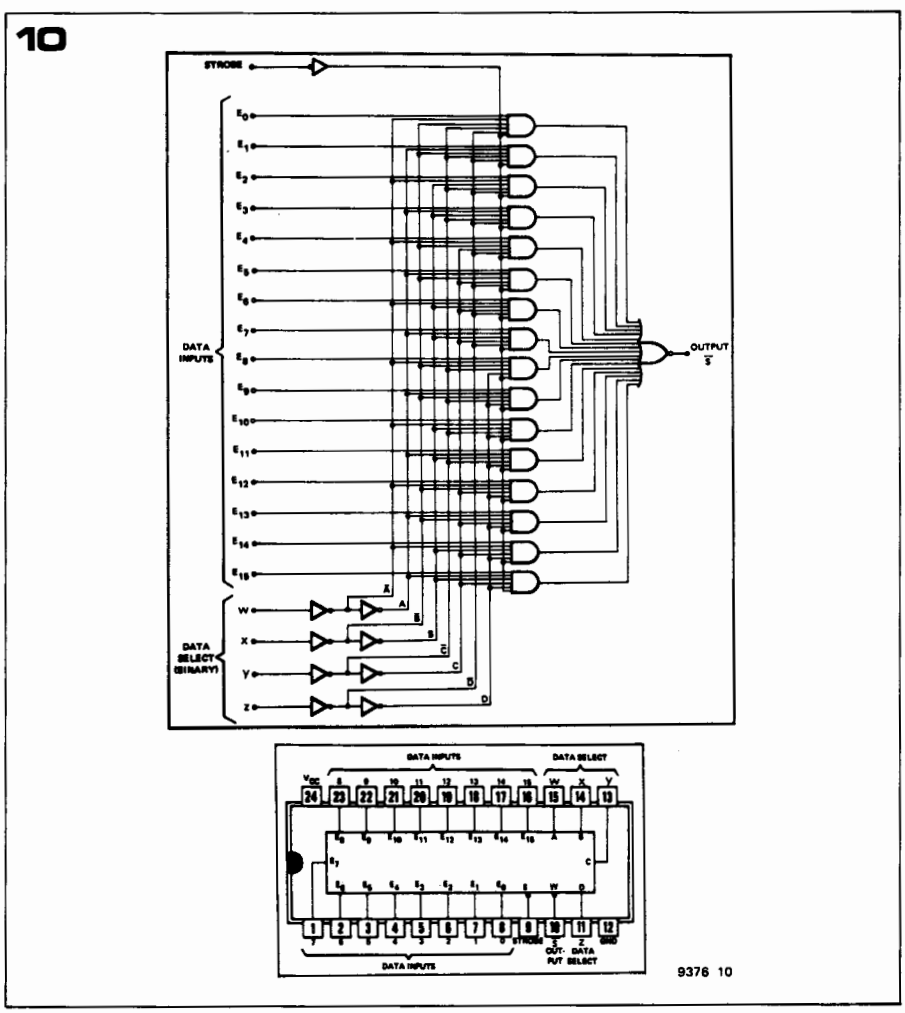
The Digisplay requires only a single 5 volt power supply.

Applications

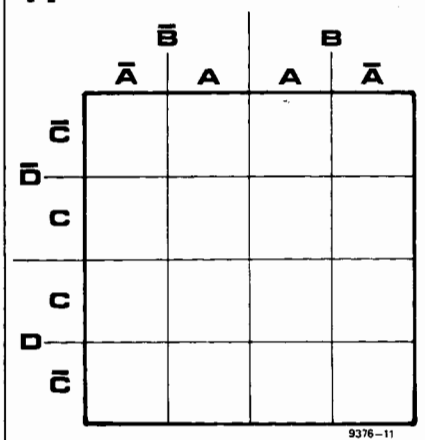
The Digisplay may be used to test ICs by making the following simple connections: connect the IC ground to the Digisplay ground; connect the Digisplay horizontal and vertical outputs to the corresponding terminals of the oscilloscope; connect the sixteen multiplexor inputs of the Digisplay to the corresponding IC pins (for which 'testclips' would be convenient). The internal clock of the Digisplay is used. No special precautions are necesarry, although it must be remembered that these connections will augment fan-outs by one, but this should not present problems in most cases.

