digits on TV

W. Frenken

This circuit can be used to display a row of up to eight digits on a TV screen. It gives a video output consisting of horizontal and vertical synchronisation pulses and the black/white pattern corresponding to the digits in seven segment format.
It will accept an eight-digit parallel input and, as an example, the circuit of an eight-decade frequency counter is also given.
Leading-zero blanking is included in the design.

There are numerous applications for a circuit that will display digits on a TV screen. The maximum number of digits that will fit on the screen depends, of course, on the size chosen. For most applications the size chosen here should be suitable: eight digits over the full width of the screen. In principle, the system can be used to give up to five rows, or a total of 40 digits. However, in this circuit the display is limited to one row.

The digits are 'written' on the screen as a pattern of white dashes, stacked vertically on top of each other. The European TV system has 625 lines for each complete picture, and each picture is built up in 40 ms (two 'fields' of 20 ms each). This means that each line takes 64 μs. The line sync pulse takes 12 μs, leaving 52 μs for the visible portion of each line.

It so happens, as we will see later on, that it simplifies matters to use multiples of 1 μs for all the time-slots. Since we want eight digits in a row, with gaps between them, an obvious 'binary' choice is 2 μs per digit with 2 μs gaps. This will give digits that are nearly one inch wide (2 cm) on a large screen.

The seven-segment digits are built up as shown in figure 1. Letters A to G indicate the seven segments, and each digit consists of two or more of these segments. Since the digit width is 2 μs, the shortest time-slots (t1 and t3) are only ½ μs — but this will not present any problems.

The next point is to choose the correct height. The height of one vertical unit (L1, for instance) should be the same as one horizontal unit. The horizontal unit is ½ μs, which is $\frac{1}{104}$ times the length of one line — the duration of one (visible) line is 52 μs. Since the width-to-height relationship of a TV picture is 4:3, the 'vertical unit' must be

$$\frac{4}{3} \times \frac{(312.5 - 20)}{104} = 3.75 \text{ lines}$$

in each field (the duration of each visible field is $\frac{625}{2}$ — 20 lines). This is not easy to obtain, so 4 lines are used instead — a nice round binary number ... The digit height is then 24 lines per field (48 lines per picture), or just over one inch (3 cm).

Timebase generator

The first requirement for a stable picture is that it is built up at a speed that is 'TV compatible'. Furthermore, the digit generator and the television set must run in synchronism. Both requirements can be met by using a sync generator that runs at normal TV speed, and using this to generate the digit pattern.

The basic timebase is shown in figure 2. The clock generator consists of N1 and N2 with the 1 MHz crystal. This drives a divide-by-eight counter (FF1 to FF3) which, in turn, drives a second divide-by-eight counter (FF4 to FF6). The total division ratio is therefore 64, so that the output period time is 64 μs — the line frequency. The simplicity of this division is one of the main reasons for choosing 1 μs as the basic time unit!

At this stage, it is interesting to mention the so-called octal way of counting. Since binary systems work in powers of 2, basic counts are 2, 4, 8, 16, 32, etc, corresponding to the basic counts in 'normal' decimal counting of 9, 99, 999, etc. 'Octal' counting groups three basic binary counts together, and uses decimal numbers. The basic counts are then 7, 77, 777, etc. To give a few examples: binary 1 is octal 1; binary 11 is octal 3; binary 101.110 is octal 56. To distinguish octal numbers from decimal numbers, an extra 8 is added, thus: 56_8 (octal) = 46 (decimal). The two divide-by-eight (= octal) counters therefore count from 00_8 to 77_8.

The total picture consists of 625 lines and two fields. This means that the field synchronisation is required every 312.5 lines, or, to put it differently, every 625 half-lines. The output of FF5 corresponds to half-lines. This output is connected to four divide-by-five counters in cascade (IC4 . . . IC7). The output of FF7 can now be used for field sync.

Sync generator

The actual sync signals required are rather more complicated. The outputs of the counters FF1 to FF6 and IC4 to IC7 are connected to the video generator shown in figure 3.

The line sync pulse $\overline{\text{LSY}}$ (output from N27) is derived from Q3 to Q6, so it lasts from 111100 to 111111, or 74_8 . . . 77_8, which is 4 μs (see table 1 and figure 4). The 'blacker-than-black' signal, or line blanking ($\overline{\text{LBL}}$, output of N12), lasts from 72_8 (through 77_8 and 00_8) to 05_8. This corresponds to 58 μs in one line through to 5 μs in the next line. The total duration is therefore 12 μs, with the 4 μs sync pulse 'off-centre' in the middle.

