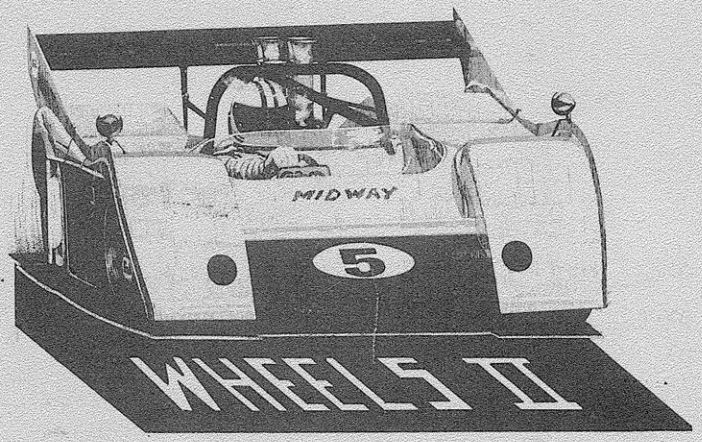


COMPUTER SERVICE MANUAL

Distributed by: William Arkush

2876 Culver Ave. Dayton, Ohio 45429

(513) 296-0330

VIDEO GAME DATA LIBRARY

VOLUME II

TABLE OF CONTENTS

LOGIC SYMBOLOGY & NOTATION

The function and operation of logic circuits is described using standard logic symbology and notation. All logic functions are also described using the positive logic convention. Positive logic is a system of notation where the more positive of two levels is called the HI or (1) state and the more negative is known as the LO or (0) state. All logic circuits are of the TTL family or are directly TTL compatible. The specific voltages are +2.4 volts to +5 volts for logic HI and 0 to +0.8 volts for logic LO. Signal names overscored (e.g. $\overline{\text{V RST}}$ and pronounced "vertical reset not") go Lo to initiate events (active LO) and those not overscored go HI when active. Overscored signals are always at the logic level opposite their non-overscored counterparts.

Logic pacakges are identified in the text and in the logic schematics by their actual grid locations on the printed circuit board (PCB). For example, chip A3 is located at the intersection of the A column and the third row. Pins are identified by a third number. For instance, A3-1 refers to the first pin to the left of the reference notch of the IC.

Discrete components are not as easy to locate for they are not positioned in a way as to be readily identifiable by PCB grid coordinates. Some discretes are easy to locate because of prominent features; others will require laborious following of the appropriate PCB traces. The more important discrete components are identified in the illustration of the printed circuit board computer (page 4).

THE TEST POSITION

When mounted in the cabinet, the monitor of this game is rotated 90° to take advantage of the width of the CRT. Since this is not the normal position of a monitor, we will assume the monitor is rotated back 90° to its normal or "test" position for the discussion and test points in the following text. In other words, the cars are pointed to the left of the CRT, not to the top.

PCB ADJUSTMENTS

A number of circuits are adjusted by devices on the printed circuit board. Most of these adjustments are accomplished by turning trim pots, however the extended play score is adjusted by sub-miniature slide switches and two coin circuit adjustments can be made by moving the positions of jumper wires. The positions of all PCB adjustments are indicated on page 4. The terms "clockwise" and "counter-clockwise" refer to those directions when facing the slotted thumbwheel of the trim pot.

1ST PLAYER VOLUME: To increase Channel 1 volume, turn this pot clockwise.
2ND PLAYER VOLUME: To increase Channel 2 volume, turn this pot clockwise.
C CAR POSITION: Adjust this pot so the controlled car(s) are centered in the desired area of the CRT.
GAME TIME CONTROL: Since this adjustment affects the course speed, the game length must be adjusted prior to course speed adjustment. Clockwise rotation increases game length and the recommended time is 90 seconds.
COURSE SPEED CONTROL: Clockwise rotation increases the course speed and, generally speaking, the speed should be adjusted so a good player with only two crashes can barely reach the extended play score.
ANTI-STATIC ADJUSTMENT: Turn this pot counter-clockwise all the way and then start a new game. Rotate clockwise until the game shuts off, then back off a bit. Test adjustment according to instructions in the Game Control analysis.
EXTENDED PLAY SCORE: This adjustment is made by changing the positions of the sub-miniature switches located at J10. Initial setting should provide extended play at a score of 400.
COINS PER PLAY: Since this adjustment is pre-set by the factory at one coin per single player game and two coins per two player game, it will normally not need adjustment. However, if adjustment is necessary to accomodate a foreign currency structure, see the Game Control analysis.
PLAYS PER COIN: Changing the position of the jumper wire to B11-4 will change the game from one to two plays per coin in the single player mode and from one to two plays per two coins in the two player mode.

TEST EQUIPMENT

The test data found in the text are referenced to three test instruments which you must have in order to fully troubleshoot this computer: the logic probe, the video probe and the oscilloscope.

LOGIC PROBE: The logic probe is a digital test instrument designed for checking integrated circuit outputs. The probe indicates if the examined signal is a logic HI, LO or changing states (PULSING). All logic probe test point data are referenced to the Kurz-Kasch Model LP-520, however a number of other manufacturers such as Tektronix, Hewlett-Packard, etc. offer comparable logic probes.

THE VIDEO PROBE: the video probe is a simple, but extremely useful, test device. It consists of two IC test clips (or one clip and a test prod), a length of rubber-coated wire and a 4.7K, 1/4 Watt carbon resistor. To use the video probe, connect the clip to the negative side (-) of the video coupling capacitor (page 4) and touch the probe tip to the node under examination.

The video probe simply couples the desired signal directly to the TV monitor where it is displayed on the CRT along with any other information already entering the monitor through the video input line. The video probe is useful for viewing signals used in the development of other signals where the developmental signals are not otherwise visible. Generally, a number of signals are used in the development of a final video display and the video probe provides a graphic display of the development process. Unfortunately, the video probe can only be used for viewing video signals. Extremely fast signals such as CLK will not be displayed and nor will it display either analog or very slow digital signals.

If desired, a separate video monitor may be used to display video probe signals, however frame sync will need to be connected to this monitor as well. Also, there is no real limit to the number of signals which can be displayed at any given time, however each new signal requires clipping on another probe or jumper wire. If a number of probes are connected in this way, a composite of developmental signals can be displayed to check the processing of complex video displays.

OSCILLOSCOPE: Although most computer malfunctions can be successfully tracked down using only a logic probe and a video probe, the use of an oscilloscope is highly recommended as there are a great many timing relationships in this game which cannot otherwise be examined. Additionally, the only way analog waveforms can be accurately examined is with an oscilloscope.

TEST DATA

To check test points, first consult the block diagram (Figure 7) to locate the suspected and related circuits. The test points are arranged in a logical order beginning with developmental signals and ending with section outputs. First, check the section outputs and, if correct, you are in the wrong area entirely. If not OK, proceed to check developmental signals. Follow these signals backwards until you find correct inputs or signals. The malfunction must then be in that area between the correct inputs and incorrect outputs. If all the signals in a section appear incorrect, examine related section outputs using the block diagram as a general guide.

Data for a signal or test point is given only for those instruments which are capable of testing that signal. Since the logic probe is incapable of checking analog signals, logic probe test data will appear only for digital signals. And, although some indication of signal periodicity is obtainable from the logic probe, oscilloscope data will take precedence where precise timing relationships or specific signal periodicity are in question.

Likewise, video probe data will be given only for those digital signals which are capable of being displayed by the monitor and they are presented in two forms: (1) Verbal explanations are used where the appearance of the signal is readily described in words and not too complicated in form. Please bear in mind that the signal you see displayed on your CRT may not exactly correspond to the description you read in the text due to the fact that these descriptions are, of necessity, highly condensed. Minor discrepancies should not be a cause for concern since, generally speaking, malfunctioning signals will show marked distortion. (2) Actual illustrations may be used for complicated or especially important signals where verbalization would be inadequate. This type of information is indicated by a reference to a specific figure number.

Oscilloscope data for digital signals includes only the duration and frequency of the pulse in question. Analog signals are generally described by a characterization of the waveform. Where precise timing relationships or unusual waveforms are difficult to describe, illustrations or oscilloscope photographs will be included.

The test instrument names are abbreviated in the test point data descriptions, where LP = Logic Probe, VP = Video Probe and SC = Oscillocscope.

RASTER SCAN NOMENCLATURE

A number of terms associated with the control of the electron beam are explained here in preparation for a later section dealing with the generation of the sync signals. Since these signals are timed with the movements of the electron beam, the resulting relationships are quite important.

The TV monitor controls the vertical and horizontal deflection of the electron beam so it creates an endlessly repeated pattern of lines known as the raster. Figure 1 is a simplified illustration of the non-interlaced raster scan used in this game. The electron beam theoretically begins the frame in the upper left corner of the CRT and is deflected so it sweeps one horizontal line ending at the right. The beam is then repositioned back to the left side by a process known as horizontal retrace. During retrace, the video information is blanked out so no undesired illumination can occur. Retrace blanking is an internal function of the monitor and is not related in any way to any "blanking" signals produced by the computer, except that both occur more or less simultaneously. Computer signals V BLNK and H BLNK are used in the development of sync as well as in other circuits.

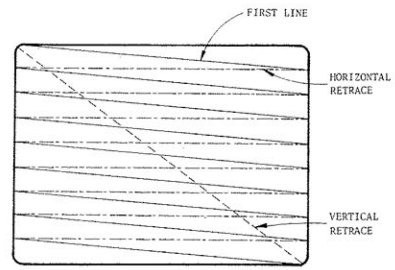


Figure 1
Raster Scan Nomenclature

After the electron beam has been reset to the left again, it sweeps the second horizontal line. Since the beam is deflected vertically as well as horizontally, the second line appears beneath the first. The beam continues to scan in this fashion until the end of the last line which is located in the lower right corner of the CRT. At this point, the beam is reset back to the left and to the top to the place of its beginning. This reset process is known as vertical reset and the video information is blanked out during this time.

Each time the electron beam has scanned the entire CRT, it is said to have completed one full field which contains 272 horizontal lines. In the non-interlaced raster scheme, successive fields are laid approximately on top of one another. Two complete fields constitue one full frame and the monitor frame rate is 30/sec.

DEFINING CRT LOCATIONS

Confusion has frequently arisen about some of the terms used to describe computer events and video probe displays with respect to CRT locations. For example, 128V is said to be LO in the upper half of the CRT which may appear to be confusing unless you understand that signal 128V is LO for the first 128 horizontal lines (or counts of the vertical sync chain), rises HI for another 128 counts, etc. When viewed with the video probe, 128V causes the upper half of the CRT to darken.

The combination of the vertical and horizontal addresses produces a set of coordinates for every point in the raster. For example, to illuminate a point in the approximate center of the CRT, the electron beam must be intensified as it reaches the intersection of 367V and 284H.

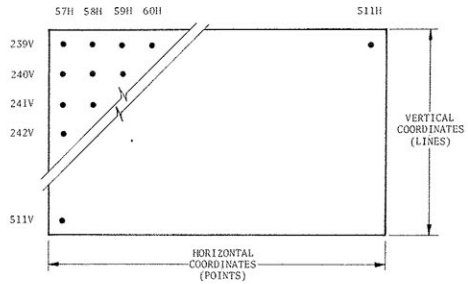


Figure 2
The Coordinate System

The nomenclature associated with this system is illustrated in the figure above. In most cases, the first line is denoted 1V (1st Vertical). However, due to the unique construction of the sync circuitry in this game, the first line is numbered 239V. Since the raster contains 272 lines, the last line occurs at 511V. Every addressable point along each of these lines also has a location or number. In this game, the first point occurs at 57H, the second at 58H and so on until the end of the line which occurs at 511H.

THE WINDOW PARADOX

The term "window" is one which has been coined to describe the area of the CRT affected by the condition or state of the signal. A certain paradox exists when using this term because vertical signals produce horizontally appearing windows and visa versa. For example, the horizontal signal 128H appears as a set of alternating light and dark vertical columns or windows. Since the electron beam is intensified whenever the input is HI, it will cause the first 128 clock pulses (two clock pulses equals 1H) to be dark, the next 128 light and so on (Figure 3). Each line is scanned the same way as long as the signal is present, so the result will appear as a set of vertical windows when all 272 lines are stacked on top of each other. The dark areas are called LO windows and the light ones HI windows.

Figure 3

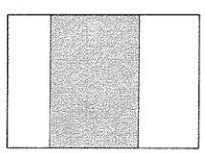


Figure 4

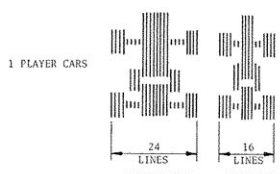


Figure 5

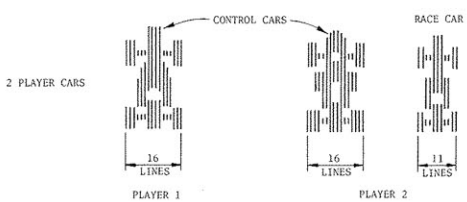


Figure 6

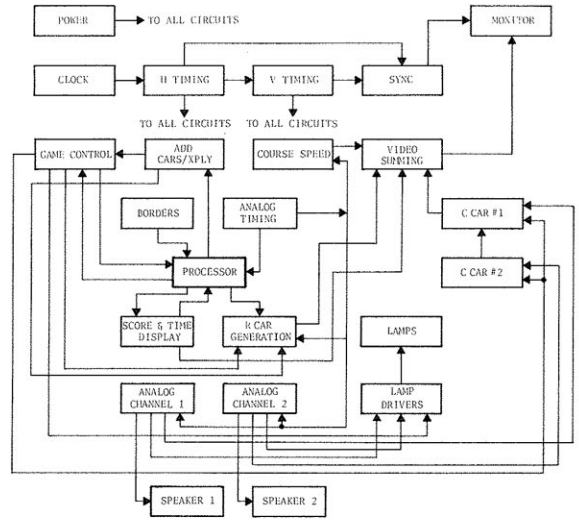


Figure 7
The Block Diagram

THE LAMP DRIVER CIRCUIT

Power for this circuit (+15 volts peak-to-peak) is supplied through pin 19 of J1 and is connected to the emitters of all the driver transistors through 270Ω resistors. Each transistor also has a 1N914 diode wired across the +15 volt line and the base of the transistor which provides one diode drop. This causes the potential at the base to be 0.6 volts lower than at the emitter to insure the transistor is adequately turned on when the output of the associated inverter goes LO. When the appropriate input signal goes HI, the inverter output drops LO, forward biasing the transistor and turning on the associated lamp.

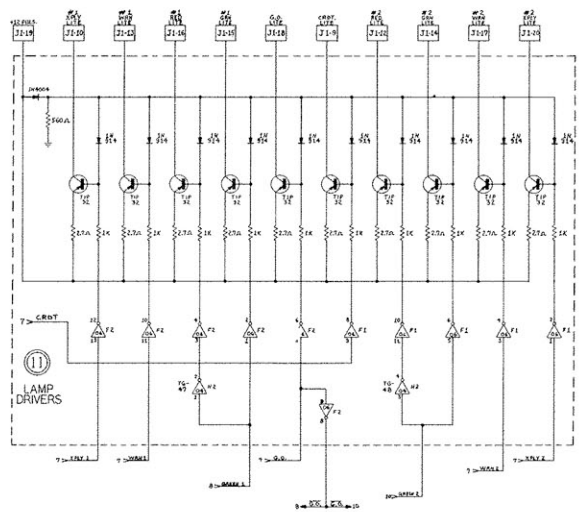


Figure 8
Lamp Driver Schematic

THE ROM READER

Thirteen separate 256-bit ROMs (Read Only Memories) are used in Wheels II to store image information and to generate miscellaneous signals. ROMs are not complex devices, but they can be a headache to troubleshoot. Normally, correct ROM operation is most easily verified by the "substitution technique" where the technician simply plugs in a known-to-be-good ROM and if the game suddenly begins to function correctly, the other ROM must have failed. However, there are thirteen ROMs in this game and the technician must have the full set to have the substitution capability. Since each ROM is custom programmed and fairly expensive, this technique is unrealistic.

In order to test a ROM without using the substitution technique, the information must be read out and examined for correct logic levels. This can be done with a logic probe by tying the address inputs at the desired word and checking all eight bits for correct logic levels, but this is an extremely laborious process. Furthermore, you still need to know what the bit pattern in a good ROM should be.

The following procedure eliminates the cost of the substitution technique and the labor of the logic probe by displaying the entire bit pattern of the ROM on the monitor CRT. However, the procedure does require assembling a small "test station", but this should take only a few minutes. The only parts required are three IC sockets (two 16 pin and one 14 pin), a 2" by 4" piece of vector board, a 9312 8-to-1 multiplexer, a NAND gate and a few feet of #30 AWG wire.

Connect the IC sockets as indicated in Figure 9. Power, ground and all signal inputs are readily found on your Wheels II PCB. Insert the multiplexer and NAND gate and the ROM is question. The result will be a bit pattern displayed in the area of the CRT determined by the strobe input of the multiplexer (Figure 4). Simply compare the pattern you see on your monitor with the one in this manual under the appropriate ROM number.