The matrix on the screen will now dis-





11



play the state of each pin of the IC as either a '0' or a '1'. Read the matrix from left to right, top to bottom for pins 1 to 16 respectively. Open-circuit inputs and broken connections will show as logic '1's, as is usual for TTL circuitry.

Figure 14 shows an example of an IC 7442 binary-coded-decimal to decimal decoder with binary five (0101) applied to its inputs, and the resulting display.

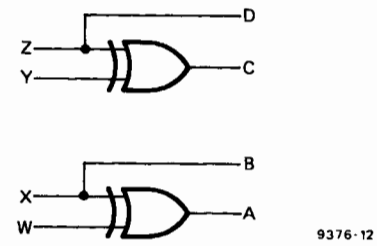
The Digisplay may be used for automatic Karnaugh mapping because of the way test signals ABCD are coded. Karnaugh maps are particular forms of truth tables, arranged in a standard way to permit easy verification of logic operations and rapid diagnosis of faults. The layout of a Karnaugh map for logic operations involving four variables has already been given in figure 11. In such a map the variables refer to squares within the matrix such that two adjacent rows or columns never differ by more than the inversion of one of the variables. The top and bottom rows are considered to be adjacent, likewise the extreme left and right columns. By convention, the states of the variables are written round the outside of the matrix, and the logical result of each state of the variables is written as '1' or '0' (as appropriate) in the relevant square. If two adjacent squares (either vertically or horizontally) show the result '1' this indicates that the variable and its inverse labelling those two adjacent squares is redundant. This is best illustrated with an example. Figure 15 gives a Karnaugh map for the 'g' output of an IC 7447 binary-coded-decimal to 7-segment decoder, with inputs ABCD from the Digisplay. With reference to the circuit diagram, the 'g' output is derived in the following manner: two AND gates have their outputs inverted and ANDed to give

$$\overline{ABC} \cdot \overline{\overline{BCD}} \quad \ldots\ldots\ldots (1)$$

at the input of the final gate. Using De Morgan's theorem the above may be rewritten

$$\overline{ABC} + \overline{BCD} \quad \ldots\ldots\ldots (2)$$

12



W	X	Y	Z	A	B	C	D	position in the matrix
0	0	0	0	0	0	0	0	1
1	0	0	0	1	0	0	0	2
0	1	0	0	1	1	0	0	3
1	1	0	0	0	1	0	0	4
0	0	1	0	0	0	1	0	5
1	0	1	0	1	0	1	0	6
0	1	1	0	1	1	1	0	7
1	1	1	0	0	1	1	0	8
0	0	0	1	0	0	1	1	9
1	0	0	1	1	0	1	1	10
0	1	0	1	1	1	1	1	11
1	1	0	1	0	1	1	1	12
0	0	1	1	0	0	0	1	13
1	0	1	1	1	0	0	1	14
0	1	1	1	1	1	0	1	15
1	1	1	1	0	1	0	1	16