Something similar is required for the field sync. The field sync pulse itself, $\overline{\text{FSY}}$, lasts from half-lines 590 . . . 594; the equalizing pulses $\overline{\text{FSE}}$ from 585 . . . 589 and from 594 . . . 599; and the field blanking $\overline{\text{FBL}}$ (blacker-than-black) from 585 . . . 624. To be slightly more specific, the sequence is as follows. During the first five half-lines, 585 . . . 589, the output of N19 ($\overline{\text{FSG}}$) is low. This allows the equalizing pulses $\overline{\text{EQP}}$ from N20 to pass gate N22, and from there through N26 to the output. During the next five half-lines, 590 . . . 594, the output from N18 (field sync, FSY) is high. This field sync pulse is mixed with the equalizing pulses $\overline{\text{EQP}}$ in gate N25, giving the output $\overline{\text{FSY}}$. Finally, during the third five half-lines, 595 . . . 599, the second set of equalizing pulses are passed through gates

Figure 1. Seven-segment display format on the TV screen. The digits are 2 μs wide (t1 + t2 + t3) and 6 'vertical units' high. Each vertical unit is a total of 8 lines on the screen.

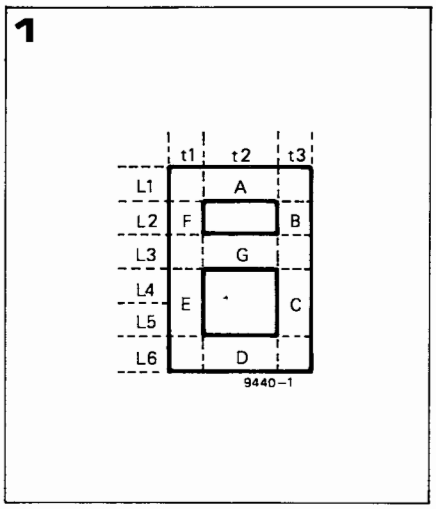
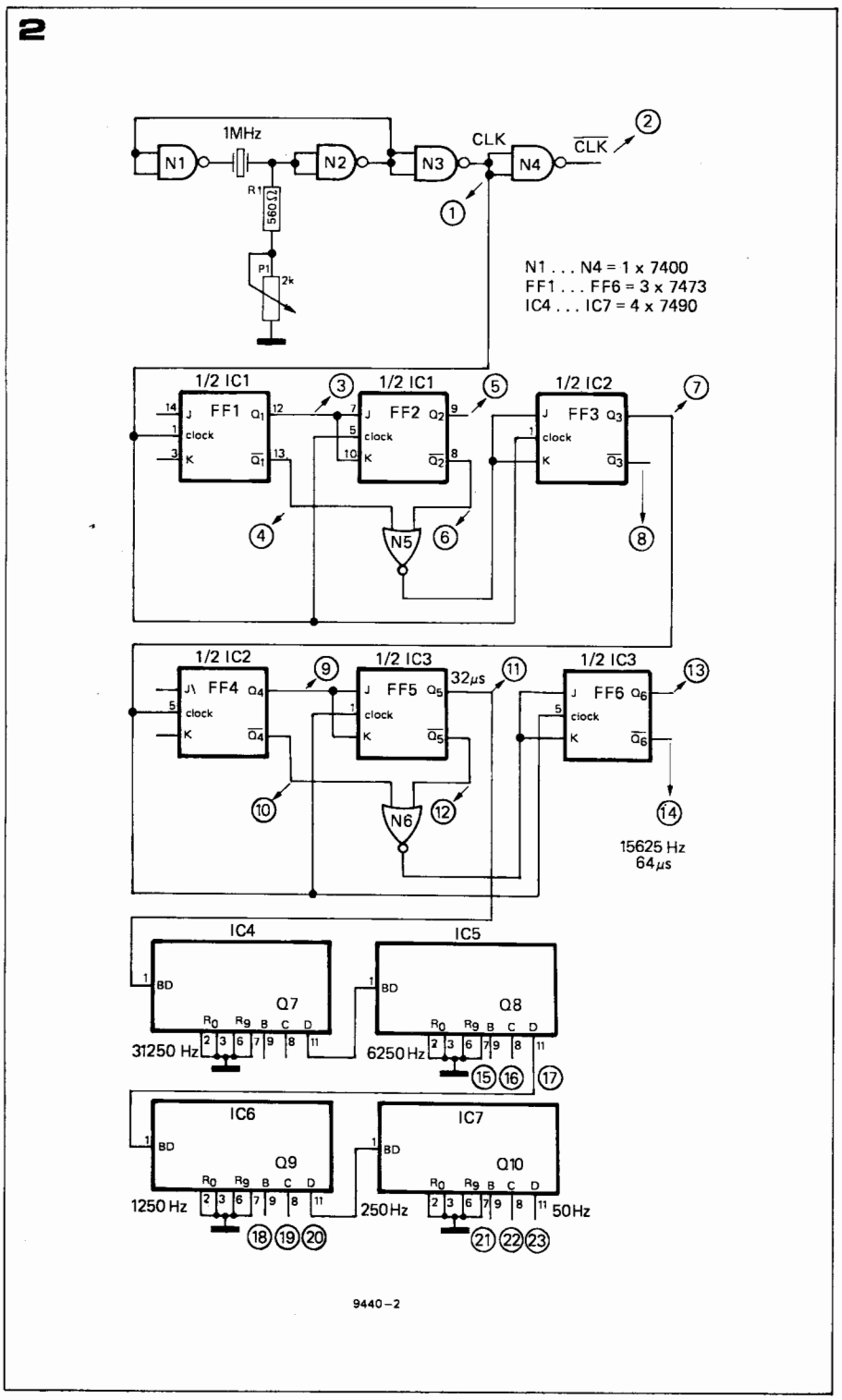


Figure 2. The timebase generator, comprising a 1 MHz clock generator, a 1 . . . 64 μs divider and a half-line counter.