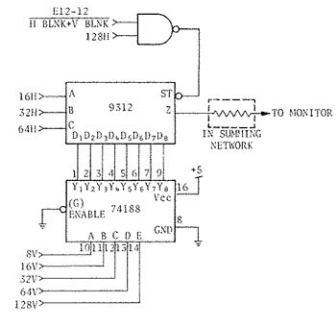
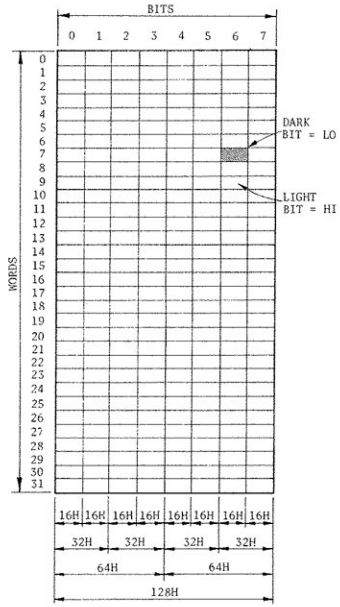


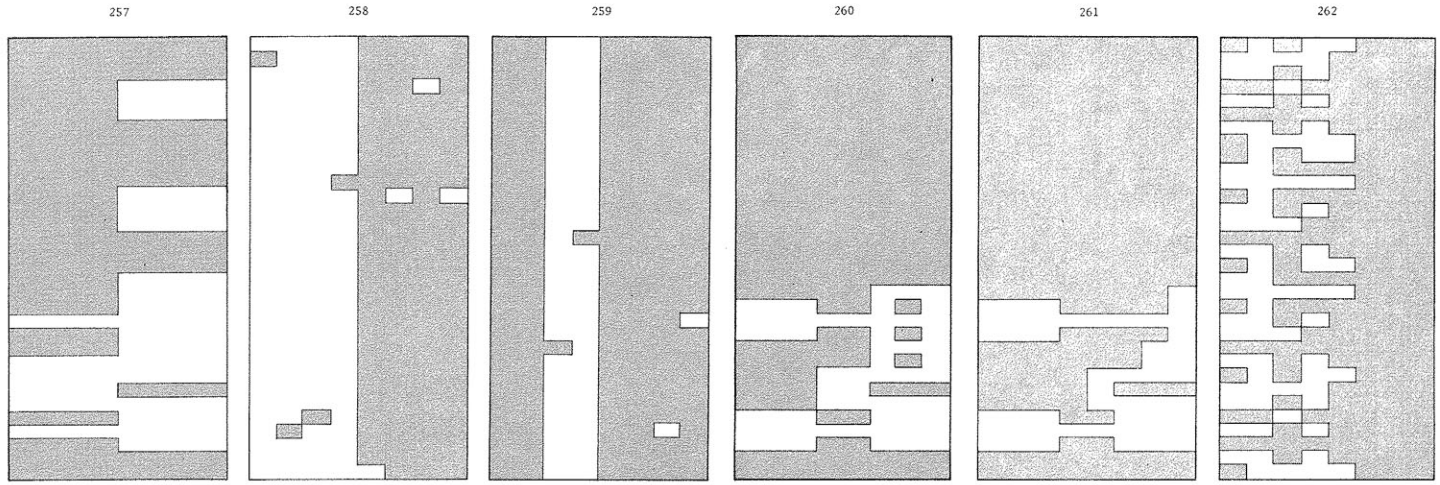
Figure 9
ROM Reader Schematic

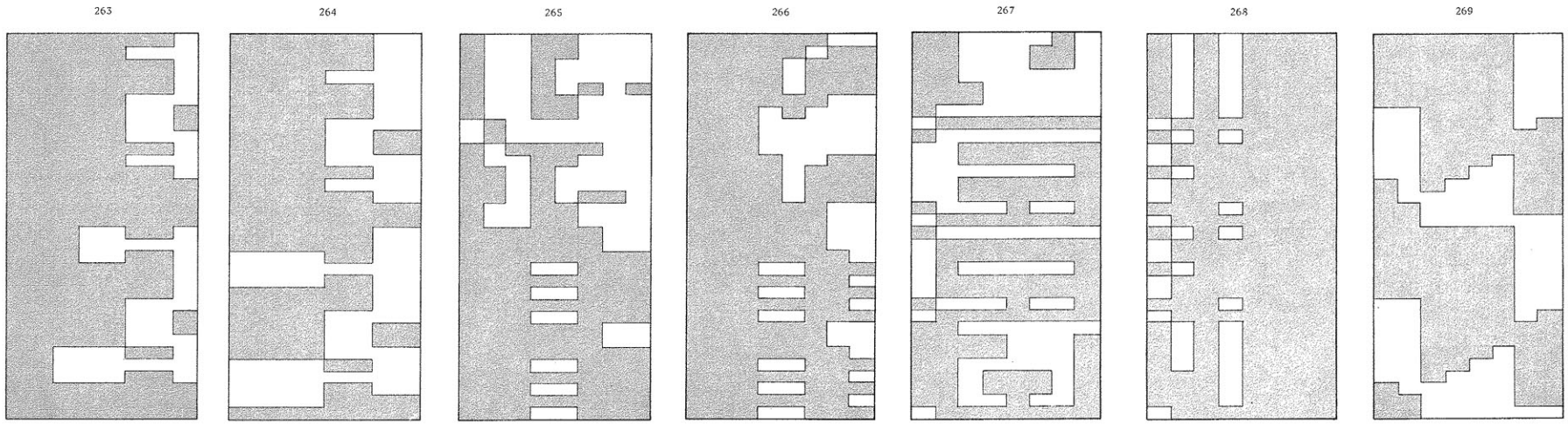
Before we discuss exactly what is happening in the test circuit, we need to make a few relevant comments about the ROMs themselves. The 74188 is a fusable-link, 256-bit programmable read only memory (PROM) organized as 32 words of eight bits each. Each word is read out be a different five-bit address which causes all eight bits of that word to appear simultaneously at the Y1 to Y8 outputs.

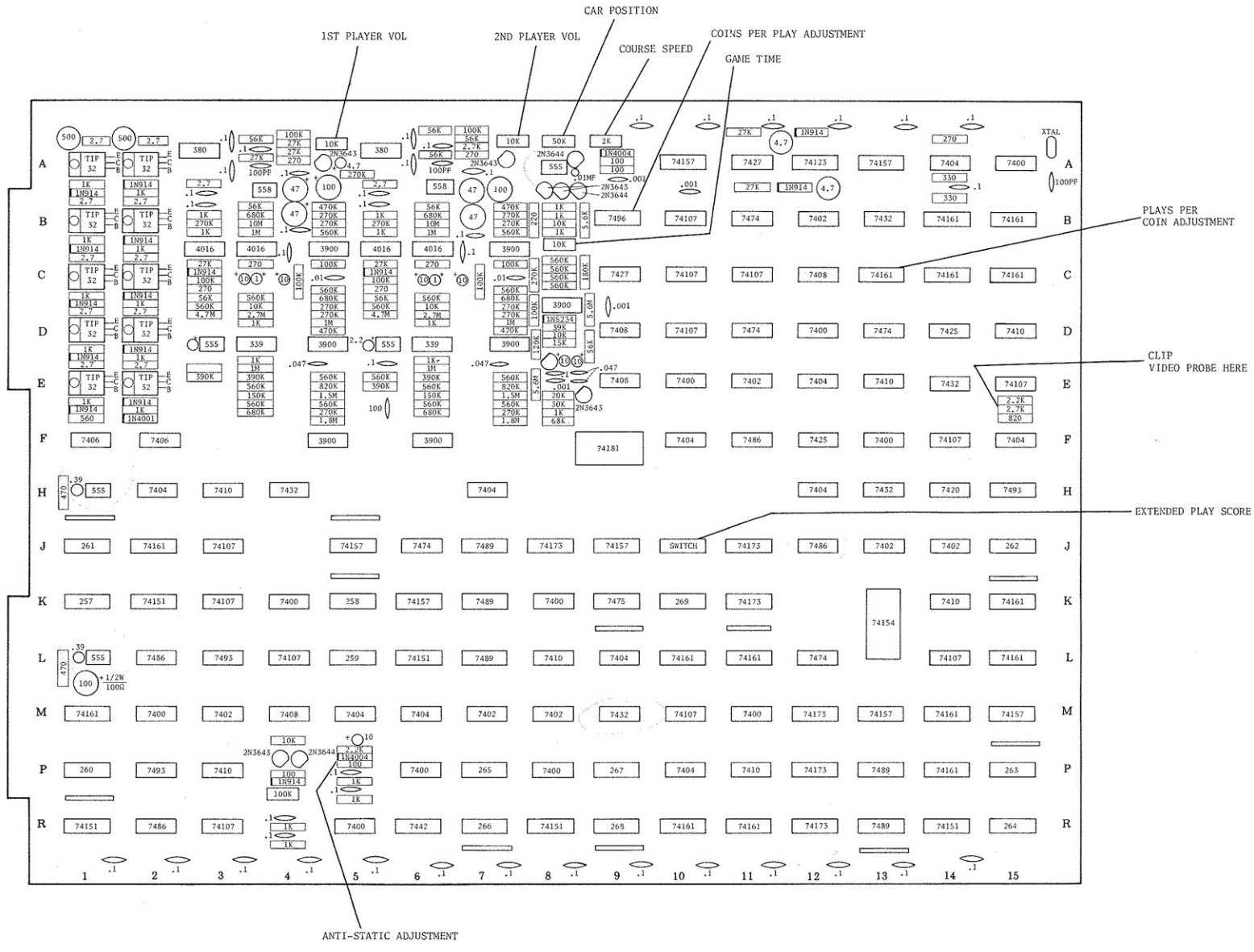
The data can be electrically programmed at any of the 256 bit locations. Prior to programming, the memory contains LOs at all locations. The programming procedure simply blows fusable metal links which results in HI logic level conditions at the selected locations. The procedure is irreversable and, once programmed, the output for that bit is permanently programmed HI. Since the 74188 PROM is directly replaceable with the 74488 mask-programmed ROM, do not be alarmed if games in the future contains the less expensive mask-programmed device.

When plugged into the test circuit illustrated in Figure 9, the ROM data is organized into a rectangle which contains eight bits across the top by 32 words down the CRT. The data is read out by a simple sync progression and directed to appear within the rectangle by the multiplexer. Since the multiplexer is strobed by 128H, the information is enabled only when 128H is LO which occurs in a wide vertical band in the center of the CRT (Figure 4). Also, notice that we have connected the signal from E12-12 to the strobe as well, which prevents sync disruption. The windows for the individual bits are developed by the code at the A, B and C selects which divides the 128H wide window into eight smaller 16H wide bit-windows.









THE PRINTED CIRCUIT BOARD COMPUTER

THE POWER SUPPLY

The Power Supply consists of a transformer and a separate power supply PCB containing four distinct sections which output a number of different voltages, both regulated and unregulated. The +12, -12 and -0.7 volt supplies are used in the analog circuitry and the +5 volt line powers the integrated circuits.

The line voltage is initially filtered by a Corcom line filter, which removes undesired transients before the transformer primary. The transformer has two secondary windings which produce 10 and 12 volts AC and these voltages are sent to the following circuits for processing:

(1) +15 VOLTS: The 12 VAC line is full-wave rectified by two 1N4720 diodes and the resulting unfiltered, unregulated pulsating waveform (about 15 volts peak-to-peak) is sent directly to the lamp driver circuit via pin 19 of logic jack J1.

(2) -12 VOLTS: The 10 VAC waveform from the other secondary is full-wave rectified by two 1N4004 diodes and the resulting voltage is negative with respect to ground because the cathodes of the diodes are wired to the transformer secondary. This pulsating waveform is filtered by a 2200 MF capacitor resulting in a filtered, but unregulated, -12 volt supply which is available to the analog circuitry through pin 11 of J1.

(3) -0.7 VOLTS: A diode wired across the -12 volt line and ground provides the potential of one diode drop (0.7 volts) which is used to offset the player's accelerator pots from ground and is also available to the analog circuitry through pin 1 of jack J2.

(4) +5 VOLTS: This is a solidly designed, closely-regulated, high current supply which is fully adjustable by a 1K trim pot located on the Power Supply PCB. Initially, the 10 volt AC output from the transformer is full-wave rectified by two 10 amp, 50 volt (10A50) diodes. This current is connected to a Darlington pass transistor (PMD12K40) which is controlled by a 2N2905 transistor operated by a LM305 integrated voltage regulator. This configuration is a standard current-limited, foldback supply where the amount of load controls the LM305. The load is sensed by a voltage divider network consisting of a 270Ω and a 27Ω resistor which causes the LM305 to supply more or less base current to the Darlington pass transistor. As the base current to the Darlington rises, it supplies more output current, thereby stabilizing the voltage with respect to the load.

The voltage adjustment pot controls the LM305 feedback input and is connected between a voltage divider network wired across +5 volts and ground. As the potential at the pot increases, the 2N2905 is turned on harder, turning on the pass transistor harder which then supplies more voltage.

NOTE: The following test point measurements are made with a 20,000Ω/V Volt-Ohmmeter at the indicated points.

1. +15 VDC
2. +12 VDC
3. -15 VDC
4. -.7 VDC
5. +15 VDC
6. -6 VDC
7. +5 VDC

THIS CIRCUIT IS ADJUSTED WITH A VOM SO THE +5 OUTPUT IS EXACTLY 5.1 VOLTS

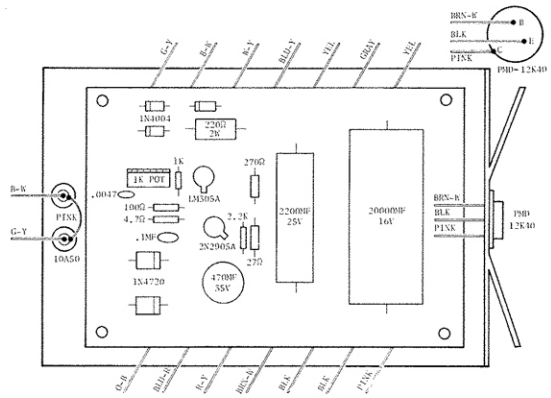


Figure 10
The Power Supply PCB

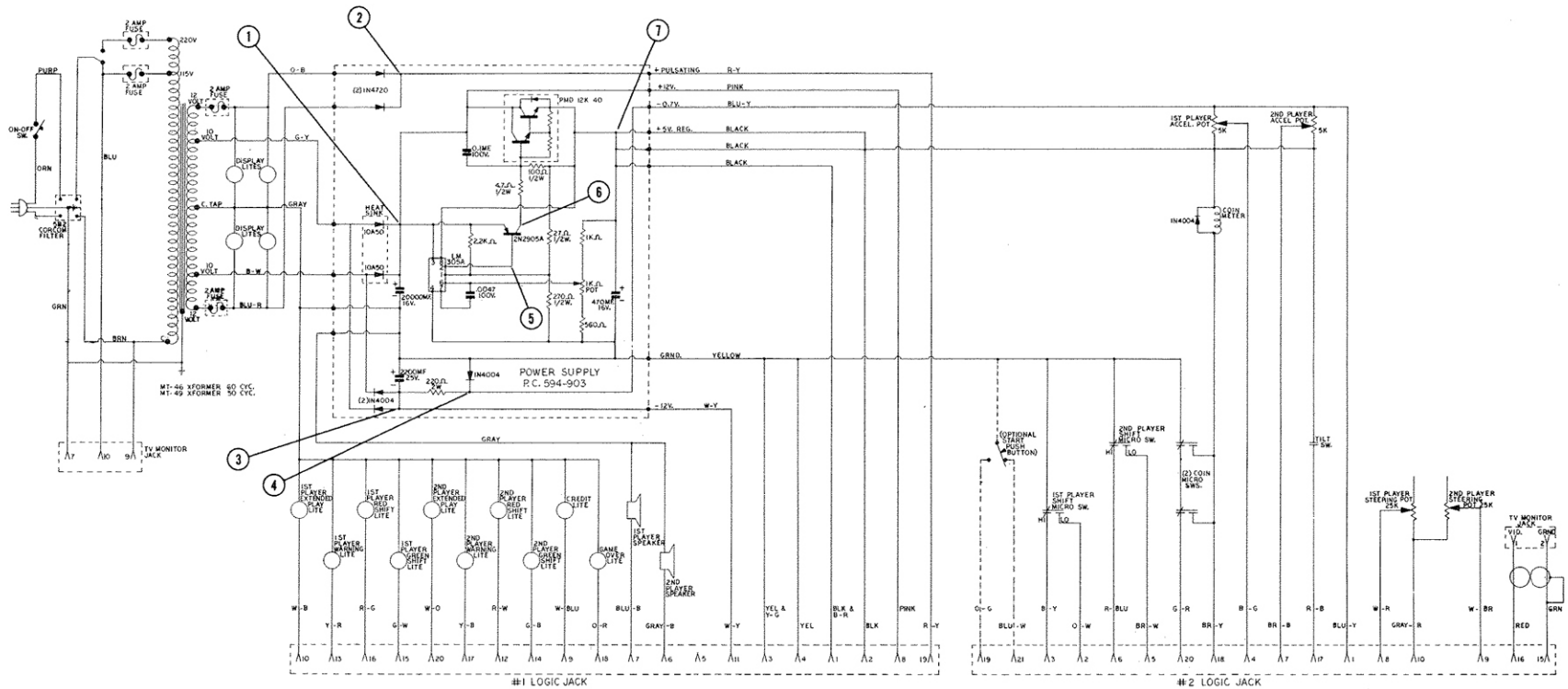


Figure 11
The Power Supply Schematic

THE OSCILLATOR

The Oscillator produces CLOCK, the master timing signal which synchronizes the entire computer. The crystal generates a 14.318 MHz frequency and is kept oscillating by two closed-loop amplifiers which consist of a feedback resistor and an inverter each. The amplifiers operate out of phase with respect to each other and are coupled together by a 0.1μf capacitor. A third inverter serves as a buffer-amplifier for the signal before it enters flip-flop F14, which produces the actual 7 MHz CLK frequency. Initially, the signal is sinusoidal in form, however F14 tends to square up the waveform so it is more acceptable to digital circuits.

MAIN TIMING/SYNCHRONIZATION

The main timing and synchronization circuitry in this game is a bit different from the way timing and sync are ordinarily accomplished. Normally, 7493 ripple-through counters are used and are arranged in such a way that they count to a certain number and are then reset back to zero before repeating the count. The following circuitry, however, utilizes 74161 (9316) synchronously-presettable four-bit binary counters to generate the main timing signals. This type of counter is used extensively in motion circuits because it is capable of being preset to any desired number but is used in this circuit because it is a rising-edge triggered device. By keeping all devices rising edge triggered, many problems are avoided. Because of these counters, the following sync circuitry may be a bit confusing at first, but if you regard it as a motion chain which is preset to a base number before counting, it should be easier to comprehend.

The functions of this circuit are primarily twofold: (1) Main timing generates submultiples of CLOCK which are used in most circuits of the game to provide precisely timed signals used in the development of other, more complex, signals where all signals have a synchronous relationship to CLOCK and to the movements of the electron beam. (2) Some of these submultiples are combined to form sync signals which synchronize the operations of the vertical and horizontal oscillators in the TV monitor with the operation of the PCB computer.

HORIZONTAL MAIN TIMING: The horizontal main timing counter chain divides the frequency of CLOCK 2^9 or 512 times to produce various submultiples of CLOCK known as 1H, 2H, 4H...256H. The first two counters provide eight bits and flip-flop M10 supplies the ninth. When examined with the video probe, the submultiples appear as a series of vertical columns where the width double with each successive division.