13

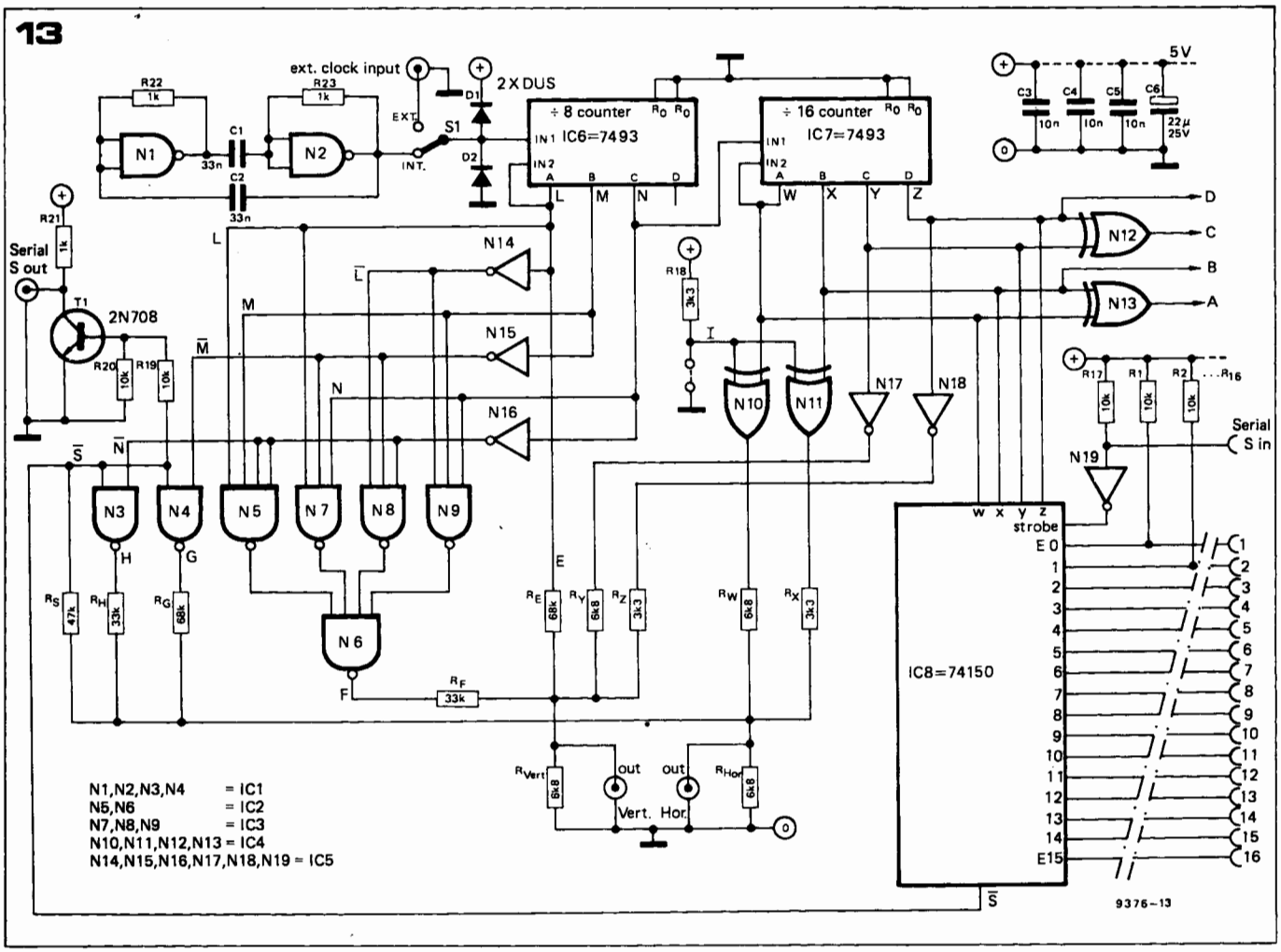


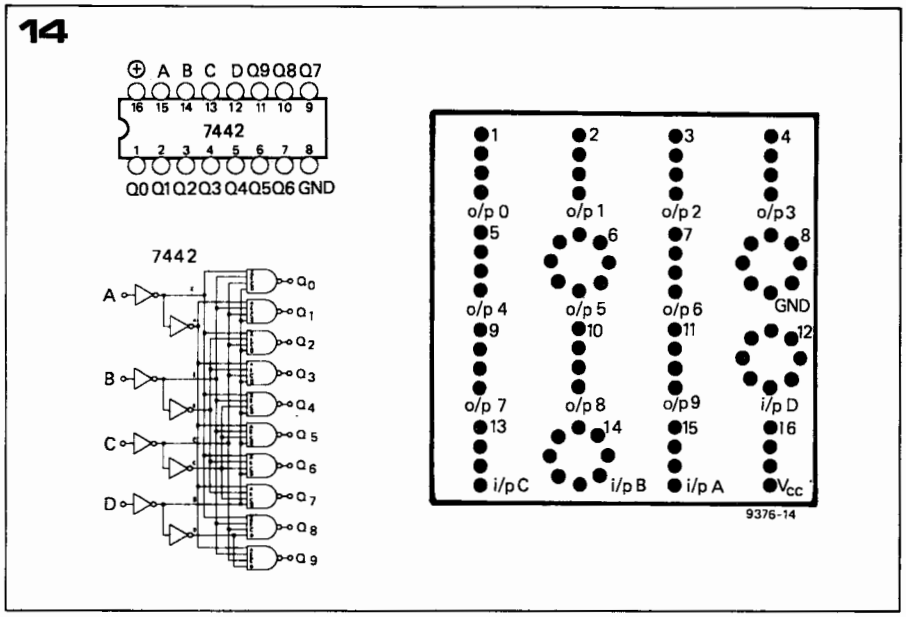
Figure 11. Layout of a four-variable Karnaugh map, which could be used as transparency