N22 and N26. During this whole period (585 . . . 599) the output of N24 is low, providing the first part of the frame blanking $\overline{FBL}$; during the following period (600 . . . 624) gate N13 takes over this frame blanking function.

The normal line sync (LSN) is blocked by gate N23 during the field sync sequence.

The function $\overline{END}$ corresponds to the end of each field (half-lines 575 . . . 599), and it can be useful for synchronising external circuits.

The video input is connected to N28. Resistors R3, R4 and R5 are a simple digital-to-analog converter which produces the correct video output levels for the various signals:

'white' = 100%;
'black' = 35% ($\overline{BLA}$);
'blanking' = 30% ($\overline{LBL}$) and $\overline{FBL}$);
'sync' = 0% ($\overline{SYP}$).

Character generator

The circuits described so far produce basic timebase outputs (from ½ μs to field sync) and all necessary sync pulses. There is also an input for digital video signals. The next step is to produce digits in the desired video format.

The height of the characters is determined by the 'vertical unit': 4 lines. In figure 5, the input is at line frequency. Two flip-flops (FF7 and FF8) are used as a divide-by-four counter; output Q13 is the 'vertical unit'. The following three flip-flops (FF9 to FF11) form a vertical unit counter, and the six vertical units required (L1 . . . L6) can be derived from their outputs.

Gates N35 and N36 form a flip-flop. When this is set, output $\overline{Q11}$ holds FF7 . . . FF11 in the reset condition, so that no digits can be generated. Gate N34 produces the signal $\overline{DST}$ (display start), resetting flip-flop N35/N36. This enables the other flip-flops, but it takes a further four lines for Q13 to change state for the first time, starting the character generation. If the inputs to N34 are connected as shown, the digits will be written at the bottom of the screen.

The vertical unit counter counts from L0 to L6. When it switches to L7, flip-flop N35/N36 is set by gate N33. This, in turn, resets and blocks FF7 . . . FF11. End of digit.

The width of the digits is determined by the horizontal units: ½ μs, 1 μs and ½ μs for t1, t2 and t3 respectively. These units are derived from the timebase generator by part of the circuit shown in figure 6.

N56 derives t1 from Q1, $\overline{Q2}$ and CLK, so t1 corresponds to the first ½ μs out of every 4 μs. N57 produces t2, the following 1 μs; finally, N58 produces t3, the final ½ μs. After a further 2 μs (the gap between digits), the cycle is repeated. This total cycle, corresponding to digit-plus-gap, is repeated 16 times for each line, or 13 times for the visible portion of each line.

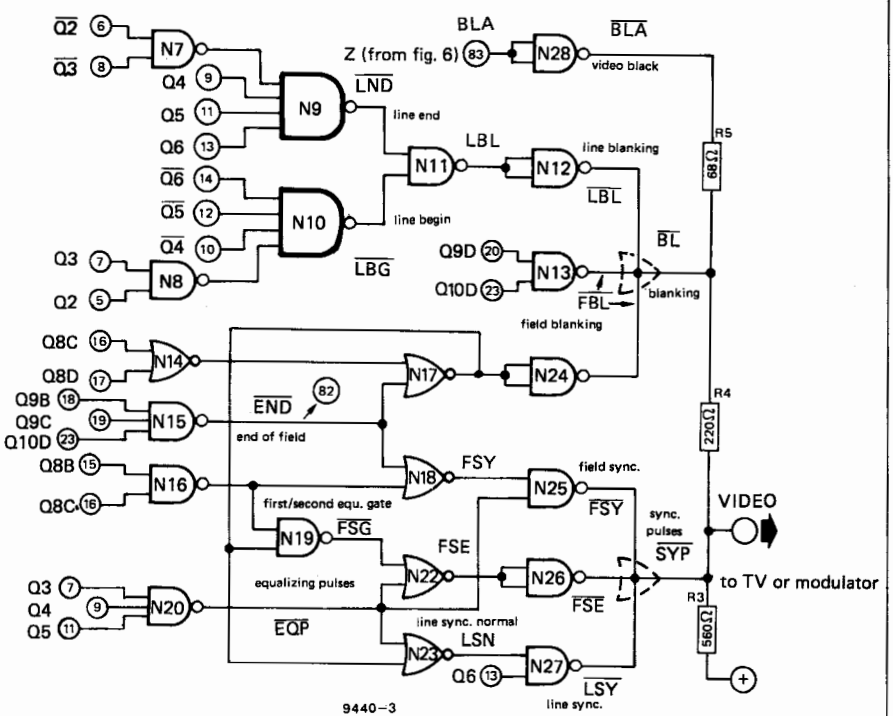
These horizontal units t1 . . . t3 are now combined with the vertical units L1 . . . L6 from figure 5 and the seven-