The horizontal reset pulse ($\overline{\text{H RST}}$) is produced at AND gate P11 and, although it has been named $\overline{\text{H RST}}$, it is actually a load pulse rather than a reset. If 7493s were used in this circuit, the name $\overline{\text{H RST}}$ would be appropriate for the signal would in fact reset the counters back to zero. However, in this circuit the $\overline{\text{H RST}}$ pulse loads the base number into the 74161s, so it would have been more appropriate to have named this signal $\overline{\text{H LD}}$.

In any case, $\overline{\text{H RST}}$ is generated by the addition of the terminal counts (TCs) from the two counters and from flip-flop M10. Since the terminal count of the first counter is 15, the TC of the second is 240 and the output of M10 256, the resulting reset (read load) pulse occurs at a count of 511 (15 + 240 + 256 = 511).

This is where things may begin to get a bit confusing. The number 511 is actually not the number used to determine line length, although the last point of the line occurs at 511. Since a base or preset number of |1|0|0|1|1|1|0|0| is loaded into the parallel entry terminals of the counters, the horizontal chain always begins counting from this number and the actual line length is 511 minus 57 or 454 clock pulses long. In other words, the $\overline{\text{H RST}}$ pulse occurs on the 454th clock pulse and this point coincides exactly with the time when the electron beam reaches the end of the line at the right side of the CRT.

The result of all this is that each line of the raster has been divided into 454 separate time elements where each element corresponds to a separately addressable point which may be illuminated to form an image. This being true, one can say that 1H equals two clock pulses and that points along the line are spaced 1H apart beginning with 57H and ending at 511H.

The timing relationships used in the development of both H BLNK and $\overline{\text{H SYNC}}$ are illustrated in Figure 12. First, $\overline{\text{H RST}}$ clears flip-flop F14 and then 128H goes HI at 128. However, the count of 128 is actually equivalent to 71 since $\overline{\text{H RST}}$ occurs at 57 (128 - 57 = 71). This HI is clocked through 4 clock pulses later by 4H or on the 75th clock pulse, producing an H BLNK window 75 clock pulses wide which extends from 57 to 132. Any output of F14 is limited to the period of time this window is HI.

HORIZONTAL SYNC: $\overline{\text{H SYNC}}$ is developed at NAND gate H14 and it is interesting to note that two of the signals used in the development of the sync pulse are entirely unnecessary.

In any case, $\overline{32\text{H}}$ at H14-13 is LO until the 64th clock pulse, at which time it rises HI and remains HI until the 96th clock pulse. The result is an $\overline{\text{H SYNC}}$ pulse which is LO between 64H and 96H. Notice that both 64H and 256H are connected to H14 and that both signals are entirely extraneous to the operation of the circuit. They were probably included to avoid a recurring sync pulse, but this extra protection is not necessary since the pulse is limited to the blanking window. Both the inputs can be eliminated with no adverse effect on the circuit.

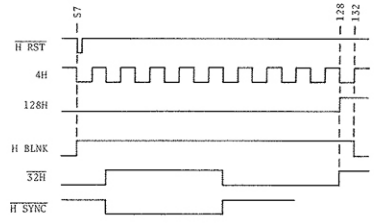


Figure 12
Development of H SYNC

VERTICAL MAIN TIMING: The Vertical Main Timing circuit is virtually identical to the horizontal one except that it counts $\overline{\text{H RST}}$ pulses rather than clock pulses. Since each H $\overline{\text{RST}}$ pulse coincides with the termination of a line, the vertical timing circuit counts lines per field whereas the hoirzontal counts points per line. The vertical reset pulse coincides with the termination of one full field of horizontal lines.

Two counters and a flip-flop provide none divisions of $\overline{\text{H RST}}$ where each successive division is one-half the frequency of the previous one. The nine resulting outputs are called 1V, 2V, 4V...256V. The vertical reset pulse is developed by ANDing the TCs of the counters (15V + 240V) with the output of the ninth-bit flip-flop (256V). The result would be a pulse at 511 except that, since these counters are preset with the base number |1|1|1|1|0|1|1|1| (239), the vertical reset pulse actually occurs with the 272nd $\overline{\text{H RST}}$ pulse (511 - 239 = 272). Therefore, there are 272 horizontal lines per field where each line has a separate address number.

VERTICAL BLANKING: Since V BLNK is the output of the ninth bit of the chain, it is actually equal to 256V. At the time of "reset" (actually load), $\overline{\text{V BLNK}}$ is LO and stays LO until the second counter fills up again. This produces a V BLNK pulse 17V wide which occurs between 239V and 256V.

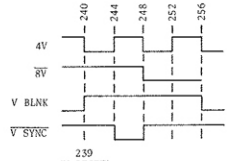


Figure 13
Development of V SYNC

VIDEO SUMMATION

However incongruous it may seem to explain the operation of the Video Summation circuitry before the signal inputs themselves are developed and explained, the fact nevertheless remains that Video Summation is organized with sync on the schematic and therefore must be explained here.

Although the primary function of any video summing circuit it to add all the separate video signals across the appropriate values of resistors and couple them to the monitor, this circuit performs a certain amount of signal development and sync protection as well.

The various displays involved in the summation process are the score, the race cars, the control car(s), the grass, the hay bales and both the stripes. All the video information generated separately for each player (e.g. R CARS, STRIPE) is entered at multiplexer R13 for separation by 128V. The score information (SCR VID) and the control cars (C CAR) are summed later in the circuit and do not require multiplexing as they are separated in the circuits which generate these signals.

Multiplexer R13 is a Quad 2-to-1, however only 3/4 are needed here. Since the device is selected by 128V SW, the information in the one player mode will fill the screen, but each player's information will be separated at 128V when in the two player mode.

FLASH: The FLASH signal is generated by NANDing each player's crash signal (CRSH 1 or CRSH 2) with signal D4 at gate F13. Signal D4 is the very slowly pulsing output of a counter in Section 7. If a crash has occurred, CRSH 1 or CRSH 2 will be HI, enabling this pulsing signal through F13, causing the appropriate player's side to flash.

STRIPE, GRASS & HAY BALES: The grass (those two light areas between the stripe and the edge of the CRT) is developed at B10. Signal 2V is exclusively ORed with 128V at F11 which, when gated with STRIPE, at F13, reduces the width of the stripe to a narrow line at the top of the CRT and two narrow lines at the bottom. It may be difficult for you to see the very bottom line, depending on how the vertical height of your monitor is adjusted. This signal is then used to clock B10. Since $\overline{\text{V BLNK}}$ clears B10, B10-6 is HI when the flip-flop is cleared at the very beginning of the raster. Therefore, the grass is enabled from the top of the CRT until the first stripe occurs which disables F13 and toggles B10 and turns off GRASS. GRASS then remains LO all the way down the CRT until the next stripe again toggles the flip-flop. This again enables GRASS and the electron beam is intensified until it encounters the very bottom stripe which again disables GRASS. GRASS remains disabled until $\overline{\text{V BLNK}}$ clears B10 at the top of the CRT.

Notice that the signal GRASS from B10-6 is NANDed with the output of E12-12. There is a very specific reason for this. H BLNK and V BLNK are first ORed together, then inverted before being gated with GRASS. Therefore, the signal resulting from D12-11 will be disabled during both vertical and horizontal blanking which is necessary to prevent GRASS from disrupting sync. The signal from E12-12 simply "locks out" GRASS during horizontal and vertical sync to prevent tearing and other undesirable effects.

After the grass is fully developed and locked out during sync, the resulting signal is gated with the hay bales (RUNR 1 and RUNR 2) at B13 which passes the grass signal, but limits the hay bales to the height of the grass window. Inversion is performed at A15 to make the image white before entering the multiplexer.

As mentioned before, the width of the stripe window is reduced by the signal at F11-6 before it is chopped up by SR116 and R216 at E14 and H13. SR116 and R216 are vertical columns which serrate the stripe window so it appears as a dotted line and the result is inverted before multiplexing so it appears white. Since SR116 and R216 are motion outputs from the Course Speed circuit, the dotted line moves at the course velocity.

The rest of the video is entered at F12 and notice that this gate is equipped with a strobe input connected to E12-12 to lock the signal out during sync. NOTE: ONLY TEXAS INSTRUMENT'S 7425 HAS THE STROBE INPUT BONDED TO THE CHIP, SO IF THE ORIGINAL CHIP FAILS AND IT IS REPLACED WITH A 7425 MANUFACTURED BY ANOTHER COMPANY, THE SIGNAL WILL NOT BE LOCKED OUT DURING SYNC AND THE RESULTING VIDEO WILL BE COMPLETELY DISTORTED.

8. LP: HI, LO & PULSING.
   SC: Pulses HI & LO for .1397μs.
9. LP: HI & PULSING.
   SC: Pulses LO for .1397μs every 63.434μs.
10. LP: HI & PULSING.
    SC: Pulses LO every 17.2ms for 63.434μs.
11. LP: LO & PULSING.
    SC: Pulses LO every 17.2ms for 253.7μs.
13. LP: HI & PULSING.
    SC: TP12 serrated by TP11.
14. LP: HI, LO & PULSING.
    VP: Light area outside borders.
15. LP: HI & PULSING.
    VP: Light horizontal stripe defining the borders.
16. LP: HI, LO & PULSING.
    VP: Light and dark columns moving with top hay bales.
17. LP: HI, LO & PULSING.
    VP: Light and dark columns moving with bottom hay bales.
18. LP: HI & PULSING.
    VP: Fully developed upper stripe moving at course velocity.
19. LP: HI & PULSING.
    VP: Fully developed lower stripe moving at course velocity.
20. LP: LO going HI when C CAR #2 crashes.
21. LP: LO going HI when C CAR #1 crashes.
22. LP: Slowly pulsing signal.

23. LP: HI & PULSING.
    VP: Top hay bale signal inverted and extending down the CRT.
24. LP: HI & PULSING.
    VP: Same as above TP except for bottom hay bales.
25. LP: HI, LO & PULSING.
    SC: Pulses HI every 63.434µs for 10.4µs & every 17.2µs for 1.08ms.
26. LP: HI & PULSING.
    VP: GRASS darkens.
    SC: Pulses HI every 63.434µs for 10.4µs when GRASS is LO.
27. LP: HI when in 1 player mode; HI; LO & PULSING when 2 player.
    VP: No visible signal during one player mode; CRT divided in half during 2 player mode.
28. LP: LO & PULSING.
    VP: Top & bottom stripes visible.
29. LP: HI going to HI, LO & PULSING when CAR #1 crashes in 1 player mode; pulses faster when either car crashes in 2 player mode.
30. LP: HI & PULSING.
    VP: Lightens hay bales.
31. LP: LO & PULSING.
    VP: Lightens R CAR images.
32. LP: LO & PULSING.
    VP: Lightens C CAR image(s).
33. LP: LO & PULSING.
    VP: Lightens score images.
34. LP: HI & PULSING.
    VP: Darkens GRASS images.
35. LP: HI & PULSING.
    VP: Darkens STRIPEs, C CARs, R CARs and SCORE.
36. LP: HI, LO & PULSING.
    VP: Light column extending from left edge of CRT to 16H to right of score numbers.
37. LP: HI during game over and 2 player; LO during 1 player.
38. LP: Same as TP36 during 2 player operation; LO for 1 player.
    VP: Same as TP36 during 2 player operation; LO for 1 player.

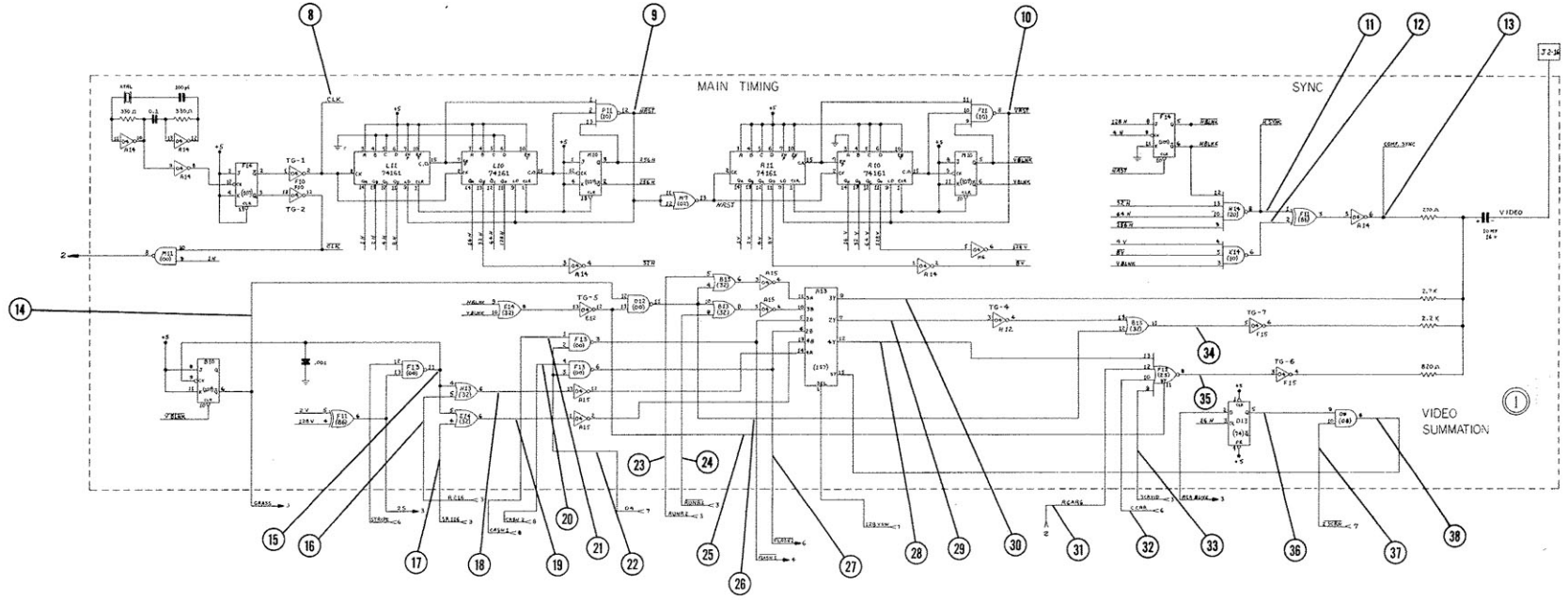


Figure 14
Main Timing/Sync/Video Summation Schematic

R CAR GENERATION & CONTROL

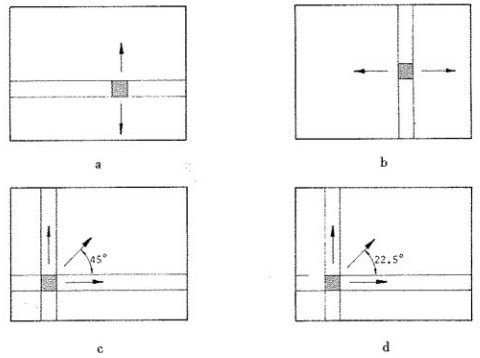
INTRODUCTION: The computer-controlled or race car (R CAR) images are both developed and controlled in this section. The degree of complexity found in this circuit is much greater than in most motion circuits because not only are there two normal motion circuits, but the section also contains complex window control circuits as well as an unusual "scratch pad" memory. This complexity combined with the fact that the schematic is drawn with signal processing logically proceeding from right to left and from bottom to top may tend to make this circuit a bit difficult to grasp at first.

To make comprehension easier, the entire circuit may be subdivided into smaller, more easily manageable, sub-sections. Two standard horizontal motion circuits are located at the extreme right of the section and these consist of the familiar nine-bit presettable counter chains plus two separation multiplexers. The outputs of the first chain address ROMS 263 and 264 to retrieve information used in the generation of the car images. The remainder of the circuitry to the left of the ROMs develops signals which select and strobe multiplexer R14 and this circuitry is also divisible into two sections. Counter H15 addresses ROM J15 which reads out a stored code used to randomize windows generated at K15. These windows are temporarily stored in RAMs P13 and R13 which in turn control latches R12, P12 and M12. The latch outputs then select R14 and form the complete car images.

THE MOTION CIRCUITS: When the machine is conditioned to the two player mode, these circuits control the horizontal motion of both sets of race cars which are separated by a line passing across the CRT at 128V. When in the one player mode, only one set of cars is enabled which fill the entire screen. We will assume the machine is set to the two player mode for much of the following R CAR discussion. The horizontal motion circuits follow the standard configuration used in every video game, however we will explain motion circuit operation in detail. Vertical race car motion is controlled by the Arithmetic Logic Unit (ALU) found in the Processor, but vertical motion is displayed in this circuit.

The illusion of motion is achieved by rapidly shifting the car image in much the same way the illusion of motion is created on the motion picture screen by the film in the projector. The eye perceives the positional shift as motion because of a phenomenon known as "retinal after-image". This phenomenon causes the last seen image to be "remembered" for a very short period of time and, as the position of the image is shifted a number of times in rapid succession, the eye overlaps the images and the brain interprets the shifting process as continuous motion. This process is enhanced by the persistence of the phosphor coating on the inside of the CRT which causes the image to linger, even after the electron beam is moved or shut off.

The speed or velocity of the image is controlled by the shift rate/frame rate ratio. In other words, if the position of the image is shifted once every frame, it will appear to move much faster than if it were shifted only once every three frames.



The direction of the moving image is controlled by varying the rate at which the vertical and horizontal motion windows are shifted with respect to each other. As the window in Figure a moves up and down, the image comtained within it also moves up and down. Figure b shows the horizontal motion window which controls the left-right motion of the image. When the two windows are combined, the image at the intersection can be vectored at any desired angle by individually varying the rate at which each window is shifted. For example, if the two windows are shifted equally, the resultant vector will be 45° (Figure c). But if the horizontal window is shifted twice as fast as the vertical, the image will move along a 22.5° vector (Figure d).

The motion circuits are constructed almost identically to the sync circuits and they run in parallel with their horizontal sync counterpart. If the operation of the motion circuits is timed with that of horizontal sync, the image will be generated in the same place each frame and will appear to be stationary. However, if the motion circuits develop their signals a little sooner or later than sync, the window will be shifted an amount equal to the resulting differential and will appear to move.

The differential is produced by loading different codes at the parallel inputs of the motion counters which controls the point at which these counters begin. The stop code is that number which times the motion circuits with the sync chain and causes a cessation of motion. Other codes, both higher and lower, cause a change in this relationship and the position of the image is shifted with respect to sync.