overlay on screen.

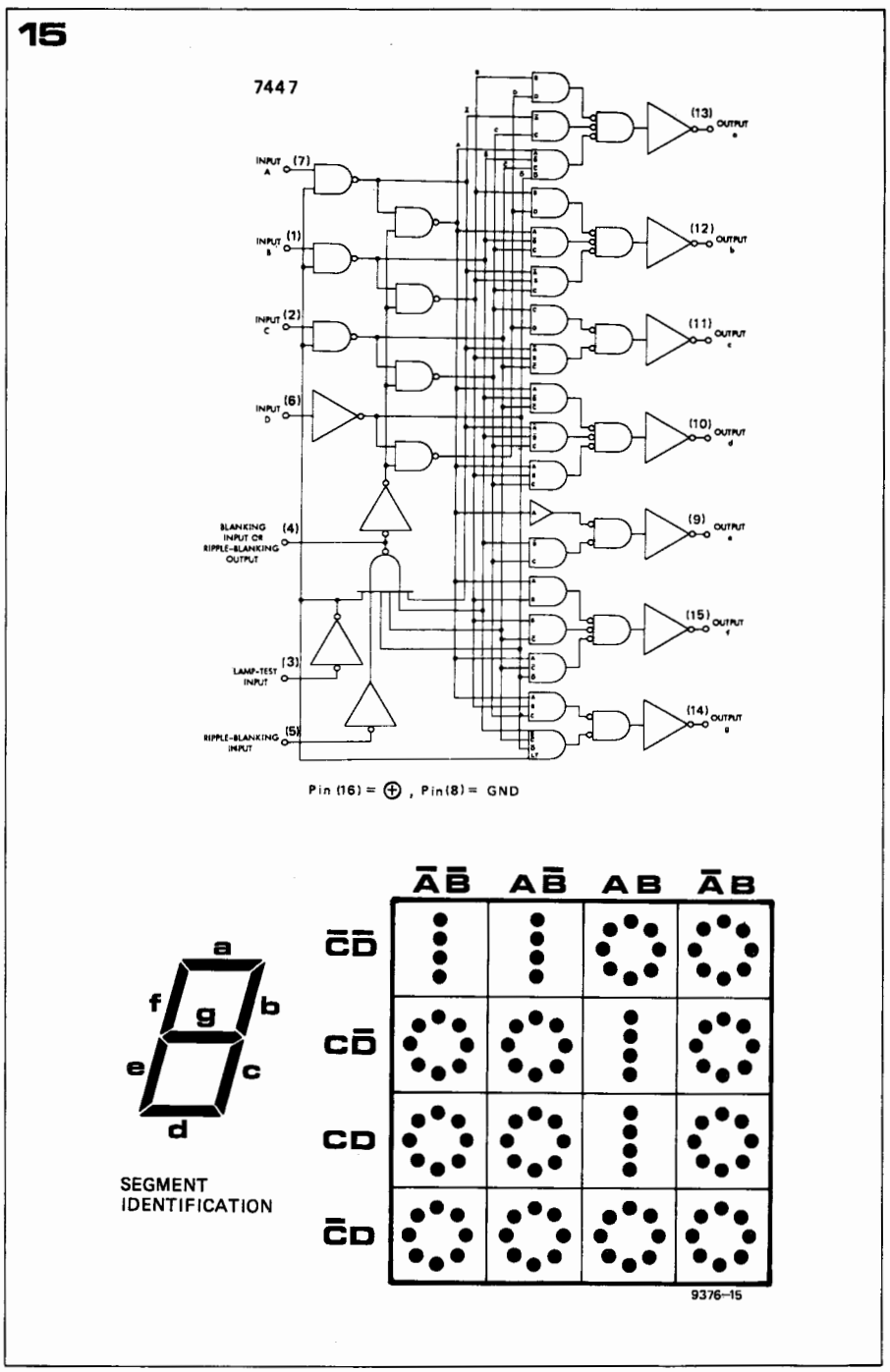
Figure 12. Binary pulse sequence W, X, Y, Z is converted to A, B, C, D required for Karnaugh mapping.