segment signals a . . . g to produce the total video output BLA. As an example, segment A is formed as follows.
Horizontal units t1 . . . t3 are combined in N62, giving a signal over the full width of the digit. Vertical unit L1 and seven-segment signal 'a' are combined in N59. The outputs from N62 and N59 are then combined in N63 to produce the video output A, corresponding to segment A in the display.
Gates N52 and N54, wired into the 'segment A' circuit, give a certain 'finesse' to the unit . . . The seven-segment decoder used gives a basic '6', that is, the top bar (segment A) is missing. However, a clearer display is obtained when this segment is added. If the A segment is driven whenever the D and G segments are 'on', the only difference is that this top bar is added to the '6'.
Gates N53 and N55 perform a similar function to improve the display of the '9'.
The digital video output BLA can be connected to the video input in figure 3 (N28) to give white digits on a black background. If black digits on a white background are required, an inverter will have to be connected in series. If both options are required, an exclusive-or gate can be connected in series, with a switch between the other input of this EXOR and ground.
In the discussion so far we have simply assumed that seven-segment signals a . . . g are available. Happily, they are not difficult to obtain. The BCD to seven-segment decoder is shown in figure 7.
The decoder is a 7447 (IC8). Under normal conditions it converts a binary code to a seven segment output. Its output is only enabled during the time that digits are being displayed (DPT = display time) and the time that the digits are not changing ($\overline{PLG}$ = parallel load gate). These functions will be explained further on.
The 'lamp-test' input is connected to S1 — after all, if an input is available, why not use it? This simply produces an '8' at the output, i.e. all seven segments, so that it gives a quick check of the logic circuits. The display should be 88888888 when this switch is closed.

Display generator

The eight-digit display is derived from a frequency counter. The circuit (figure 8) is not very sophisticated, since the main purpose is simply to derive eight digits to drive the display. It is simply one example of how to drive the rest of the circuit.
The frequency counter consists of eight decade counters (IC14 . . . IC21). The input count pulse ($\overline{CTP}$) comes from the input gate N67. The input frequency is applied to the input (INP), as one would expect . . . The 50 Hz output from the main timebase (Q10D in figure 2) is divided by 50 in IC12 and IC13 to produce a 1 second output which drives the input gate.
During the time that Q18A is low

3



N7,N8,N11,N16,N19 = 1½ x 7400
N12,N13,N24 . . . . . ,N28 = 1¾ x 7401
N9,N10 = 1 x 7420
N14,N17,N18,N22,N23 = 1¼ x 7402
N15,N20 = 2/3 x 7410