HORIZONTAL MOTION: Since horizontal reset is always loaded with the number 57, the code entered at the parallel entry terminals of the motion counters which stops motion is also 57. Notice that all the inputs to these counters are tied either HI or LO with the exceptions of the two least significant bits (LSBs). Since the signal at the A input of L15 and M14 ($\overline{TA1}$ or $\overline{TA2}$) is normally HI, the resulting code at the inputs is |1|0|0|1|1|1|0|0| or 57. $\overline{TA1}$ and $\overline{TA2}$ are produced in Section 6 of the schematic and can be seen with the video probe at the very bottom of the CRT. These signals are developed using Analog Timing and accelerator inputs so the motion count may be controlled via the accelerator and its adjustment pot. Both signals are LO only when active and then for a period of time which causes an increase or decrease in the shift rate/frame rate ratio.

The B input or second LSB of the first counter is controlled by $\overline{CAR\ TRG}$ ORed with 2V and inverted by F15 and this output is normally LO and PULSING. $\overline{CAR\ TRG}$ is produced by clocking stored information from ROM 269. When viewed with the video probe at H13-1, it appears as a 2V wide horizontal line at the very top of the CRT. When ORed with 2V, the result is a one line wide signal. Since this is entered at the 2^1 bit of the counter chain, the number 59 will be loaded once per field.

Varying the two LSBs produces the three motion codes this circuit uses: 56, 57 and 59. Since the number 59 is always loaded once per field, the motion chain is advanced two clock pulses with respect to sync and the images will be shifted two clock pulses per field. The result is that the car image appears to be overtaking the more or less stationary control car (C CAR). This is true only so long as $\overline{TA1}$ or $\overline{TA2}$ is HI and the number 56 is not being loaded at the bottom of the CRT.

As the accelerator is depressed, $\overline{TA1}$ or $\overline{TA2}$ goes LO for one clock pulse. If the accelerator is held down, the pulse width of $\overline{TA1}$ or $\overline{TA2}$ increases so that it is loaded more times per field. If under the least acceleration, $\overline{TA1}$ or $\overline{TA2}$ are LO for only one clock pulse per field which produces a code of 58 at the motion chain inputs. This still advances the motion counters with respect to sync and the race cars appear to be losing ground to the control car. The more the acceleration increases, the more times 56 is loaded and retards the motion chain a greater amount causing the control car to appear to be overtaking the race cars at an ever increasing rate of speed.

The outputs of both counter chains enter the inputs of multiplexers M15 and M13, but notice that the four LSBs of the top chain enter M13. Both are Quad 2-to-1 multiplexers which are selected by 128V SW. When the game is in the one player mode, 128V SW remains HI which selects only the first player's race cars. When in the two player mode, 128V SW acts like 128V and separates the CRT into upper and lower halves. The first player's race cars appear in the lower half and the second player's in the upper half.

THE ROMS: After multiplexing, the four LSBs from the first chain are used to address ROMs 263 and 264. These are 8-bit by 32-word ROMs (see page 2) and two are necessary since the largest size race car is 16 bits long which requires the addition of two eight bit words. Notice that the outputs of both ROMs are OR-tied and since this type of ROM has open collector outputs, they can be operated in this fashion without the outputs of one trying to pull the outputs of the other. These ROMs are addressed by a five bit word where the first four bits are the four LSBs from the counter chain and the fifth is controlled by signal C. Notice also, that both ROMs are enabled at pin 15 by 2 SCRN or $\overline{2\ SCRN}$ which produces two differently sized sets of cars. When the machine is in the one player mode, 2 SCRN and $\overline{2\ SCRN}$ are LO and HI respectively which selects R15 and the larger sized race cars result. Since more images need to be squeezed into the CRT when in the two player mode, 2 SCRN goes HI and $\overline{2\ SCRN}$ drops LO which selects P15 and the smaller sized images appear.

If you examine the inputs to the ROMs with the video probe, you can view the address patterns which bear no particular meaning other than this is the way the information is organized within the ROM. However, if you examine the ROM outputs, you can see the horizontal car image information being read out of the ROM (Figure 15). For example, pin 4 of P15 reveals the horizontal information used to generate the wheels, P15-7 produces information for the axles and body and P15-9 creates the center of the car body. R15 stores very similar information except for the larger sized cars.

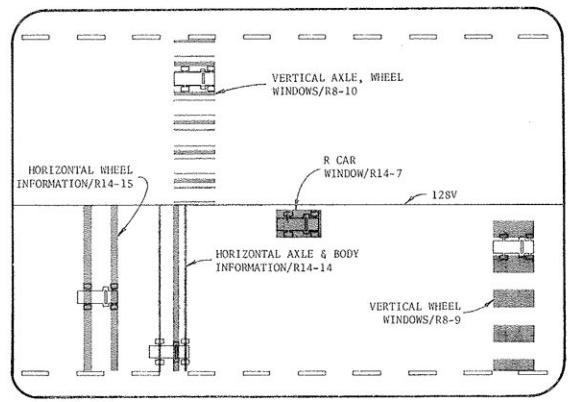


Figure 15
R Car Generation Data

This information enters multiplexer R14 where it is selected by vertically moving windows to produce the complete car image, but more about this later.

HORIZONTAL WINDOWS: This section consists of decoder K13, counter H15, ROM J15, NOR J14 and OR H13. Decoder K13 is a 4-to-16 addressed by the four MSBs from the bottom counter chain. The result is a series of 32H wide windows moving according to the motion chain differential. These horizontal windows are divided at 128V and are areas which limit the image horizontally. The windows are combined with the outputs from ROM 262.

ROM 262 is addressed by counter H15 which simply counts from 0 to 15 extremely slowly. H15 is clocked by WRN which is signal V RST divided 32 times or, in other words, a pulse occurring every 32 fields. If you happen to own a logic clip, this is an excellent place to use it because the counter is operating so slowly that it can actually be observed. At any rate, the outputs sequentially address the ROM which then reads out a simple stored code. This code provides the psuedorandom signal which causes the windows from K13 to move in an unpredictable fashion.

THE MEMORY CIRCUIT: After combination with the ROM outputs, the windows are temporarily stored in RAMS P13 and R13. The RAMs provide a "scratch pad" memory for the psuedorandom motion. Note that there are two memories and that P13 is selected by $\overline{128V\ SW}$. If the machine is set to the two player mode, P13 operates the top cars and R13 serves the bottom ones.

There are 16 different locations for storing four-bit words in the RAM, although only three are used here. The data is written during the one clock pulse wide WE (Write Enable) pulse which occurs coincidently with the first clock pulse of the line. The data is read out for the rest of the line by the count from the four MSBs of the motion chain. The result is 16 three-bit words which are used to control the information coming from the 74173 latches.

The latches are equipped with tri-state outputs (low impedance HI and LO states, but high impedance off state) which permits OR-tying of the outputs. Inputs M and N are control lines for turning the outputs on or off and these inputs are active LO. G1 and G2 are data enables and the data is loaded when these controls are both LO.

The vertical motion signals FL1, FL2, FL4 and FL8 are produced by the Processor and are moving horizontal bars which are ORed with each other. This information is loaded into the appropriate latch by FC3, FC5 or FC7 in conjunction with the output from J12-8. FC3, FC5 and FC7 are also produced by the Processor. The vertical motion information is then read out by the output of M9-11 in conjunction with the outputs from the RAMs. The output of M9-11 is developed by RCR BLNK. You can see with the video probe that RCR BLNK is the horizontal component of the score window and it keeps the race cars from appearing in the score area. Since this signal is LO all the way across the CRT except in the score area, the M inputs of the latches are enabled for most of the line and the outputs will be enabled as soon as the N inputs drop LO.

If you examine the data inputs to R14 with the video probe, you will notice the ROM information entering. For example, the horizontal wheel information for the small car appears at pin 2. At pins 9, 10 and 11 however, you will see vertically moving windows which select only certain parts of the ROM information so that it is limited in the vertical dimension as well. Since this signal is moving vertically, the vertical motion of the resulting image is controlled. The strobe input at pin 7 of R14 contains complete rectangular windows for the car images. These windows are 16 lines high by 32 clock pulses long for both sizes of cars.

TROUBLESHOOTING THE RAMS

Random Access Memories are used both in this circuit and in the Processor. These RAMs could conceivably be tested by entering a known word, controlling the address location of that word and checking the output word with a logic probe while manipulating the ME and WE pins the whole time. But since this requires controlling ten pins while reading four others, it would seem that this technique is unnecessarily difficult. Since there are a number of RAMs in this game and since the correct operation of these devices is both critically important and difficult to check, we are recommending the following troubleshooting procedure.

This technique involves comparing the suspected RAM on the PCB with a known-to-be-good reference RAM. It does not require removal of the suspected RAM, however a simple comparator circuit will have to be built.

Building the comparator requires an IC clip, three IC sockets (2-14 pin and 1-16 pin), a reference RAM (7489), a package of exclusive ORs (7486), a hex inverter (7404), four resistors (220Ω) and four LEDs. Wire the IC clip pins to the other pins indicated in Figure 16 in such a way that, when the clip is placed on the suspected RAM, the pins of that RAM will be connected to the comparator as indicated. Use #22 AWG stranded wire to connect the IC clip and #30 AWG wire for the rest of the circuit.

Since the outputs from both the reference and the suspected IC are all connected in matched pairs to the exclusive OR gates, the output of the OR will go HI if the inputs do not match. This HI is inverted by one of the inverters (which also act as buffers) and the resulting LO sinks enough current through the LED to light it. Therefore, any incorrect test RAM outputs will be immediately indicated by a lighted LED.

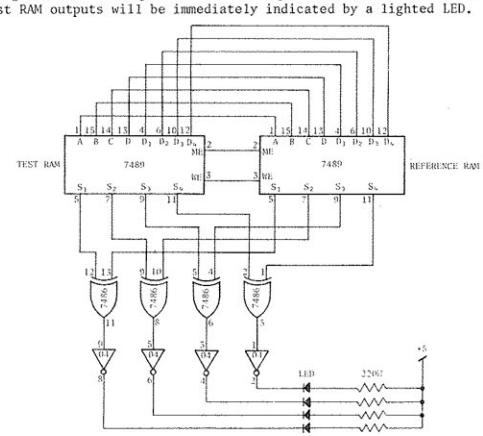


Figure 16 The RAM Comparator Circuit

39. LP: Very slowly pulsing signal.
40. LP: Pulses at half of TP39 rate.
41. LP: LO going HI when a score of 400 is reached.
42. Same as TP41, except when score of 200 is reached.
43. LP: HI, LO & PULSING.
    VP: Light column extending from left edge CRT to right of score.
44. LP: HI & PULSING.
    VP: Vertically moving horizontal bars disappearing & reappearing.
45. Same as TP27.
46. LP: LO & PULSING.
    SC: See timing diagram $\overline{\text{PHI 2}}$.
47. Same as TP37.
48. LP: HI, LO & PULSING.
    VP: Looks like 1H.
49. LP: HI & PULSING.
    VP: Narrow dark bar at bottom of CRT. Becomes wider and narrower depending on setting of SET TOP SPEED and ACCEL pots.
50. Same as TP49.
51. LP: HI & PULSING.
52. LP: HI & PULSING.
    VP: Dark vertical line moving with 1st player's R CARs.
53. LP: HI & PULSING.
    VP: Dark vertical line moving with 2nd player's R CARs.
54. LP: HI & PULSING.
    VP: Dark horizontal windows moving with R CARs.

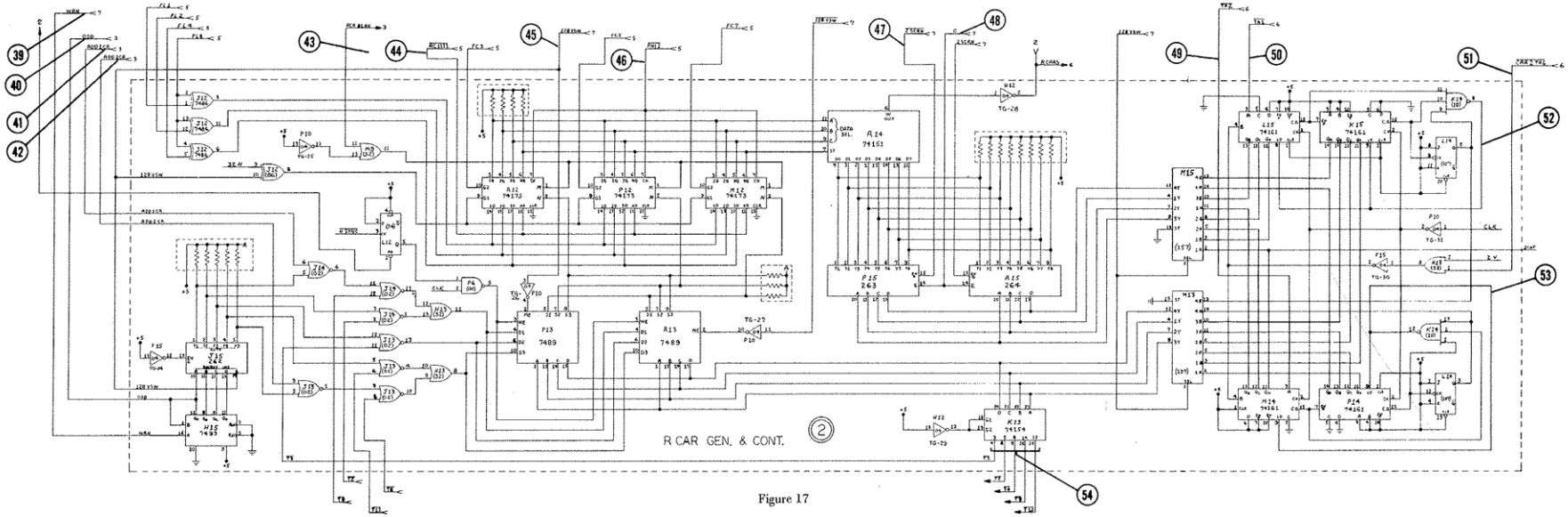


Figure 17

SCORE & TIME DISPLAY

Looking at the score display on the CRT, it is easy to assume this is just another seven-segment display. That is, until the numeral "1" comes up. The shape of this numeral should immediately indicate that this is not a normal score display and, when the schematic is examined, you will in fact see that two ROMs (267 and 268) are used in this circuit. Whereas a seven-segment display creates different numerals by turning the a, b, c, d, e, f and g segments on and off through a BCD to seven-segment decoder, this circuit simply stores and recalls entire numerals. Additionally, this circuit does not count the score, it merely displays it. The score storage functions are performed by the Processor which outputs HL score signals to the display circuit. The whole numerals stored by the ROMs and individually selected at multiplexer R8 by the HL signals are strobed by a window signal which causes them to appear in the proper places on the CRT.

Since there are two OR-tied ROMs, they must be selectively enabled and this is done by entering HL8 and $\overline{\text{HL8}}$ at the appropriate inputs. Since a single ROM would not hold enough information, two are used where each stores a few numbers. The actual numbers are read out by 2H, 4H, 8H, 2V and 4V at the address inputs. The complete numerals are always being read out by the resulting address code. For example, if you examine R8-13 with the video probe, you will see the CRT filled with 1s and 9s. The numeral 1 is stored by one ROM and the 9 by the other and since the ROMs are OR-tied, both are displayed simultaneously. To display only one numeral, we must disable the other ROM.

All of the numerals appear at the inputs of R8 and the individual ones needed are selected by the conditions of the signals at the A, B and C inputs. For example, to select the numeral 4, the top ROM is enabled and the code 1 1 0 is entered at the A, B and C select lines of R8.

Three different sets of numbers may be displayed on the CRT depending on whether the machine is in the one or two player mode. In the two player mode, the time information is displayed in the center of the score window and consists of two numerals which count down from 99. Each player's score appears in his side of the score window and is displayed by three digits which count up from 0 0 0. If in the one player mode, only the time and first player's information appear.

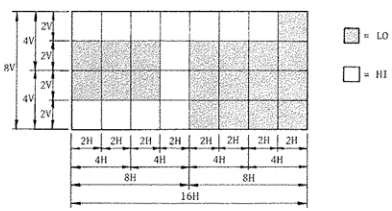


Figure 18
Numeral Generation

The horizontal dimension of the score window (RCR BLNK) is developed by C11 and its P8 gate. The vertical windows change according to how many players are in the game and if a game is in progress or not. This information is developed by the remaining P9 gates and both the horizontal and vertical windows are then gated together to form completely bounded areas for the numerals.

The development of the horizontal score window is a bit unusual. Remember that H BLNK goes LO at horizontal reset and stays LO for 75 clock pulses and it clears C11 when it drops LO. After 75 clock pulses, it returns HI and this HI is clocked through on the falling edge since C11 is a master-slave flip-flop. At this time 16H is LO, but at 144H 16H rises HI and stays HI until 160H at which time it again returns LO and toggles the flip-flop. Therefore, RCR BLNK occurs between 57 and 160 or for a total of 103 clock pulses. Although the right hand boundry of this window occurs at the correct place, the whole window is too large, so RCR BLNK is gated with 16H at P8 and the result is a 16H wide horizontal score window which occurs within the RCR BLNK window. The horizontal score window is then divided up vertically into individual score windows for the time and players by the signal from P8-6.

The conditions of the vertical inputs determine whether the score for one or two players is displayed. If in the one player mode, 2 SCRN is LO and this signal disables V CHAR 2 (Vertical Characters-2nd player) and a HI is produced from P8-3. This HI enables $\overline{\text{V CHAR 1}}$ through P8-11 and, if examined with a video probe, this signal appears as two vertical windows which provide areas for the first player's score and time information. However, if the game is in the two player mode, 2 SCRN is HI which allows V CHAR 2 through P8-3 and this signal is superimposed on top of $\overline{\text{V CHAR 1}}$ and provides the upper window as well. Since these windows are much too large for the numerals, the vertical window information at P8-11 is gated with 8V and the result at P8-6 is windows which are properly spaced apart from each other vertically and turned on for the correct players

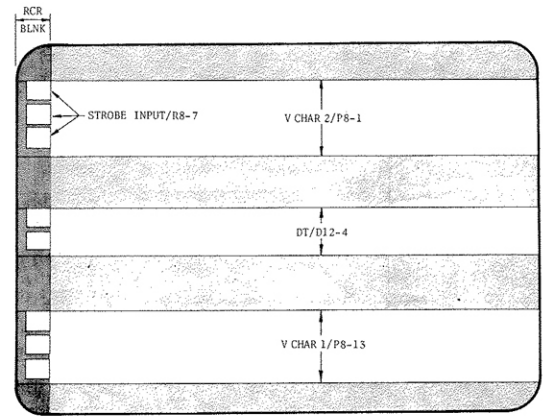


Figure 19
Score & Time Display Windows

Notice that the B select (pin 10) of R8 is controlled by the output of C12-8. This signal is used to prevent the time display from counting when the game is not being played. G.O. (Game Over) is the end game signal and it is HI during the attract mode (before the game is started) and LO during the play mode. This signal controls DT (Display Time) which is the vertical window for the time numerals. During the game, G.O. is LO and DT is HI only during the time display window. Therefore, the output at D12-6 is LO only during the time display window and then only while a game is in progress. Only during this time can the information at HL2 read out numerals other than zero for the time display.