Figure 13. Complete circuit diagram.

Figure 14. Example display obtained of four inputs and ten outputs of IC 7442 BCD-decimal decoder with binary 5(0101) applied at input.

Figure 15. Karnaugh map for 'g' output of IC7447 BCD to 7-segment decoder obtained on oscilloscope screen by means of the Digisplay.





The inversion by the final gate gives the output 'g' of

$$ABC + \overline{B}\overline{C}\overline{D} \quad \ldots\ldots\ldots (3)$$

Now consider the Karnaugh map. The top row has two adjacent '1's which pinpoint the variable A as redundant for this term, i.e.

$$\overline{C}DA\overline{B} + \overline{C}\overline{D}A\overline{B} = \overline{C}\overline{D}\overline{B} \quad \ldots\ldots (4)$$

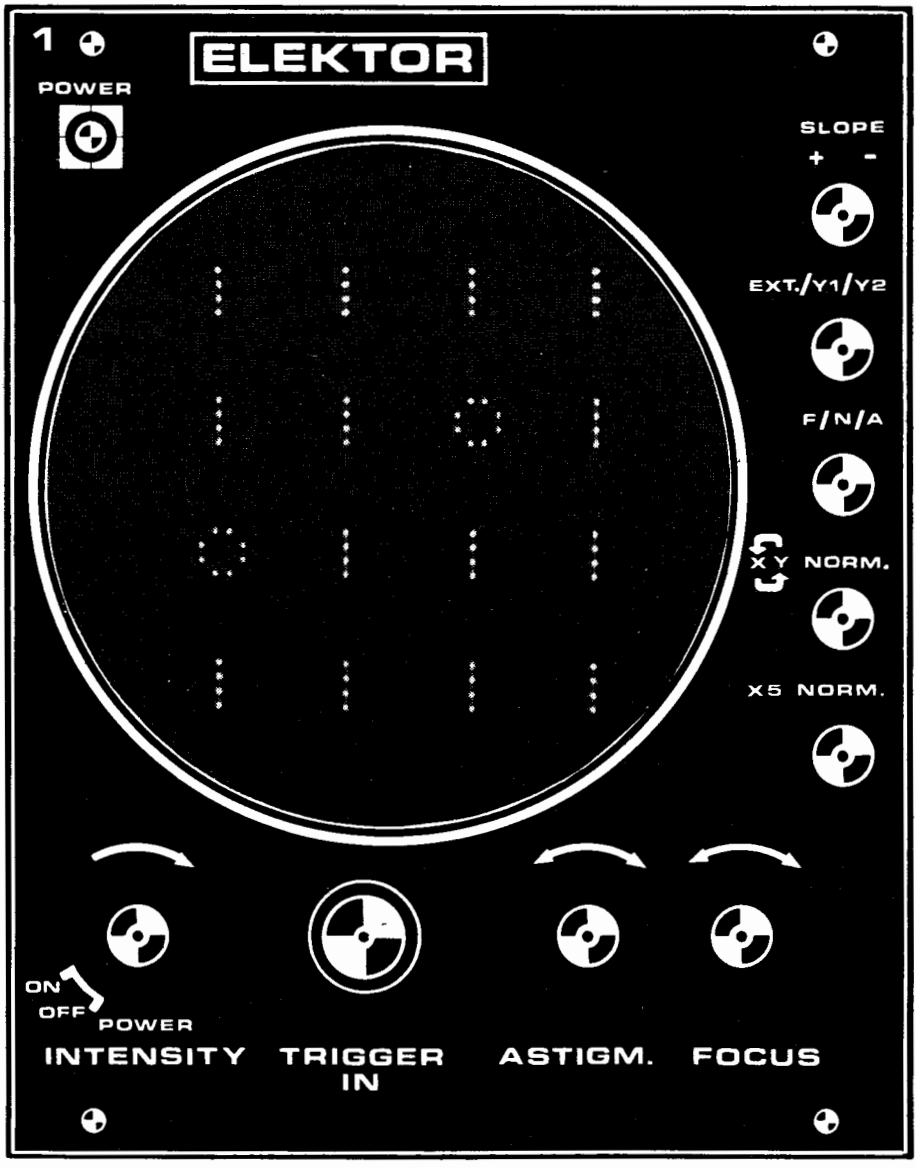
The other two adjacent '1's pinpoint D as redundant, i.e.