4

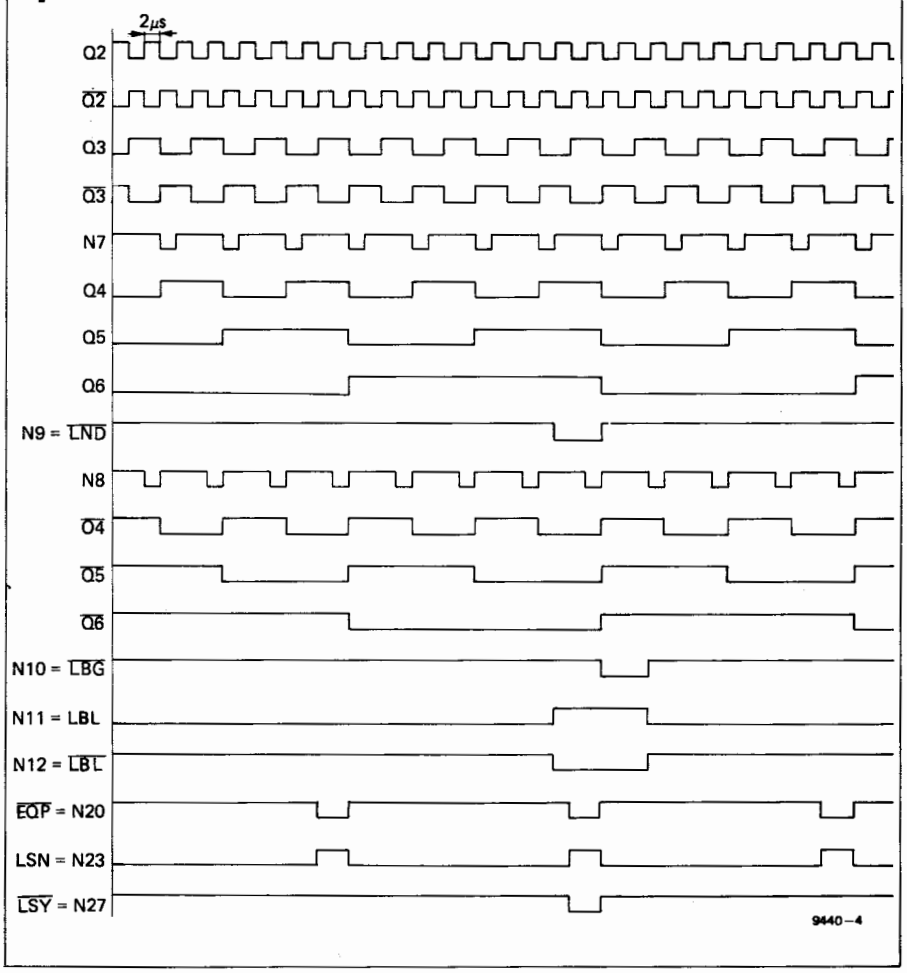


Figure 3, The sync generator. This produces the line and field sync pulses, line and field blanking, and includes a video input.

Figure 4. Pulse diagram showing the derivation of the line sync (LSY) and line blanking (LBL) pulses.

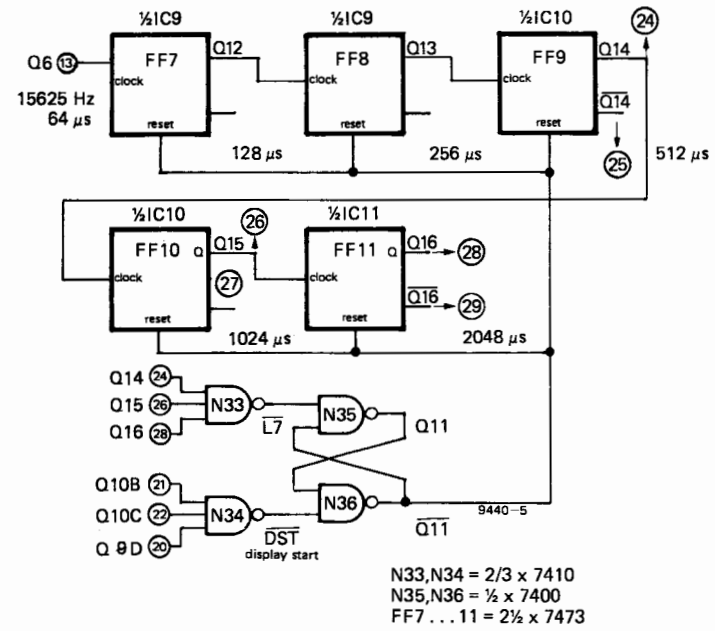
Figure 5. The vertical unit counter. This also produces the 'display start' signal corresponding to the top of the display.

Figure 6. The horizontal unit counter and seven-segment to video encoder.

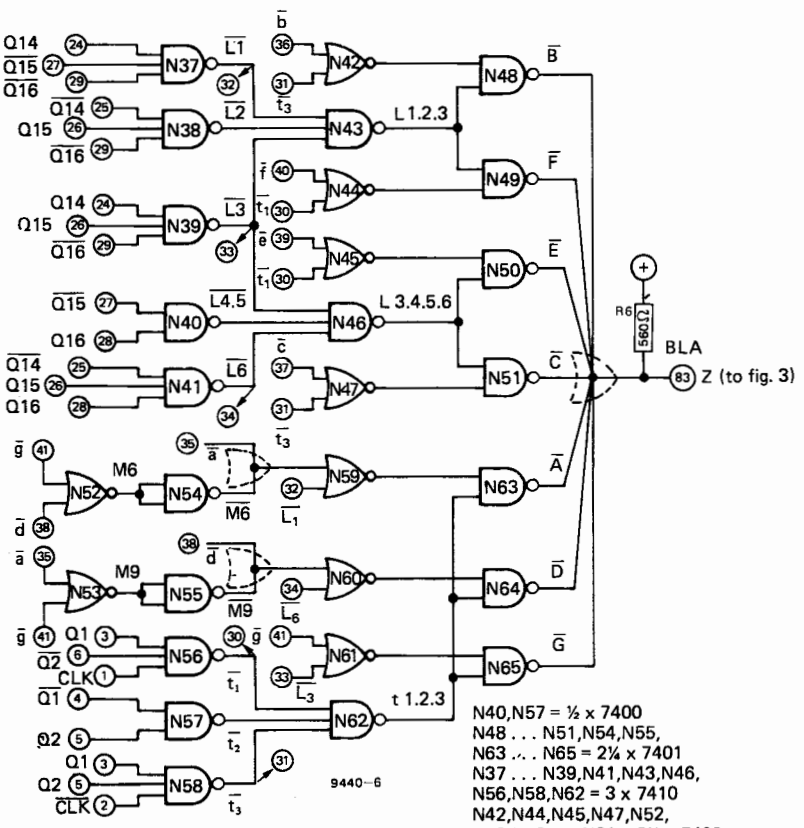
Table I. Part of the counting sequence of the two divide-by-eight counters (FF1 . . . FF6).