Since the outputs from P8-8 and P8-6 are normally HI, M9 acts essentially like an AND gate waiting for two LOs before M9-8 will go LO which activates the active LO strobe input of R8. Therefore, R8 is enabled only during the horizontal score window produced from P8-8 and during the vertical window from P8-6.

ADD CARS/EXTENDED PLAY

This circuit adds more race cars to the field as scoring increases to increase the degree of difficulty. It also generates the extended play signals $\overline{XP1}$ and $\overline{XP2}$ which enter the Game Control circuit and extend the game length as an award for achieving high score. The point at which the extended play is awarded is user-programmable by changing the positions of the switches located at PCB position J10. $\overline{D7}$ and $\overline{D8}$ are also produced in this section and are used by Game Control to end the game.

The ROM stores a simple code, wherein the specific word is selected by the code at the HL score inputs from the Processor. In other words, the ROM output is determined by the player's score. ODD is simply $\overline{V\ RST}$ divided 64 times, or a very slowly pulsing line (approximately once per second).

The first, second and either the third, fourth, fifth or sixth bits of the ROM (depending on which switch is closed) are entered at the inputs of the latches. Each player has a separate latch where J11 is used for the first player and K11 for the second. The latches are alternately selected by 128V SW which divides the screen during the two player mode. The latches are clocked by $\overline{MSD1}$ or $\overline{MSD2}$ so the latches are enabled only when the most significant digit of the score is being displayed.

Let us assume we have closed the first score adjustment switch. In the truth table, this corresponds to connecting the line underneath "2". Therefore, when the score accumulates to 300, extended play will go into effect when the connected line goes LO. Cars are added when the information from the ROM goes HI. For example, ADD 1 CAR occurs when a score of 400 is reached and ADD 2 CAR at 200.

ODD	SCORE		D7 D8	0 1 2 3 4 5 6 7	HL1 HL2 HL4 HL8	SCORE REQUIRED FOR EXTENDED PLAY	
ODD=0		0		0 0 0 0 0 0 1 1	0 0 0 0		
		1		0 0 0 0 0 0 1 1	1 0 0 0		
		2		0 0 0 0 0 0 1 1	0 1 0 0		
		3		0 0 0 0 0 0 1 1	1 1 0 0		
		4		0 0 0 0 0 0 1 1	0 0 1 0		
		5		0 0 0 0 0 0 1 1	1 0 1 0		
	9	6		1 1 0 0 0 0 1 1	0 1 1 0		
	8	7		1 1 0 0 0 0 1 0	1 1 1 0		
	7	8		1 1 0 0 0 0 0 0	0 0 0 1		
	6	9		1 1 0 0 0 0 0 0	1 0 0 1	600	
	5	10		1 1 0 0 0 1 0 0	0 1 0 1	500	
	4	11		1 1 0 0 1 1 0 0	1 1 0 1	400	ADD 1 CAR
	3	12		0 1 0 1 1 1 0 0	0 0 1 1	300	
	2	13		0 1 1 1 1 1 0 0	1 0 1 1		ADD 2 CAR
	1	14		0 0 1 1 1 1 0 0	0 1 1 1		
	0	15		0 0 1 1 1 1 1 1	1 1 1 1		
ODD=1		16		0 0 0 0 0 0 1 1	0 0 0 0		
		17		0 0 0 0 0 0 1 1	1 0 0 0		
		18		0 0 0 0 0 0 1 1	0 1 0 0		
		19		0 0 0 0 0 0 1 1	1 1 0 0		
		20		0 0 0 0 0 0 1 1	0 0 1 0		
		21		0 0 0 0 0 0 1 1	1 0 1 0		
		22		1 1 0 0 0 0 1 1	0 1 1 0		
		23		1 1 0 0 0 0 1 0	1 1 1 0		
		24		1 1 0 0 0 0 0 0	0 0 0 1		
		25		1 1 0 0 0 0 0 0	1 0 0 1		
		26		1 1 0 0 0 1 0 0	0 1 0 1		
		27		1 1 0 0 1 1 0 0	1 1 0 1		
		28		1 1 0 1 1 1 0 0	0 0 1 1		
		29		0 1 1 1 1 1 0 0	1 0 1 1		
		30		0 0 1 1 1 1 0 0	0 1 1 1		
		31		0 0 1 1 1 1 1 1	1 1 1 1		

Table 1
Truth Table For Add Cars/Extended Play

BORDERS

This section consists of two sub-circuits which generate signals needed by the Processor.

The upper circuit uses sync outputs to develop $\overline{DONOTH}$ which is used in the Processor to halt Processor functions and to generate H SWITCH. H SWITCH is then used to develop V1 and V2. See Figure 21 for the following discussion.

$\overline{V\ BLNK}$ clears C11 at 239V causing the Q output of this flip-flop to drop LO. 2V after $\overline{V\ BLNK}$ returns HI, or at 260V, C11-5 also returns HI producing a pulse which is HI from 239V to 260V. Meanwhile, $\overline{V\ RST}$ is NANDed with $\overline{V\ BLNK}$ at D12. Since $\overline{V\ RST}$ is LO for one clock pulse between 511V and 239V and $\overline{V\ BLNK}$ is LO from 239V to 256V, the inverse addition of these signals produces a HI pulse extending from 511V to 256V from D12-3. This signal and the signal from C11-5 are NORed at B12 to produce a signal which is HI from 256V to 260V. This signal is in turn NORed with $\overline{H\ BLNK}$ at B12 and the output at B12-4 disables the $\overline{H\ BLNK}$ pulses occurring in the interval between 256V and 260V. Since the $\overline{H\ BLNK}$ pulse extends from 57H to 132H, each $\overline{DONOTH}$ pulse is HI during this period.

H SWITCH is developed by NANDing $\overline{DONOTH}$, $\overline{H\ RST}$ and 64H at P11 in the Processor. $\overline{DONOTH}$ forms a HI window from 57H to 132H, part of which is selected by 64H. Since 64H is HI between 64H and 128H, H SWITCH is also HI during this time. $\overline{H\ RST}$ is an extraneous signal.

To develop V1, H SWITCH is ANDed with the signal from C11-5 which disables the H SWITCH pulses from 239V to 260V. V2 is formed by a similar process, except that the signal from D12-3 enables H SWITCH pulses from 239V to 256V.

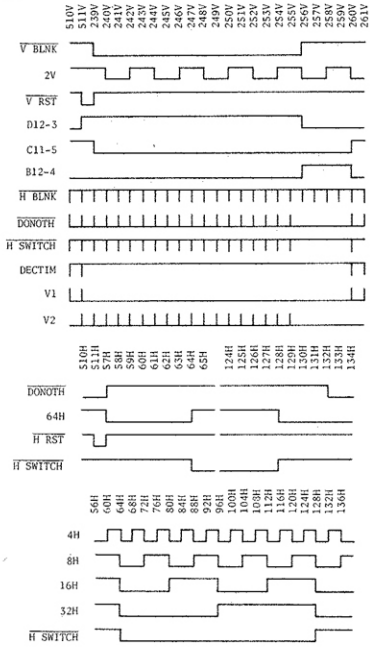


Figure 21
Development of Processor Controls

The bottom circuit develops $\overline{BDRY\ II}$ and BDRY using BDRY I, $\overline{LDVL}$ and 2V. BDRY I is read out from a ROM in Section 6 and it appears simply as a horizontal white bar across the middle of the CRT. NANDing this with 2V reduces the window in width and inverts it so it appears black. The resulting signal ($\overline{BDRY\ II}$) is gated with LDVL (Leading Vertical Divider Line) from the Game Control circuit and again inverted so it appears white. This white line is then ORed with GRASS at M9 and the resulting signal (BDRY) is HI from the top of the CRT down to and including the bottom of the stripe window.

COURSE SPEED

The race course consists of moving dotted white lines (stripes) and sets of bars (hay bales) moving along with the dotted lines. While this circuit does not actually generate these displays (this is done in Video Summation), it does develop signals which cause the final displays to move horizontally and in proportion to acceleration. SR116 and R216 both serrate the stripes and cause them to move while RUNR 1 and RUNR 2 perform a similar function with the hay bales.

The function of a moving race track is to impart the feeling of motion to a more or less stationary control car. As the control car picks up speed, velocity is indicated by causing the track to move past the car, as if the car were actually passing a dotted road line and objects along the track. Even though the control car is not really moving, the player interprets the relative speed between his control car and the moving track and is "tricked" into thinking his car is actually moving.

The basic construction and operation of these motion circuits are identical to the circuits used in the generation of R CARs. Therefore, we will not dwell on the basic concepts of motion circuits and their operation and will refer you instead to pages 7 to 9 for a comprehensive analysis of the subject.

A few specifics do bear mentioning here, however. Notice that, like R CAR, these circuits utilize the analog inputs $\overline{TA1}$ and $\overline{TA2}$ to control acceleration and that both of these signals are connected to the LSB-input of the first counter. However, the course motion circuits, unlike R CAR, have their second LSBs tied LO permanently and there is a very good reason for this. If the second LSBs could go HI, the number 59 would be loaded once per field and the course would appear to be overtaking the control cars at quite a good clip. Since hay bales and dotted lines do not normally move about, it would be quite strange to be passed on the track by a hay bale!

The operation of both TA signals here is similar to that of R CAR. If either $\overline{TA1}$ or $\overline{TA2}$ are HI, the number 57 is loaded into the appropriate chain which stops motion. But if either signal is LO, 56 is loaded causing the count of the appropriate chain to become retarded by one clock pulse. As acceleration increases, the width of $\overline{TA1}$ or $\overline{TA2}$ increases and it is loaded more times per field, increasing motion to the right. As the track display moves to the right, the control car image appears to be travelling to the left.

The 32nd count of the chain (Q_A output of C15 or C14) is called SR116 or R216. When viewed with the video probe, these signals appear as 32H wide vertically moving columns where the accelerator and the speed adjustment pot control the velocity. These signals are then taken to the Video Summation circuit where they are used to serrate the windows for both stripes which turns the solid stripes into moving dotted lines.

The preliminary signal for forming the hay bales is produced at D14 where the Q_A, Q_B and Q_C outputs of the second counters are NORed to produce a non-recurring hay bale window of the desired width. Signal 2S is the result of exclusively ORing 128V with 2V and it is used to separate the hay bale window into a stack of bars. Since the hay bale windows are generated simultaneously with the stripe serration signals, they move together at the same rate.

55. Inverse of TP37.
56. LP: HI, LO & PULSING.
    VP: Dark bars from 368V to 399V and from 432V to 479V.
57. LP: HI, LO & PULSING.
    VP: Light bars from 288V to 335V, 368V to 399V & 432V to 479V.
58. Same as TP43.
59. LP: HI & PULSING.
    VP: Dark windows over numbers.
60. LP: HI & PULSING.
    VP: Dark bar over hundreds digit in 1st player's score.
61. LP: HI & PULSING.
    VP: Dark bar over hundreds digit in 2nd player's score.
62. Same as TP41.
63. Same as TP42.
64. LP: LO & PULSING.
    SC: See timing diagram for V1.
65. LP: LO & PULSING.
    SC: See timing diagram for V2.
66. LP: LO & PULSING.
    SC: See diagram for $\overline{DONOTH}$.
67. LP: LO & PULSING.
    VP: Narrow light bar in center of CRT during 2 player mode; 1/4 down CRT during 1 player mode.
68. LP: HI & PULSING.
    VP: Dark bars in same locations as TP68 except only 2 lines wide.
69. LP: Pulses LO for ≈32.5ms, 32.5 ms after coin switch is closed.
70. LP: HI, LO & PULSING.
    VP: Inverse of TP60 with GRASS added. Hi when TP69 drops LO.
71. Same as TP50.
72. Same as TP49.
73. LP: LO & PULSING.
    VP: Dark vertical line moving with 1st player's hay bales.
74. LP: LO & PULSING.
    VP: Dark vertical line moving with 2nd player's hay bales.
75. Same as TP17.
76. Same as TP16.
77. Same as TP23.
78. Same as TP24.

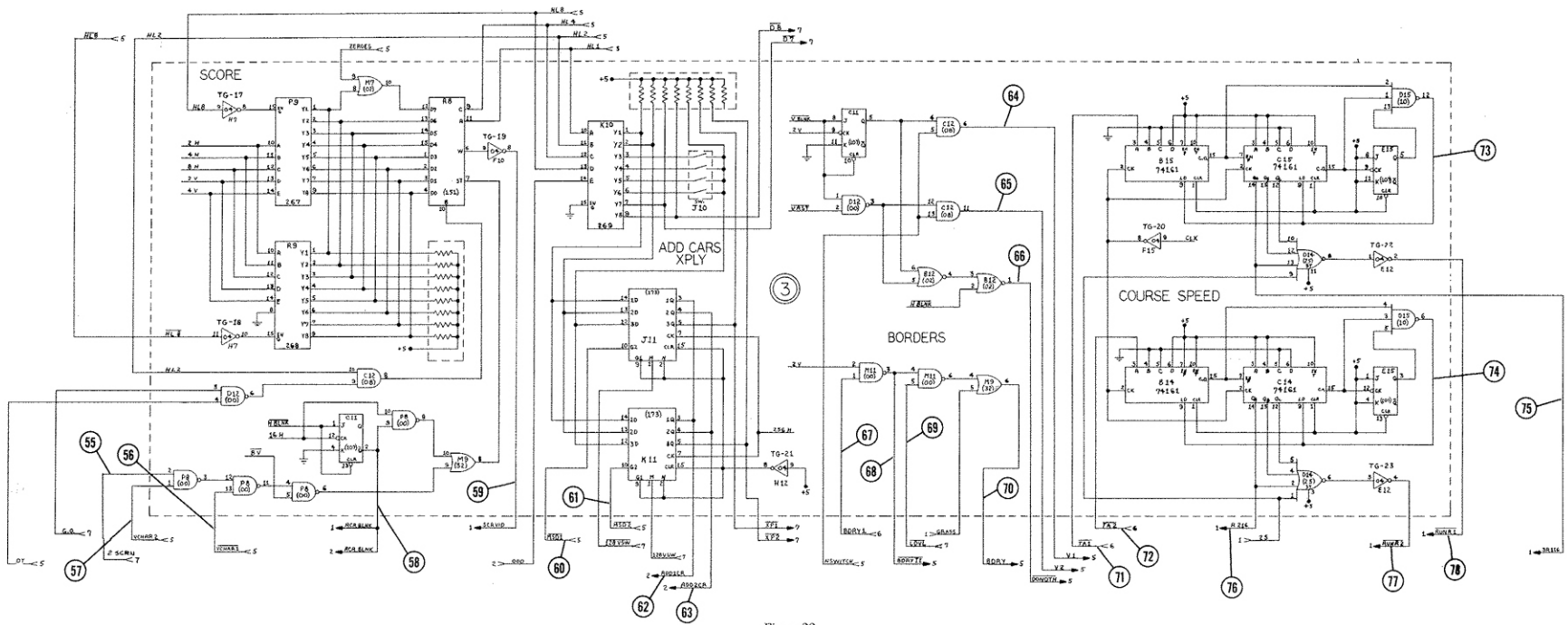


Figure 22
Add Cars/Xply, Borders & Course Speed Schematic

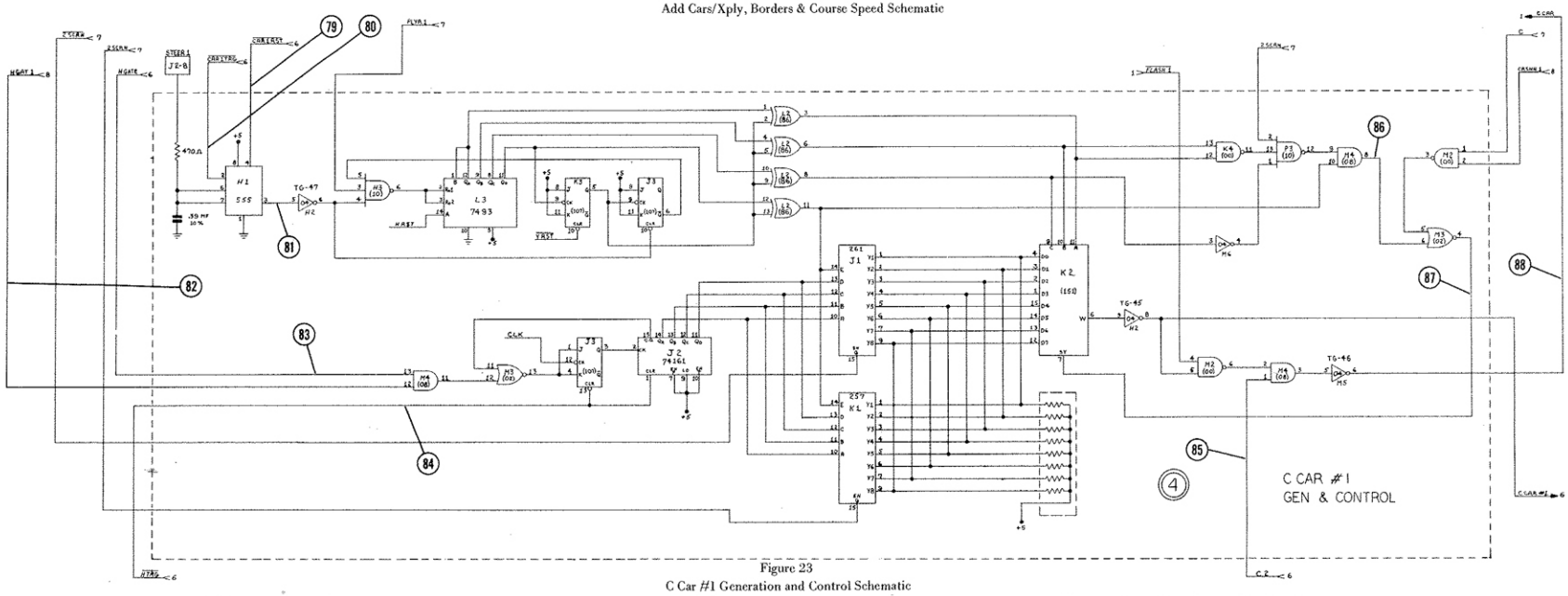


Figure 23
C Car #1 Generation and Control Schematic

C CAR #1 GENERATION & CONTROL

The generation and control of the first player's car is identical to the generation of the second player's car (see C CAR #2, Section 6, page 11) with one minor exception. Since this circuit is used both in the single and in the two player modes, it must be capable of generating two differently sized car images, hence the inclusion of another ROM. The ROMs are selected at pin 15 by 2 SCRN or $\overline{2\ SCRN}$ depending on whether the game is in the one or two player configuration. For example, when the game is in the two player mode, J1-15 ($\overline{2\ SCRN}$) is HI, K1-15 (2 SCRN) is LO and the larger size car image is selected.