$$ABC\overline{D} + ABCD = ABC \quad \ldots\ldots (5)$$

Comparison between (3), (4) and (5) indicates that the map gives a true picture of the output 'g'.

Another use of the Digisplay is as a programmed pulse generator. This is achieved by means of the multiplexor. If, at this point, a variable frequency is required, an external clock pulse generator can be connected. Every bit within one full pass of the multiplexor can be defined as a unique combination of the variables W, X, Y and Z. The complete set of the functions is shown in figure 12, and each function may be correlated with one of the multiplexor inputs E0-E15. For this application the $\overline{S}$ output of the multiplexor has been inverted by T1 so that the true state of the input is available at S. By connecting relevant multiplexor inputs either to ground or V_{CC}, the desired combination of sixteen pulses will be available at S. This may be verified by connecting the Digisplay to an oscilloscope in the usual way.

digisplay on board



The 'Digisplay' circuit described in E13 (May 1976, p.538) has proved to be more popular than we had expected. Whenever the working model was demonstrated at radio shows we received a large number of requests for a design for a printed circuit board.

A suitable board design is shown in figures 1 and 2. Since the unit will normally be used as a more-or-less self-contained measuring instrument, a suitable power supply was included on the board. The circuit of this supply is shown in figure 3. The fuse and transformer are not mounted on the board, of course.

Photo 1. This logic tester displays the states of sixteen binary signals (either '0' or '1') simultaneously, in the convenient form of a 4 x 4 matrix on an oscilloscope screen.