n_{10} decimal count	'8's Q6	Q5	Q4	'1's Q3	Q2	Q1	n_8 octal count
0	0	0	0	0	0	0	00
1	0	0	0	0	0	1	01
2	0	0	0	0	1	0	02
3	0	0	0	0	1	1	03
4	0	0	0	1	0	0	04
5	0	0	0	1	0	1	05
6	0	0	0	1	1	0	06
7	0	0	0	1	1	1	07
8	0	0	1	0	0	0	10
9	0	0	1	0	0	1	11
10	0	0	1	0	1	0	12
11	0	0	1	0	1	1	13
⋮							
26	0	1	1	0	1	0	32
27	0	1	1	0	1	1	33
28	0	1	1	1	0	0	34
29	0	1	1	1	0	1	35
30	0	1	1	1	1	0	36
31	0	1	1	1	1	1	37
32	1	0	0	0	0	0	40
33	1	0	0	0	0	1	41
⋮							
59	1	1	1	0	1	1	73
60	1	1	1	1	0	0	74
61	1	1	1	1	0	1	75
62	1	1	1	1	1	0	76
63	1	1	1	1	1	1	77
64	0	0	0	0	0	0	00

5



6



(1 second), the input gate is blocked and a short output pulse is given at $\overline{PLG}$ and PLG. These latter outputs block the seven-segment decoder (figure 7) and load the shift register (figure 9), as will be explained further on. Towards the end of the $\overline{PLG}$ pulse, the $\overline{END}$ pulse also appears (from figure 3) at the input of N66, producing a reset pulse (RES) for the eight decade counters IC 14 . . . IC21.

A completely separate unit within the display generator is the section shown at the bottom of figure 8. This could also be labelled the 'display position generator'.

The character generator (figure 5) determines the vertical position of the digits: the half-line counter is enabled by the $\overline{DST}$ signal. However, the horizontal position is still undetermined.

The unit is designed to produce a row of 8 digits. Each digit takes 4 µs, so the total display width (including the gaps between the digits!) is 8 x 4 = 32 µs. An exclusive-or gate (N72) is used to derive the 'display time' signal (DPT) from the

main timebase. This signal lasts 32 μs and it is positioned in the centre of each line.

Gate N71 derives a signal from the vertical time unit counter in the character generator (figure 5). This signal corresponds to the total vertical height of the digits. The combination of the horizontal display position DPT with the vertical display position signal from N71 results in a signal which corresponds to the exact position of the display: $\overline{DSG}$. During this display period, pulse t3 from figure 6 is passed by gate N74. The result is a sequence of pulses which coincide with the end of each digit. This signal is used as 'clock' (SRC) for the shift register. This is the next part of the circuit to be discussed.

Display memory

During the 'count' period, the result of the preceding count must be stored somewhere. Furthermore, since there is only one character generator, the eight digits in the total display must be presented to this character generator one at a time and at the correct moments. Both of these requirements can be fulfilled by using a shift register as a memory.

This shift register (figure 9) uses eight ICs. It actually consists of four separate shift registers running in parallel, one for each binary 'bit'. The top shift register (IC22 and IC23) is used for storing the eight 'least significant bits' (Bit A) of the eight digits in the displays, so its output is connected to the 'A' input of the decoder (figure 7). The remaining three registers are used for bits 'B', 'C' and 'D' respectively.

At the end of each count period, the outputs of the eight decade counters in figure 8 correspond to the eight digits required for the display. The Most Significant Digit (i.e. the left-hand one) corresponds to the output of IC21, and so on down the row to the Least Significant Digit which corresponds to the output of IC14. The display on the TV screen is written from left to right, so the first digit will be the Most Significant Digit. The correct order to store the digits in the shift register is therefore: output from IC21 at the extreme right-hand end (the output), and all other outputs in order down the register from right to left.

The actual storing of the information into the shift register is controlled by the 'parallel load gate' signal PLG (from figure 8). This signal appears just before the start of each new count.

If no further action is taken, the Mc Significant Digit will remain at the ou put of the shift registers. This digit w then appear eight times on the scre during the display period. Since this not the intention, the shift registe must now be set into motion. This what the 'shift register clock' signal needed for.

As discussed in the previous section, th SRC signal consists of a sequence c pulses which coincide with the end c each digit. They are only present durin the display period. The shift register move their information up one positio on the trailing edge of each clock pulse This corresponds to the end of eacl digit.

The results are now as follows. At the beginning of each line of the display period, the Most Significant Digit is present (in binary code) at the output of the shift registers. After the corresponding pattern has been displayed, the shift registers move up one position so that the second digit is present at the output. After the pattern for this digit has been displayed, the shift registers advance again, and so on. As the information for the digits is shifted out at the right-hand end it is fed back in at the left, so that it is not lost.

Display blanking

The only part of the circuit not yet discussed is the lower part of figure 7: the display blanking.

The output of gate N29 is '1' outside the display time (DPT) and during the time that new information is being loaded into the shift registers ($\overline{PLG}$). This 'DBL' signal is inverted by N30 and drives one of the display blanking inputs of the decoder. The result is that all outputs go to '1' and none of the seven segments can be displayed.