Also, notice that the second player's control car (C2) is gated with the first player's at M4 and the result is inverted by M5 before going to Video Summation for display.

79. LP: HI & PULSING.
    VP: Dark narrow horizontal bar 1.5 inches from bottom of CRT.
80. LP: HI & PULSING.
    VP: Dark narrow horizontal bar in center of CRT.
81. VP: Light area extending from center of CRT to a position determined by the setting of the steering pot. Cannot extend beyond the line defined by TP79.
82. VP: Variable width dark vertical column changing with settings of Accelerator and C CAR position (H RAMP) pots.
83. LP: HI, LO & PULSING.
    VP: Light vertical column extending from 256H to 416H.
84. LP: HI & PULSING.
    VP: Dark vertical column extending from 256H to 288H.
85. LP: HI during 1 player mode; HI & PULSING during 2 player mode.
    VP: Inverted image of C CAR #2 in 2 player mode.
86. LP: LO & PULSING.
    VP: Light horizontal bar passing through the center of C CAR #1.
87. LP: HI. LO & PULSING.
    VP: 1H except where light bar in TP86 passed through car.
88. LP: LO & PULSING.
    VP: Intensifies C CAR images.

THE PROCESSOR

The Processor is the most complex circuit in this game, if not in any video game ever produced. Even though this circuit may not appear especially large, the signal flow is difficult to follow, there are many events occurring with extremely complex timing relationships and the entire architecture of the circuit is new to video games. The circuit is essentially a "hard-wired" microprocessor, and so those of you who have had microprocessor experience will find this circuit easier to understand, at least from a conceptual point of view.

Basically, the Processor is little more than an elaborate counter which can be controlled to perform a number of functions at different times on various types of information. Three types of information are handled by the Processor and a different time or mode has been allotted to each. During these modes, the Processor recalls the "old" information from memory, "updates" it and returns it back to the same location in the memory. The three groups of information are (1) the score and time data, (2) the horizontal R CAR information and (3) the data used to control the vertical movements of the R CARs.

But before we delve into the actual operation of this complex circuit, we must first build a solid foundation by describing certain important signal inputs, their timing relationships and various terms and definitions associated with the Processor.

THE PROCESSOR CLOCK: In the analysis of the Borders circuit (Section 3), we discussed the development of a number of signals such as H SWITCH, $\overline{DONOTH}$, V1 and V2. The operation of these signals is extremely important to the Processor as are the signals PHI 1 and PHI 2, the Processor system clocks. Two phases of the Processor clock are generated so that the RAMs can be loaded at different times with respect to each other and so that data can be loaded into the latches at a different time than the data is loaded into the RAMs. The development of both phases of the clock is illustrated in Figure 20. Both signals occur only during the period when H SWITCH is active (between 64H and 128H) and each active H SWITCH period contains 16 PHI 1 and PHI 2 pulses which are out of phase by one clock pulse with respect to each other.

The development of these signals begins when H SWITCH at P3-3 goes HI at 64H and drops LO at 128H providing a 54H wide window at P3 in which PHI 1 and PHI 2 may be developed. Since $\overline{CLK}$ is connected to P3 instead of CLK, the count advances on the falling edge and this is an important factor to keep in mind. Also connected to this gate is the signal 2H and the result is two one-clock-pulse-wide signals which can occur only within the 2H window. This output is connected to one of the inputs of the two M7 NORs and 1H or $\overline{1H}$ is connected to the other input. Therefore, there must be two LOs at these NORs before the output can go HI. For example, when 1H is LO and the signal from P3-6 is also LO, a one clock pulse wide PHI 1 pulse occurs. Even though our timing diagram shows only a few examples, be aware that there are 16 PHI 1 and PHI 2 pulses per each active H SWITCH period. This is a particularly important point because the memories are organized as 16 words of four bits each, hence all locations are addressed during each H SWITCH period.

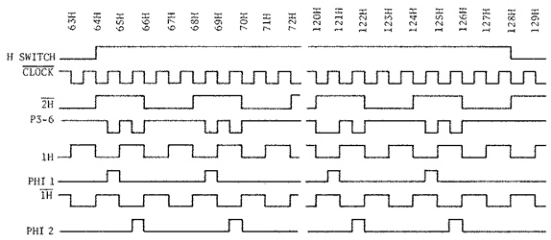


Figure 20
Development of the Processor Clock

THE INSTRUCTION CYCLE: A complete instruction cycle consists of the process by which the data to be updated is recalled from memory, entered in the ALU, has the necessary operation performed on it and is finally returned back into the same location in the memory. Notice that the input latch K9 is clocked by PHI 1 and the output latch J8 is clocked by PHI 2. Therefore, each instruction cycle can be said to extend from the rising edge of PHI 1 to the falling edge of PHI 2.

THE ALU: The 74181 Arithmetic Logic Unit is capable of performing as many as 32 different operations on two four-bit words entered at the A and B inputs. The specific function is selected by the address code at the four function select lines (S_0-S_3) in conjunction with the state of the carry-input (Cn) line. The output code appears at F_0-F_3. Since two of the function selects are tied together, only four of the 16 normal operations are enabled, however this number is exapndable to eight, depending on the state of the carry-input.

ALU SELECTS	C_n=0	C_n=1
0 0 0 0	F=A+1	F=A
0 1 1 0	F=A-B	F=A-B-1
1 0 0 1	F=A+B+1	F=A+B
1 1 1 1	F=A	F=A-1

Table 2
ALU Functions

Whenever the carry-input is HI, the ALU subtracts the number one from the operation is would have performed had the carry-in been LO. For example, when the code at the selects is 0 1 1 0, the ALU subtracts B from A if the carry-in is LO. However, if the carry-in is HI, the ALU subtracts 1 from (A-B).

PROCESSOR MODES: The four Processor modes are controlled by the code developed by mode controls V1 and V2.

MODE	V1	V2
1	0	0
2	1	0
3	0	1
4	1	1

Table 3
Processor Mode Controls

In Mode 1, the Processor is idle and this inactive period is used to read out miscellaneous information from the program ROMs needed in other circuits. The vertical windows for the score and time displays ($\overline{V\ CHAR\ 1}$, $\overline{V\ CHAR\ 2}$ and DT) are stored in these ROMs as are signals $\overline{MSD\ 1}$, $\overline{MSD\ 2}$ and $\overline{MSDT}$ (the Most Significant Digit for the first and second players and for the time).

In Mode 2, the vertical R CAR position information stored in K7 is updated by reading it out of memory, performing the necessary operation on it (e.g. adding one) and re-loading it back into the same memory location. Mode 3 provides a period in which the score and time information stored in J7 can be updated as necessary. Similarly, Mode 4 is used to update the horizontal R CAR position or velocity and this information is stored in RAM K7.

GENERAL PROCESSOR ARCHITECTURE: The addressing of the entire four-bit system is provided by multiplexer K6 which outputs either a fast or slow address code. This system address controls both ROMs, all three RAMs and decoder R6. Although the ROMs do produce certain miscellaneous signals used in other circuits, their primary function is to provide the system program. In other words, the ROMs tell the Processor which function to perform next. The ROMs select L6 which controls the carry-input of the ALU and they also select the ALU function at ALU inputs S_0/S_3 and S_1/S_2. The random logic to the left of L6 is used in conjunction with the ROMs to test for a 10 or 15 condition at the ALU outputs so that proper decimal and binary counting can be achieved. K9 and J8 are input and output latches provided for the ALU which - as mentioned before - provides the actual arithmetic operations. The information processed in this circuit is stored in RAMs J7, K7 and L7 and a different RAM is used for each separate group of information.

SPECIFIC COMPONENT FUNCTIONS: Now that we have developed the input signals to the Processor and described the general architecture of the circuit, we can turn out attention to the specific operation of the components before discussing individual mode operation.

MULTIPLEXER K6: This multiplexer develops the system address which has both a fast and a slow version. When H SWITCH is active, a fast code (4H, 8H, 16H and 32H) addresses the circuit for the first 64 clock pulses of each line. Thereafter, the slow address (16V, 32V, 64V and 128V) is used. This has been done specifically so the information can be written into the RAMs quickly, at the beginning of the line, then read out more slowly, 16 lines at a time.

THE ROMS: Mode control is determined by the states of V2 or $\overline{V2}$ at the chip enables of the ROMs (pin 15) and by V1 at the fifth address bit (pin 14). The state of V2 enables the ROMs alternately and V1 selects the first or second 16 words of the enabled ROM. For example, in Mode 1 V2 is LO which enables P7 and V1 is also LO which limits the address codes to 0 0 0 0 0 to 1 1 1 1 0 (0 to 15). In Mode 2, V2 remains LO, but V1 rises HI to address words 16 (0 0 0 0 1) to 31 (1 1 1 1 1). During Mode 3, V2 goes HI disabling P7 while $\overline{V2}$ drops LO to enable R7. Concurrently, V1 is LO which reads out the first 16 words of R7. In Mode 4, the last mode, R7 remains enabled and V1 selects words 16-31.

MULTIPLEXER L6: This device is an 8-to-1 multiplexer which selects the carry-input of the ALU, hence it controls whether the ALU adds one or subtracts one from the number requiring updating. The address generated by the program ROMs selects the multiplexer output or, in other words, the question to be "asked" by the multiplexer. For example, when the first player's score needs to be updated, the ROM program selects the D6 (pin 13) input and if the first player is scoring, SCR 1 is LO. This LO is reflected at the Y output of L6 which is connected to the carry-input of the ALU and causes one to be added to the score. The second player's score and the time information is similarly updated.

Another important function performed by L6 in conjunction with the program ROMs is the 10 or 15 test. Notice that the outputs of the ALU are all connected to the random logic to the left of L6. The function of this logic is to decode the times when the ALU outputs reach a count of 10 or 15. For example, when counting the score, K8-8 will go HI if the ALU reaches a count of 10. This HI is clocked through by $\overline{PHI\ 2}$ so that on the next instruction cycle, the ALU is told to perform the operation F-A+1 which allows the count to proceed from 10 to 11. Had the count not yet reached 10, K8-8 will be LO and the ALU would have been instructed to perform F=A until the count of 10. The following truth table has been included to aid in troubleshooting this area of the Processor. The only two significant counts occur at 10 and 15 which are indicated by boxes.

DECIMAL EQUIVALENT	ALU OUTPUTS	IF $\overline{PHI\ 2}$ = 1 0 1 0 IF DECTIM = 1 0 1 0 1 0 0 0 1 1 K8-3 M8-4 K8-11 M8-1 M6-2 K8-6 L8-6 K8-8 M8-10 M8-10 M8-10 M8-10
0	0 0 0 0	1 1 1 0 0 1 1 1 1 0 0 0 0 0 0
1	1 0 0 0	1 0 1 0 0 1 1 1 1 0 0 0 0 0 0
2	0 1 0 0	1 1 1 0 0 1 1 1 1 0 0 0 0 0 0
3	1 1 0 0	1 0 1 0 0 1 1 1 1 0 0 0 0 0 0
4	0 0 1 0	1 0 1 0 0 1 1 1 1 0 0 0 0 0 0
5	1 0 1 0	1 0 0 0 0 1 1 1 1 0 0 0 0 0 0
6	0 1 1 0	1 0 1 0 0 1 1 1 1 0 0 0 0 0 0
7	1 1 1 0	1 0 0 0 0 1 1 1 1 0 0 0 0 0 0
8	0 0 0 1	1 1 1 0 0 1 1 1 1 0 0 0 0 0 0
9	1 0 0 1	1 0 1 0 0 1 1 1 1 0 0 0 0 0 0
10	0 1 0 1	0 1 1 0 1 0 1 1 1 1 0 0 0 0 1
11	1 1 0 1	0 0 1 0 1 1 1 1 1 0 0 0 0 0 0
12	0 0 1 1	1 0 1 0 0 1 1 1 1 0 0 0 0 0 0
13	1 0 1 1	1 0 0 0 0 1 1 1 1 0 0 0 0 0 0
14	0 1 1 1	0 0 1 0 1 1 1 1 1 0 0 0 0 0 0
15	1 1 1 1	0 0 0 1 1 1 1 0 1 1 0 0 0 0 0

Table 4 Truth Table For Random Logic

THE ALU & K9: As mentioned before, the ALU is capable of performing only four of its regular repertoire of 16 functions because selects S_0 and S_3 are tied together as are S_1 and S_2. Also, be aware that the ALU can perform an operation on the A inputs alone or on the A and B inputs together. The A inputs are controlled by latch K9 which accepts data from the memories only when PHI 1 is HI. When PHI 1 returns LO, the last data entered is latched and held and the function assigned by the program ROMs performed on it.

MULTIPLEXER J9 & LATCH J8: For the most part, J9 is not used except in two special circumstances and the ALU outputs normally appear at the outputs of this multiplexer. These outputs are latched into J8 during the rising edge of PHI 2 and during the time PHI 2 is HI, the outputs of the latch are written into either J7 or K7. Since H SWITCH is connected to the G1 and G2 inputs of J8, the information is latched only when H SWITCH is active.

Notice that the clear line from output latch J8 is controlled by the output from M8-10. Whenever the score count has reached 10, M8-10 pulses HI which clears J9 and all its outputs drop LO with the result that 0 0 0 0 is entered at the inputs of the score memory. This process provides the carry-over function when the score count changes from 9 to 10 or 99 to 100. When the change from 9 to 10 occurs, zeroes are written into the ones digit location in the score memory and the tens location of J7 is then incremented by one. When the change between 99 and 100 occurs, the same process results except that zeroes are written into both the ones and tens locations and the one-hundreds location is incremented by one.

Although J9 is used infrequently, it does have two important functions: (1) It loads the number 1 0 0 1 into the latch which then loads that number into the memory if the count of the Processor ever reaches 1 1 1 1 or 15. Therefore, numbers greater than 15 cannot be loaded into the memory because at that point the number 10 is dropped in instead. (2) The strobe of J9 is connected to LD, a credit circuit output, which goes HI when the game starts thereby disabling the multiplexer and casuing it to output zeroes into all memory locations. This has a number of interesting effects. Since this resets all the score, time and car information to zero, the timer starts at 00 and is quickly incremented to 01 and the race car images all appear in a straight line at the bottom of the screen but quickly assume their normal positions as the codes begin to change.

THE RAMS: Since the ME (Memory Enable) pin of the RAM must be LO to read or write data, DECTIM (or $\overline{\text{DECTIM}}$) controls which memory is active. Figure 24 illustrates the DECTIM and $\overline{\text{DECTIM}}$ waveforms. J7 is controlled by $\overline{\text{DECTIM}}$ which is normally LO, and pulses HI enabling the memory from 239V to 560V. K7, however, is enabled at every H SWITCH active period except between 239V and 560V. The function of this is to update the score information once per frame, and the car information once per line. But keep in mind that no matter what the condition of DECTIM, the data can only be written into the memories when $\overline{\text{PHI 2}}$ pulses LO.

Another important feature of the RAMs is that they use complementary outputs which is why the $\overline{\text{Q}}$ outputs of input latch K9 are used. This way, the information entered into the ALU is true.

Figure 24

RAM L7 is a little different from the other two in several respects. For one thing, since the ME is always LO, the memory is always enabled. Furthermore, the A input is tied LO so only every other word is capable of being used. Also, the data is never re-written back into L7 like it is into J7 and K7. L7 receives more or less new data each time. The time when the data can be written into this memory is rather limited, unlike the other two. Data is written into L7 only when the output of the ALU is 15, when BDRY is HI or when the LSB from K6 is HI and then only while PHI 2 is HI. This particular timing is developed by E9 and L8 and the reason for the inclusion of BDRY is to give the cars a different horizontal velocity when the car hits the center divider in the two player mode.

The function of NOISE H at L7 is simply to load it haphazardly with random data. NOISE H is a squared-up or "digitized" white noise output of Analog Timing.

MODE OPERATION: Now that we have discussed the important signal inputs to this circuit and the specific functions of the individual components, we can turn our attention to the operation of the Processor during its various modes.

MODE 1: The first mode is used only to read out miscellaneous information from the program ROMs and this data is used elsewhere in the computer. Mode 1 is enabled only when V1 and V2 are both LO and during this time the vertical windows for the time and score information are read out as are the most significant digit signals for the score and time. The following truth table illustrates this information as it is read from ROM P7 and correlates it with the vertical timing of the raster.