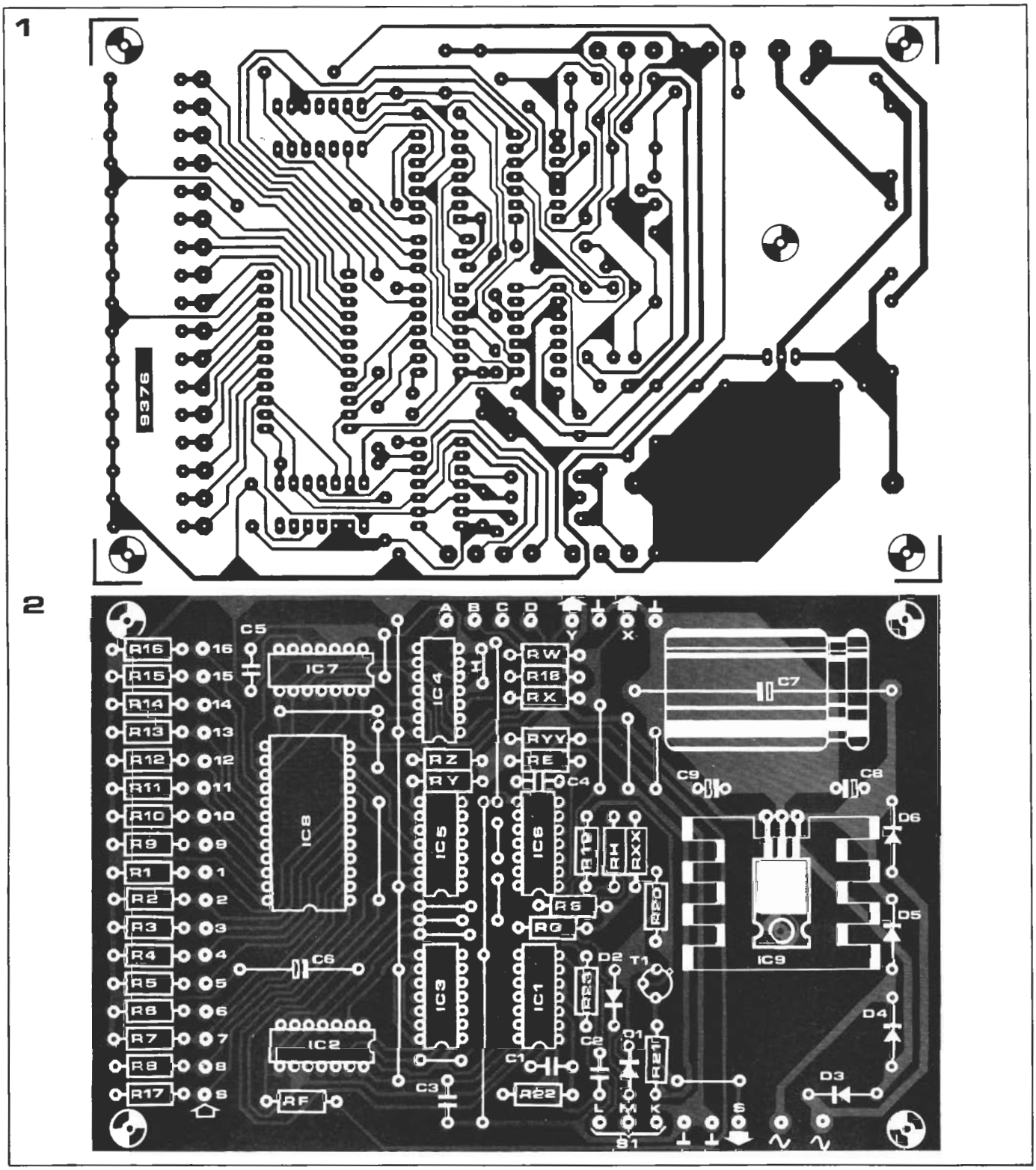


Figure 1. Printed circuit board design for the Digisplay (EPS 9376).

Figure 2. Component layout. Note that there is sufficient space on the board for an adequate heat-sink for IC9.

Figure 3. The power supply circuit.

Figure 4. Complete circuit of the Digisplay, as originally published in May 1976.

Parts list

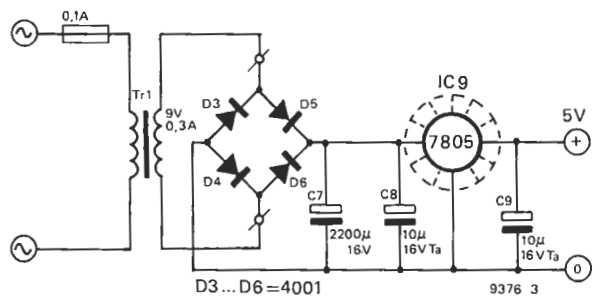
Resistors:
R1 . . . R17,R19,R20 = 10 k
R18,Rx,Rz = 3k3
R21 . . . R23 = 1 k
Ryy,Rxx,Rw,Ry = 6k8
Re,Rg = 68 k
Rf,Rh = 33 k
Rs = 47 k