Furthermore, flip-flop N31/N32 is reset and the output of N31 drives the other display blanking input. Since the DST signal corresponds to the true display time, including the horizontal boundaries, this flip-flop is reset during each line after the pattern for the last digit has been displayed. The display blanking now remains operative until the flip-flop is set.

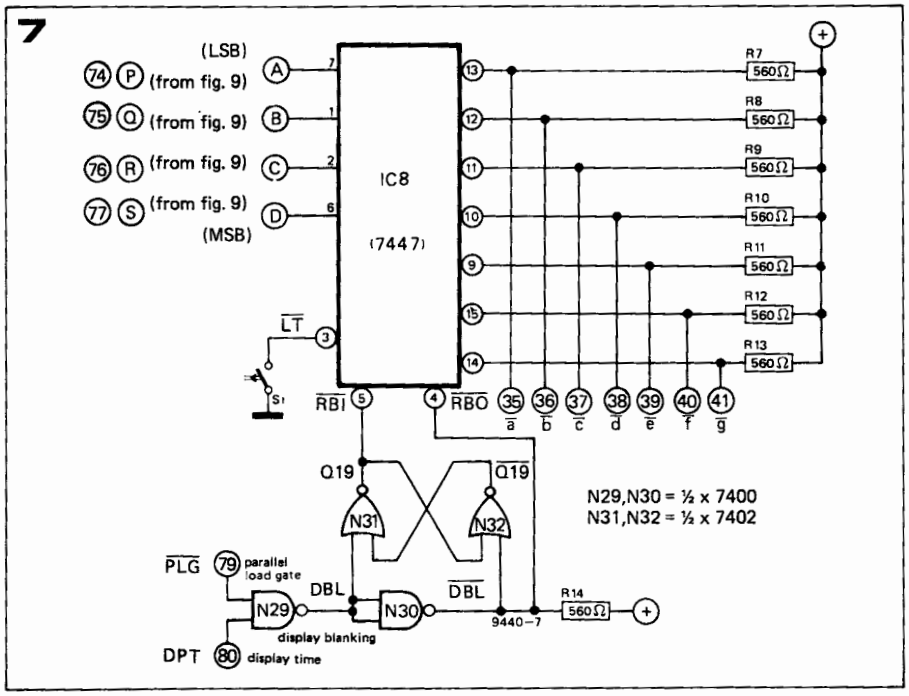
As soon as the display time is reached during the next line, this overriding reset is removed. However, this does not mean that the flip-flop is automatically set. For this to happen, the RB0 output of the decoder must become logic '1' (note that 'RB0' can be used both as input and as output!). This output remains '0' as long as a zero is present at the decoder input, however. The result is that any 'leading zeroes' are suppressed: the display blanking remains operative.

As soon as a digit is presented to the decoder that is not zero, RB0 becomes '1'. This sets the flip-flop, and all further digits (including any zeroes) are displayed until the DPT signal again resets the flip-flop.

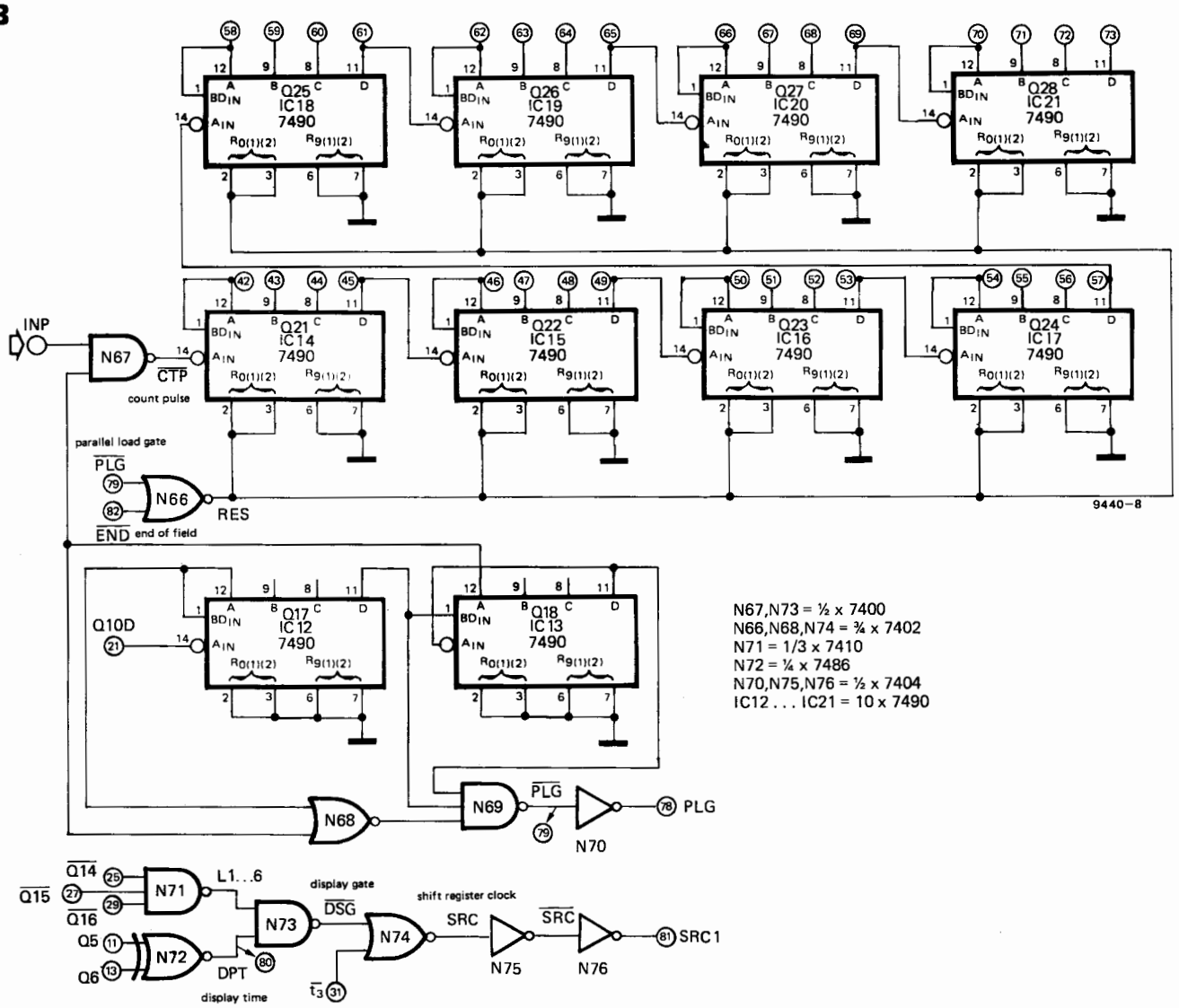
Figure 7. The binary to seven-segment decoder. The circuit includes display blanking and leading zero blanking.