	ROM P7 OUTPUT								INFORMATION READ OUT
		DT	V CHAR 1	V CHAR 2			$\overline{\text{MSD 1}}$	$\overline{\text{MSD 2}}$	
	0	1	2	3	4	5	6	7	
0	0	1	1	0	0	1	1	1	256-271
1	0	1	1	0	0	1	1	1	272-287
2	0	1	1	0	1	1	1	1	288-303
3	0	1	1	0	1	1	1	1	304-319
4	0	1	1	0	1	0	1	0	320-335
5	0	1	1	0	0	1	1	1	336-351
6	0	1	1	0	0	1	1	1	352-367
7	1	0	1	1	1	1	1	1	368-383
8	1	0	1	1	1	1	1	1	384-399
9	0	1	0	0	0	0	1	1	400-415
10	0	1	1	0	0	1	1	1	416-431
11	0	0	1	0	1	1	1	1	432-447
12	0	0	1	0	1	1	1	1	448-463
13	0	0	1	0	1	0	0	1	464-479
14	0	1	1	0	0	1	1	1	480-495
15	0	1	1	0	0	1	1	1	496-511

Table 5
Truth Table For Mode 1

MODE 2: During this mode, the vertical position information for the RCARs is updated and the program for this mode along with its associated functions are illustrated in Table 6. This mode is enabled only during the time when V1=1 and V2=0 which reads out the second 16 words of program ROM P7. From the truth table, you can see that the first three bit positions of the ROM are all filled with zeroes for the entire mode and this is a condition which exists for all except Mode 1. The next three bit positions are the selects for multiplexer L6 and these instructions determine the state of the ALU carry-input. In this case, L6 has a choice of looking at data input D_0 or D_3. D_0 is always HI since it is tied to +5, but D_3 goes LO if the carry-output from the ALU is active, or in other words, if a number greater than 15 is trying to come out of the ALU. If a number greater than 15 occurs in the previous instruction cycle, a LO carry-out appears as a HI from the $\overline{\text{Q}}$ output at J6-6 on the next instruction cycle. Since this line is connected to the D_3 input of L6, this HI appears at the Y output of L6 on the next instruction cycle and is inverted by H7 placing a LO at the carry-input of the ALU. This LO causes the ALU to perform the F=A+1 function rather than F=A and the ALU is therefore instructed to add one to the number which it received from the memory. This prevents the ALU from counting beyond the number 16, which it cannot be allowed to do since this is a four-bit system and five-bit numbers cannot exist.

	ROM P7 OUTPUT								DATA READ IN	C_n	ALU FUNCTION	DATA READ OUT
	0	1	2	3	4	5	6	7				
16	0	0	0	0	0	0	1	1	64-67H	0	F=A	256-271V
17	0	0	0	0	0	0	1	1	68-71H	0	F=A	272-287V
18	0	0	0	0	0	0	0	0	72-75H	0	F=A+1	288-303V
19	0	0	0	1	1	0	0	0	76-79H	0 IF>15	F=A+1 OR F=A	304-319V
20	0	0	0	0	0	0	0	0	80-83H	0	F=A+1	320-335V
21	0	0	0	1	1	0	0	0	84-87H	0 IF>15	F=A+1 OR F=A	336-351V
22	0	0	0	0	0	0	0	0	88-91H	0	F=A+1	352-367V
23	0	0	0	1	1	0	0	0	92-95H	0 IF>15	F=A+1 OR F=A	368-383V
24	0	0	0	0	0	0	1	1	96-99H	0	F=A	384-399V
25	0	0	0	0	0	0	1	1	100-103H	0	F=A	400-415V
26	0	0	0	0	0	0	0	0	104-107H	0	F=A+1	416-431V
27	0	0	0	1	1	0	0	0	108-111H	0 IF>15	F=A+1 OR F=A	432-447V
28	0	0	0	0	0	0	0	0	112-115H	0	F=A+1	448-463V
29	0	0	0	1	1	0	0	0	116-119H	0 IF>15	F=A+1 OR F=A	464-479V
30	0	0	0	0	0	0	0	0	120-123H	0	F=A+1	480-495V
31	0	0	0	1	1	0	0	0	124-127H	0 IF>15	F=A+1 OR F=A	496-511V

Table 6
Truth Table For Mode 2

In Mode 2, the ALU performs the operations as directed by the program ROMs sequentially and the effect this has on the R CARs is as follows: The "old" information is recalled from memory K7 and processed by the ALU according to the ROM program step. When it is outputted by latch J8, it is not only rewritten back into the same place in memory K7 but also appears as FL signals which are used as data inputs for latches R12, P12 and M12 in the R CAR Generation and Control circuit. Since the clock for these latches is $\overline{\text{PHI 1}}$, the information is latched when $\overline{\text{PHI 1}}$ rises HI. FC3, FC5 and FC7 are connected to the output enables of the latches and the outputs are enabled only when these signals go LO. Please be aware that, although the FC signals are not overscored on the schematics, they are in fact active LO signals. The FC signals are only active between 64H and 128H and they are developed by R6 (in the Processor) which is a one-of-ten decoder addressed by the system address generator K6. When the address code reaches a certain BCD number, the appropriate output drops LO. For example, when the number 1 1 0 0 is at the address inputs, the third output is selected and FC3 drops LO. As this signal drops LO, the associated latch is enabled when $\overline{\text{PHI 1}}$ rises HI. When enabled, the latch releases the FL information which consists of the other FL signals exclusively ORed with FL8. This information appears at the selects and strobe of R14 and thereby controls the vertical motion of the cars.

MODE 3: The information for the score and time is updated during this mode which is enabled only when V1=0 and V2=1. Since the process for updating both the score and time is virtually identical, we will discuss only the processing of the more complex time information.

	ROM R7 OUTPUT								DATA READ IN	C_n	ALU FUNCTION	DATA READ OUT
	0	1	2	3	4	5	6	7				
0	0	0	0	0	0	0	1	1	64-67H	0	F=A	256-271V
1	0	0	0	0	0	1	0	0	68-71H	$\overline{\text{SCR 2}}$	F=A+1 OR F=A	272-287V
2	0	0	0	0	1	0	0	0	72-75H	0 IF 10 OR 15	F=A+1 OR F=A	288-303V
3	0	0	0	0	1	0	0	0	76-79H	0 IF 10 OR 15	F=A+1 OR F=A	304-319V
4	0	0	0	0	1	0	0	0	80-83H	0 IF 10 OR 15	F=A+1 OR F=A	320-335V
5	0	0	0	0	0	0	1	1	84-87H	0	F=A	336-351V
6	0	0	0	1	0	1	1	1	88-91H	$\overline{\text{TT}}$	F=A OR F=A-1	352-367V
7	0	0	0	1	1	1	1	1	92-95H	1 IF 10 OR 15	F=A OR F=A-1	368-383V
8	0	0	0	1	1	1	1	1	96-99H	1 IF 10 OR 15	F=A OR F=A-1	384-399V
9	0	0	0	1	1	1	1	1	100-103H	1 IF 10 OR 15	F=A OR F=A-1	400-415V
10	0	0	0	0	1	1	0	0	104-107H	$\overline{\text{SCR 1}}$	F=A+1 OR F=A	416-431V
11	0	0	0	0	1	0	0	0	108-111H	0 IF 10 OR 15	F=A+1 OR F=A	432-447V
12	0	0	0	0	1	0	0	0	112-115H	0 IF 10 OR 15	F=A+1 OR F=A	448-463V
13	0	0	0	0	1	0	0	0	116-119H	0 IF 10 OR 15	F=A+1 OR F=A	464-479V
14	0	0	0	0	0	0	1	1	120-123H	0	F=A	480-495V
15	0	0	0	0	0	0	1	1	124-127H	0	F=A	496-511V

Table 7
Truth Table For Mode 3

The time rate signal is known as $\overline{\text{TT}}$ and the development of this input is as follows: First, V GATE (the track speed signal from the analog circuitry) is ANDed with $\overline{\text{V RST}}$ and that result is ANDed with TIMGAT to form TT. TT appears as a narrow horizontal band at the bottom of the CRT with the video probe and its width varies with the positions of the SET TOP SPEED and TIMSET adjustment pots. The TIMGAT signal is adjusted by the TIMSET pot in Analog Timing and the length of the time unit can therefore be adjusted since TIMGAT is used in the development of TT. The track speed signal is necessary to compensate for the fact that, as the track moves faster, more points are accumulated. The track speed input therefore penalizes the player as scoring increases so that the point where extended play is awarded is made more difficult to reach.

After TT is generated, it enters Section 7 where it is NANDed at E13 with the signal from C13-15 which rises HI once every 16 frames. Therefore, TT is allowed through only once every 16 frames if the game is in the play mode and then only when one of the cars moves (in other words, $\overline{\text{TA1}}$ or $\overline{\text{TA2}}$ at E12 drops LO). The result of this process is known as $\overline{\text{TT}}$ and it must be kept in mind that $\overline{\text{TT}}$ is not always the inverse of TT.

$\overline{\text{TT}}$ then enters L6 is the Processor. This signal is normally HI and only drops LO when the timing pulse is active. Looking at the truth table for Mode 3, you can see that when the 6th instruction ($\overline{\text{TT}}$) comes up, the ALU performs F=A if $\overline{\text{TT}}$ is HI. But when the time decrement pulse occurs, the ALU function is F=A-1 which causes the time to be decremented by one.

The full decrementation process is a bit more complicated, but let us say that the machine has started from 99 and is decremented by the above process from 99 to 90. During this process, the number 9 in the tens location of the memory is not decremented (F=A) while the number in the ones location has the function F=A-1 performed on it until the number 90 is reached. At this point, both numbers in both memory locations have the function F=A-1 performed on them and the number displayed is then 89.

MODE 4: Mode 4 is enabled only when both V1 and V2 are HI and is more or less the same process as Mode 2. The main difference is that when the FC signals are LO, the ALU performs F=A+B+1. In Modes 2 and 3, the ALU performed functions on the A inputs only. However, in Mode 4, it operates on both the A and B inputs. This takes the outputs from K7 and L7, adds them together and then adds one to the result. The updated information is then written back into K7 but not into L7 and this is necessary to provide someplace to store the information since L7 is continuously being updated. The result is that the horizontal velocity of the cars is noticeably randomized and the vertical motion is also randomized to a certain extent.

89. LP: HI & PULSING.
SC: See H SWITCH timing diagram.

90. LP: LO & PULSING.
SC: See PHI 2 timing diagram.

91. LP: LO & PULSING.
SC: See PHI 1 timing diagram.

92. LP: HI & PULSING.
VP: See TP44.

93. LP: LO & PULSING.
SC: See DECTIM timing diagram.
94. See TP44.

95. See Table 4.
96. See Table 4.
97. See TP64.
98. See TP65.

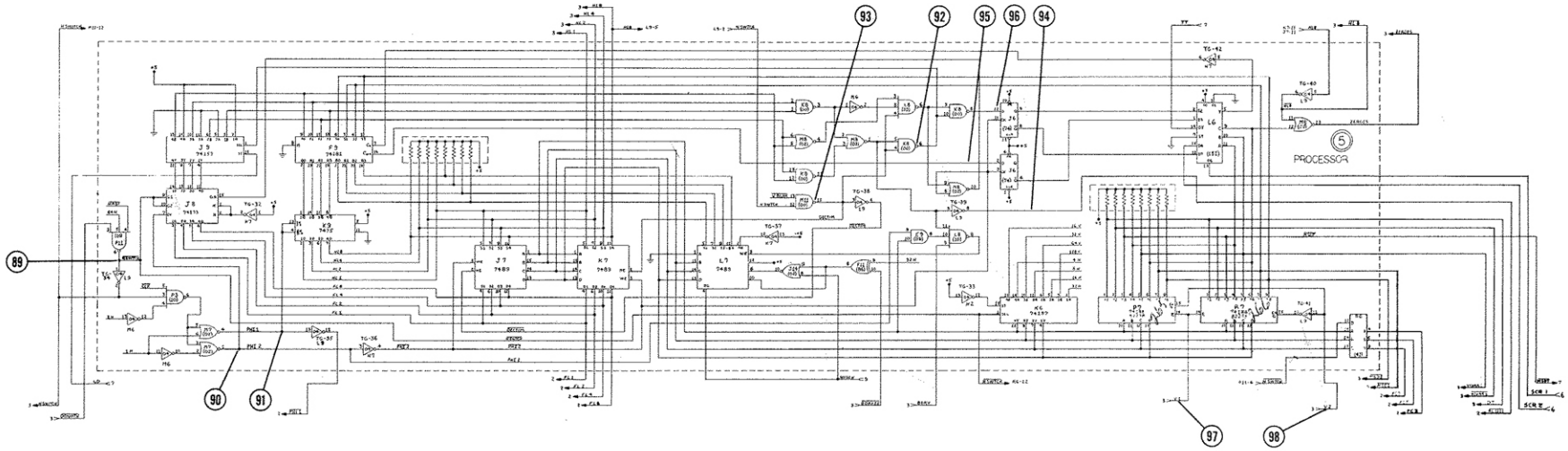


Figure 25
The Processor Schematic

	ROM R7 OUTPUT 0 1 2 3 4 5 6 7	DATA READ IN	C_n	ALU FUNCTION	DATA READ OUT
16	0 0 0 0 0 0 1 1	64-67H	0	F=A	256-271V
17	0 0 0 0 0 0 1 1	68-71H	0	F=A	272-287V
18	0 0 0 0 0 0 0 1	72-75H	0	F=A	288-303V
19	0 0 0 1 1 0 0 0	76-79H	0 IF>15	F=A+1 OR F=A	304-319V
20	0 0 0 0 0 0 0 1	80-83H	0	F=A+B+1	320-335V
21	0 0 0 1 1 0 0 0	84-87H	0 IF>15	F=A+1 OR F=A	336-351V
22	0 0 0 0 0 0 0 1	88-91H	0	F=A+B+1	352-367V
23	0 0 0 1 1 0 0 0	92-95H	0 IF>15	F=A+1 OR F=A	368-383V
24	0 0 0 0 0 0 1 1	96-99H	0	F=A	348-399V
25	0 0 0 0 0 0 1 1	100-103H	0	F=A	400-415V
26	0 0 0 0 0 0 0 1	104-107H	0	F=A+B+1	416-431V
27	0 0 0 1 1 0 0 0	108-111H	0 IF>15	F=A+1 OR F=A	432-447V
28	0 0 0 0 0 0 0 1	112-115H	0	F=A+B+1	448-463V
29	0 0 0 1 1 0 0 0	116-119H	0 IF>15	F=A+1 OR F=A	464-479V
30	0 0 0 0 0 0 0 1	120-123H	0	F=A+B+1	480-495V
31	0 0 0 1 1 0 0 0	124-128H	0 IF>15	F=A+1 OR F=A	496-511V

Table 8
Truth Table For Mode 4

C CAR #2 GENERATION & CONTROL

Two separate circuits develop the control car images (C CARs) and determine the vertical and horizontal motion of each player's car. The circuits are identical except that the C CAR #2 has additional memory area which is used for both circuits. The control for vertical motion is the player's steering pot, however horizontal motion is the result of acceleration and is controlled via analog signal H GAT 1. This last signal produces a spurt of leftward motion when the accelerator is depressed and a return to the right when the car deccelerates.

Two ROMs store information which is used both in these circuits as well as in other areas of the computer. The ROMs are addressed by a series of sync counter outputs and enabled by 128V or $\overline{128V}$ which enables ROM 258 (K5) for the upper player and ROM 259 (L5) for the lower player. This information is multiplexed at J5 to enable only the first player's information during the single player mode. When in the one player mode, 2 SCRN is LO which selects the information from K5 at the multiplexer. Although STRIPE is stored in this circuit, it is sent directly to Video Summation where it is used in the formation of the dotted line. BDRY 1 is another miscellaneous signal stored in this circuit and it is used only in the Processor.

The CAR TRG and CAR RST signals from the ROMs operate the trigger (pin 2) and the reset (pin 4) inputs of the 555 timer which is the actual vertical motion controlling device. When in the single player mode, $\overline{\text{CAR 1 TRG}}$ occurs just above the top stripe and is a thin, dark horizontal line. $\overline{\text{CAR 1 RST}}$ is similar but occurs 32 lines above the bottom stripe. In the two player mode, $\overline{\text{CAR 1 TRG}}$ still occurs above the top stripe, however $\overline{\text{CAR 1 RST}}$ occurs 32 lines above 128V. $\overline{\text{CAR 2 TRG}}$ and $\overline{\text{CAR 2 RST}}$ behave similarly, except for the second player's car when the machine is in the two player mode. The signals determine the boundaries of the area or time in which which the 555 can cause generation of the car image.

The vertical position of the car is controlled by a 25K steering pot which is connected both to a capacitor and to the threshold input of the timer. Turning the pot varies the capacitor charge time which in turn varies the pulse width from the timer output. As you turn the steering pot all the way to the left, the output drops LO almost immediately after $\overline{\text{CAR 2 TRG}}$ goes HI. As the pot is turned to the right, the capacitor takes a longer amount of time to charge and the output remains HI until the threshold is reached. Varying charge time causes the output pulse to be generated sooner or later with respect to sync, which in turn causes the image to appear further up or down the CRT by enabling P3 sooner or later.

Normally, the timer output is HI with LO pulsing which is inverted at M5 and the resulting signal clears flip-flop R3 and its $\overline{Q}$ output goes HI. This HI is connected back to P3 and when the game is set to the two player mode, PLYR 2 goes HI and P3 "waits" for a HI from M5-8. When it receives this HI, P3 is fully enabled and its output drops LO.

Whenever P3-8 is HI, P2 is cleared and reset to zero. Only when P3-8 has dropped LO does P2 begin counting H RST pulses or lines. A total of 16 lines are counted before the Q_D output toggles the first R3 flip-flop. After another 16 counts, the first R3 toggles again which in turn toggles the second half of this flip-flop. The net result is a LO from R3-6 which returns back to P3 and resets the counter. The counter outputs are all connected to exclusive OR gates where the count can be selectively inverted by the signal from R3-3. If R3-3 is LO, the outputs of the exclusive ORs count from 0 0 0 0 to 1 1 1 1, however if R3-3 is HI, the count is reversed so it counts down from 1 1 1 1 to 0 0 0 0. The three MSBs from R2 are later used to select and strobe out information from R1 so that the first half of the image is mirrored. But before we delve into this aspect, we must first discuss how the information is read out of the ROMs before it is multiplexed by this mirrored count.

The address circuitry begins with the generation of H GATE at L4 which establishes the horizontal limits of the area in which the car may appear. This signal is ANDed with H GAT 2, an analog-produced signal, which controls the actual horizontal position of the cars via a window which is adjustable by a trim pot. Adjusting this pot determines the normal position of the car on the CRT horizontally. These two signals form a composite position signal at M4-6. Gate M3 is looking for two LO inputs, one of which comes from M4-6 and the other from the counter's carry-out (CO) line.

The counter and flip-flop K3 are both cleared by $\overline{\text{H TRG}}$ which produces a LO carry pulse and, when M4-6 drops LO, a HI results from M3-1 and this HI is clocked through by CLK thereby counting up M1. After 32 counts, the carry-output again returns HI and M3 is disabled which in turn disables K3. The result at the counter outputs is simply a progression from 0 0 0 0 to 1 1 1 1 which is used along with the signal from R3-6 to address ROM P1 at the correct time and retrieve information used in the generation of the horizontal aspects of the car. This horizontal information is then selected by vertically moving signals at the select inputs of R1 which determines the vertical limits for the car image and imparts vertical motion. The strobe input at R1-7 is a long window which allows only one car image to appear and be displayed. The resulting single image is inverted at M5 and gated with $\overline{\text{FLASH 2}}$ which causes the car image to be turned on and off rapidly to indicate a crash has occurred. The final control car signal is then sent to Section 4 (Control Car #1) where it is gated with the first car signal before entering Video Summation.