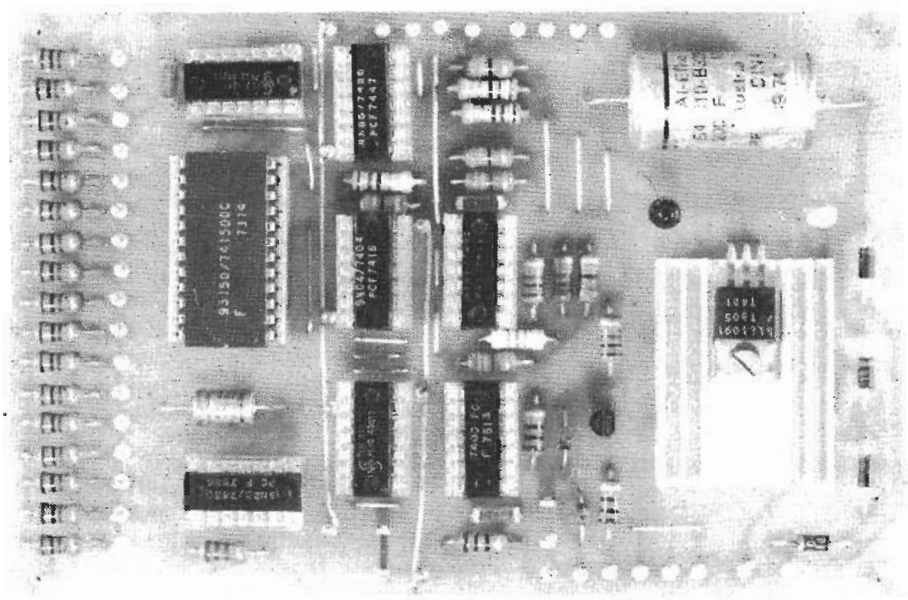
Capacitors:
C1,C2 = 33 n
C3 . . . C5 = 10 n
C6 = 22 µ/6.3 . . . 25 V
C7 = 2200 µ/16 V
C8,C9 = 10 µ/16 V tantalum

Semiconductors:
T1 = BC547,BC107 or equ.
D1,D2 = DUS
D3 . . . D6 = 1N4001
IC1 = 7400
IC2 = 7420
IC3 = 7410
IC4 = 7486
IC5 = 7404
IC6,IC7 = 7493
IC8 = 74150
IC9 = 7805

Sundries:
S1 = SPDT (single pole double throw)
Tr1 = mains transformer, 9 V/300 mA secondary
Fuse = 100 mA, slow blow.

3





Note that the regulator IC (IC9) should be provided with an adequate heat-sink. There is sufficient room on the board for this.

The complete circuit of the Digisplay is shown in figure 4; however, we will not annoy our regular readers by repeating the article as originally published last May . . . . .

For our not-so-regular readers, we can sum it up very briefly as follows. The Digisplay tests the states of sixteen binary signals and displays the results on an oscilloscope screen in the particularly convenient form of a four-by-four matrix of '0' and '1' characters. It may be used to test digital integrated circuits; to display logic states in the form of Karnaugh maps; as a programmed 16-bit pulse generator.

A fixed-frequency clock generator is incorporated in the circuit, and an external clock input is also provided.

For testing digital ICs and for Karnaugh mapping, the 'X' and 'Y' outputs are connected to the Horizontal (X) and Vertical (Y) inputs of an oscilloscope. The test inputs '1' to '16' are connected to the relevant points in the circuit under test and the unit is switched to the internal clock generator. For Karnaugh mapping, the BCD code corresponding to each position of the display matrix is available at the A, B, C and D outputs. The serial in- and outputs ('S') are required when the unit is used as a programmed pulse generator.

4