Figure 8. The frequency counter and shift register clock pulse generator.

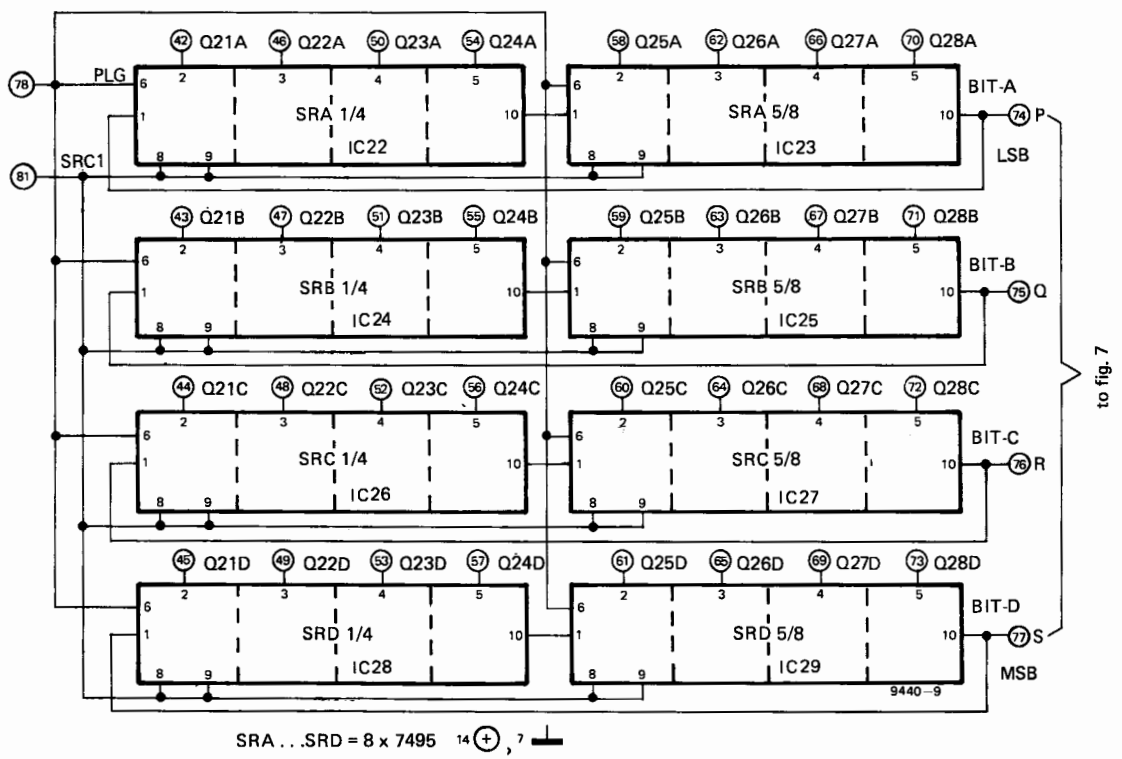
Figure 9. The shift register.



8



9



SRA . . .SRD = 8 x 7495 14 (+), 7 ⏚