99. LP: Pulses LO when C CAR #2 hits an R CAR or C CAR #1.
100. LP: Pulses LO when C CAR #1 hits an R CAR or C CAR #2.
101. See TP51.
102. See TP80.
103. See TP79.
104. LP: LO & PULSING.
VP: Light stripes over dotted lines.
105. See TP67.
106. Similar to TP81 except for 2nd player's car.
107. See TP84.
108. See TP83.
109. See TP85.
110. LP: HI & PULSING.
VP: Dark bar at top of screen which changes width according to setting of SET TOP SPEED pot and ACCEL pot for 2nd player.
111. Same as TP110 except for player #1.
112. LP: LO & PULSING.
VP: Light bar at top of screen which changes width according to setting of SET TOP SPEED.
113. LP: LO & PULSING.
VP: Dark bar at top of screen which changes width according to setting of SET TOP SPEED pot and 1st player ACCELERATOR pot.
114. LP: Same as TP113 except for 2nd player.
115. LP: LO & PULSING.
VP: Light bar at top of screen which changes width according to SET TOP SPEED & TIMSET pots.

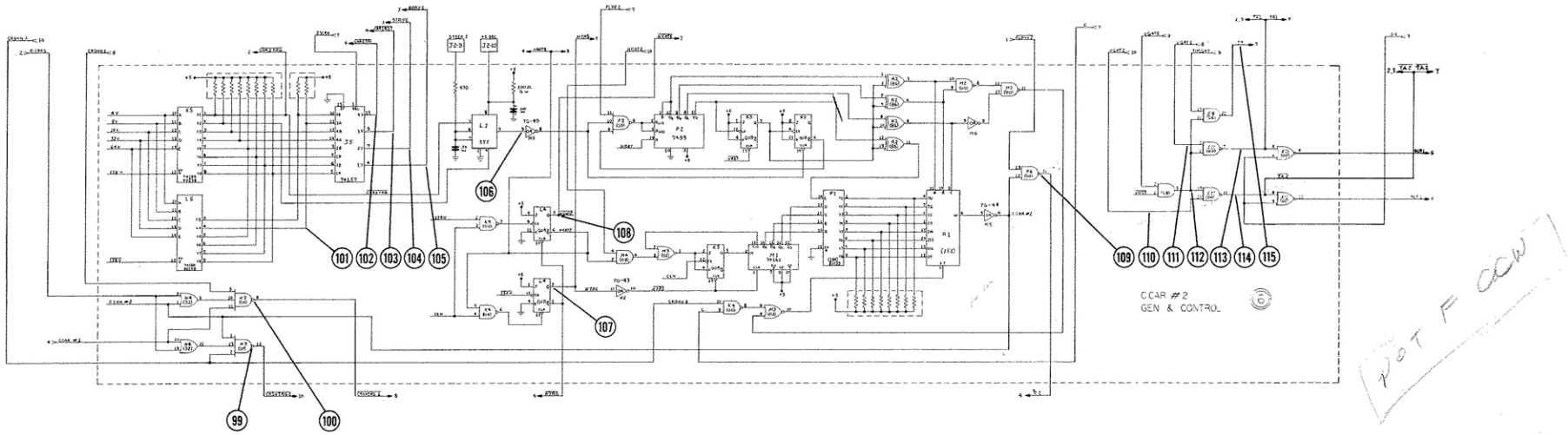


Figure 26
C Car #2 Generation & Control Schematic

GAME CONTROL

Like any other credit circuit, Game Control performs a number of loosely related functions governing the starting of the game and how many players are in the game. While some of this credit circuit is quite unusual, the coin, start and anti-static portions follow the general lines found in almost every video game.

THE COIN CIRCUIT: When the coin is deposited, the R-S flip-flop constructed from both halves of R5 is toggled and $\overline{\text{COIN}}$ drops LO clocking latch B9. Latch B9 performs the coin accumulator function necessary when a game is designed to accept several coins per play. Since the serial input (pin 9) of B9 is tied HI, each time it is clocked by the deposit of a coin, the HI at the output is advanced by one stage. Before any coins are deposited, the latch outputs are at 0 0 0 0 0 and the first coin produces an output configuration of 1 0 0 0 0. The second coin deposited clocks the latch again to produce 1 1 0 0 0 and so on until the fifth coin fills up the entire latch and it outputs 1 1 1 1 1. The number of coins required per game is fully adjustable according to where ACTIVE and 2 PLYR are connected. For example, if both these inputs are jumpered to pin 15, only one coin is required whereas if they are connected to pin 14, two coins will have to be deposited before the game will be enabled. The clear input (pin 16) is active only when the game ends at which time $\overline{\text{PLYOVR}}$ resets the latch outputs back to 0 0 0 0 0.

THE CREDIT CIRCUIT: Both halves of B11 are known as the credit flip-flops. The credit circuit is said to be "armed" when $\overline{\text{COIN}}$ presets these flip-flops and produces a HI $\overline{\text{PLYOVR}}$. Notice that this HI also enables latch B9 so that it can count additional coins if the accumulator is set up to do so. $\overline{\text{COIN}}$ also clears the R-S flip-flop composed of both halves of P6 if a static charge has occurred and this allows the game to be enabled if a prior static discharge has cleared credit from the credit flip-flops.

Please note that there is another jumper adjustment to be made here. If one play per coin is desired, connect B11-4 to +5 volts. However, if two plays per coin is desired, connect this line to $\overline{\text{COIN}}$. The way this works is fairly simple. If both preset inputs of the flip-flops are connected to $\overline{\text{COIN}}$, the deposit of the coin presets the outputs of both flip-flops HI. At the end of the first game, G.O. clocks both flip-flops simultaneously and the HI at the Q iutput of the first flip-flop is clocked through the second and $\overline{\text{PLYOVR}}$ remains HI enabling the game. However, if the preset of the first flip-flop is tied HI, its Q output will not be preset HI and when the first game ends, the LO at this flip-flop will be clocked through the other flip-flop and game credit will be removed.

THE STATIC CIRCUIT: The function of the static system is to prevent a player from obtaining free games by inducing static discharges in the credit circuit. Without the static system, a large enough discharge could induce transients into the circuit which would be interpreted as a coin deposit. The inclusion of the static system prevents credit from being registered in the credit flip-flops even though transients may get into the credit circuit.

If a large enough static discharge is released, a HI is induced in the antenna, the 2N3643 conducts and brings down the base of the 2N3644 which places a logic LO at P6-5. This LO produces a HI from P6-6 which is inverted by M5 and clears both the credit flip-flops and removes any game credit. But since $\overline{\text{COIN}}$ is connected to P6-9, the deposit of a coin will clear the R-S antenna flip-flop after a static discharge has occurred and allow the game to be started.

THE START CIRCUIT: Because of an inadvertent engineering oversight, the start circuit of this game does not function properly. Even though provisions for a start switch have been included, the game will always start when the coin is deposited and operating the start switch will have no significant effect on the circuit. This situation may be rectified by installing a modification kit available from your Midway distributor.

When the coin is depsoited, the B11 flip-flops are preset and $\overline{2\ \text{GHS}}$ drops LO. Since the game has not yet started, $\overline{\text{G.O.}}$ is LO, placing LOs at pins 9 and 10 of A11. Since there is normally a HI at A11-11, the output at A11-8 is LO. This LO is connected to A11-1 and since COIN goes HI when the coin is deposited, A11-12 will go LO. This LO triggers one-shot A12 producing a LO pulse from the $\overline{\text{Q}}$ output which presets the game over flip-flop and $\overline{\text{G.O.}}$ goes HI. Therefore, the game will start before the start switch is ever pressed.

At the same time the $\overline{\text{Q}}$ output of A12 drops LO, A12-13 rises HI. On the falling edge of this pulse, the second half of A12 is triggered and this signal is called LD. Although the schematic indicates this signal in an input to Section 4, it actually enters the Processor (Section 5) where it resets all the information back to zero at the beginning of the game. When the second half of A12 is triggered, $\overline{\text{LDVL}}$ is generated which simply pulses LO for 32.5 ms when the game starts.

The game over signal which has gone HI and the LO pulse from $\overline{\text{LDVL}}$ are ANDed at D9 which produces a LO pulse at D9-11 clearing the FSTCLK and extended play flip-flops.

$\overline{\text{FSTCLK}}$ is an input to Analog Timing and its generation begins with the NORing of $\overline{\text{MSDT}}$ and $\overline{\text{D8}}$ at pins 3, 4 and 5 of A11. $\overline{\text{MSDT}}$ goes LO only when the third or hundreds digit of the time is being generated, but actually this digit is never visible. When the game starts, all zeroes are loaded into the score and time memory and on the next instruction cycle, the Processor steps the number 999 down during the game until 899. The 99 part of 899 is also never seen because as soon as it gets down to 899, the game ends and only zeroes are allowed out of the score display ROMs (see the Score Display analysis for more information on this aspect).

Signal $\overline{\text{D8}}$ drops LO when the time number 899 is reached. This produces a HI from A11-6. H TRG clocks this HI through flip-flop B10 and produces a LO at the $\overline{\text{Q}}$ output resulting in $\overline{\text{FSTCLK}}$.

EXTENDED PLAY: Both halves of C10 are used for the player's extended play flip-flops. If $\overline{\text{XPI}}$ is LO (e.g. the first player has exceeded the score required for extended play), this signal will clock a HI from C10-3 (XPLY 1) which will cause C9-12 to drop LO. This LO combined with the LO at $\overline{\text{FSTCLK}}$ results in a HI at B12-13 which is inverted at F10. This LO then selects the 2A input of multiplexer A10 and signal $\overline{\text{D7}}$ appears at A10-7 as the overtime signal which controls when the game ends.

The overtime signal $\overline{\text{OT}}$ is combined with $\overline{\text{MSDT}}$ at B13 and when both these signals are LO (during extended play and when the most significant digit of the time is being generated), a LO results from B13-3 which produces a LO from C1203. This LO is clocked through the game over flip-flop D11 by H gate and ends the game.

116. LP: Drops LO when coin switch is closed & stays LO until the switch opens again.
117. LP: Goes LO only if static discharge ended game; is reset HI by deposit of another coin.
118. LP: Goes HI when coin is deposited and stays HI until one or two games are over (depending on adjustment of B11-4).
119. LP: Drops LO when game starts and rises HI when game ends. SC: Pulses for 32.5ms.
120. LP: HI pulse with coin deposit. SC: Pulses LO for 32.5ms, 32.5 ms after the coin is deposited.
121. LP: LO pulse with coin deposit. SC: Pulses LO for 32.5ms, 32.5 ms after the coin is deposited.
122. LP: Very slowly pulsing signal.
123. LP: Drops LO when game ends. Rises HI at start of game when either player moves his car.
124. LP: Drops LO when game enters extended play.
125. LP: Pulses LO if a coin is deposited or if a crash occurs.
126. See TP22.
127. See TP27.

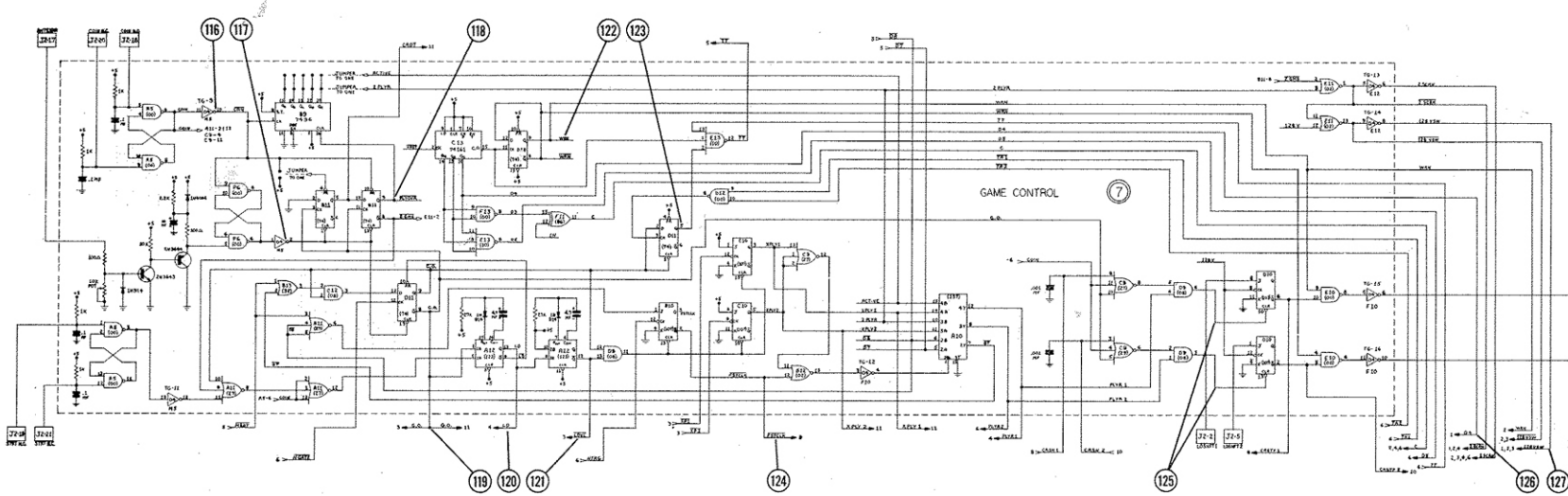


Figure 27
Game Control Schematic

However, let us assume the game has been started and G.O. is LO. Since CRSH 1 and CRSH 2 are normally LO, there are HIs at C9-6 and C9-8. If PLYR 1 and PLYR 2 are HI (cars active), D9-3 and D9-6 are HI enabling flip-flops D10. If both player's gearshifts are in low gear, CRSTP 1 and CRSTP 2 go LO and the cars begin moving. If one of the cars crashes, the output of C9-8 or C9-6 goes LO and produces a LO from D9-3 or D9-6 which clears one or the other of the D10 flip-flops causing the $\overline{Q}$ output to go HI. This enables E10 and allows WRN through. WRN is simply a very slowly pulsing signal produced by dividing $\overline{V\ RST}$ 32 times at C13 and D13. Since WRN and $\overline{WRN}$ are both entered at E10, the warning lights will flash alternately if both cars crash simultaneously.

You can see from Table 1 in the Add Cars section that signal $\overline{D7}$ goes LO at a score of 799. Therefore, the player is granted an extra 100 time units of play. However, since $\overline{FSTCLK}$ is LO at this time, the Analog Timing circuit causes the time to be decremented more quickly for these extra 100 time units.

CAR STOP: The rest of the circuitry to the right of the extended play multiplexer A10 is used to develop CRSTP 1 and CRSTP 2, signals used in the analog circuitry. Note that this circuit is only active when the game is in progress. When G.O. is HI (the game over), LOs result from C9, producing LOs at the outputs of D9 which clears both of the car stop flip-flops and holds them in the inactive states.

ANALOG CHANNEL 1

A number of functions related to the acceleration, shifting and crashing of the first player's car are controlled in this circuit. For example, when a crash occurs the car image is reset, the engine sound turned off and the crash sound enabled through. Other functions include establishing the relative position of the car in the CRT and regulating the speed of the course. But before discussing how these functions are actually operated, a bit of component description is in order.

ANALOG SWITCHES: A number of analog switches are used in this circuit to control non-digital signals. These switches pass the input signal (I) through the switch when the control line (C) is HI and this signal then appears at the output (O).

OPERATIONAL AMPLIFIERS: The LM3900 opamps are current-operated devices with a current-mirror architecture. When the (-) or inverting input has a greater amount of current flowing into it than the (+) or non-inverting input, the output of the device rises to a logic HI level. The ground of the device is offset by -0.7 volts so the output can swing to ground.

COMPARATORS: The LM339s operate essentially like the amplifiers, except their outputs are set up in such a way that they switch extremely quickly. Also, these devices are essentially voltage-operated. When the (-) input is at a higher voltage than the (+) input, the device output drops LO. Conversely, a HI output results when the (+) voltage rises above that of the (-) level.

THE ACCELERATOR CIRCUIT: This circuit stores the acceleration information in the form of a charge on a capacitor which becomes available to a number of other circuits which control various acceleration related functions. The amount of acceleration is manifested by the amount of charge in the cap and displayed by moving the car image to the left.

When the game is started or after a crash has occurred, the player's car is displayed in a position slightly to the right of the center of the CRT. As the accelerator is depressed, the car is moved leftward an amount proportional to the position of the accelerator pot and this continues until top speed in low gear is reached. Shifting into high gear produces greater top speed and the velocity of the course is increased, however there is less acceleration capability in this gear.

The player's accelerator pot is wired between +5 and -0.7 volts and the wiper is connected to the accelerator circuit through pin 4 of logic jack #2 (J2-4). The potential established by the setting of the pot charges the 10MF capacitor and the voltage at this node is limited by the 560K resistor and then entered at the non-inverting input of Opamp A. Also connected to this node are a 27K resistor and a 1N914 diode wired across a 100K resistor. When the potential at edge connector pin J2-4 is lower than the positive plate of the capacitor, the diode is forward biased. This causes the capacitor to be discharged faster than it was charged with the result that the car is relatively difficult to accelerate but brakes quickly. The feature has been specifically engineered to simulate the way a real car handles in that it is easier to brake a car than it is to accelerate it.

Many of the other sub-circuits in Analog Channel 1 control the accelerator circuit to provide a more realistic acceleration sequence when shifting gears, braking, crashing, etc. Each of the following analog switches performs one of these functions.

SWITCH 1: When a crash occurs, CRSTP 1 rises HI and closes Switch 1 discharging the 10MF capacitor to ground through a 270Ω resistor. This stops the car from accelerating and resets it back to the right side of the CRT.

SWITCH 2: This switch is essentially controlled by HISHFT 1 and provides a more realistic acceleration sequence as the car is shifted. In a real car, low gear will produce high acceleration but low top speed while high gear will result in low acceleration but a high top end. Switch 2 is used to divert some of the current reaching Opamp A when in high gear. This produces a lesser amount of acceleration for a given change in the accelerator pedal position.

SWITCH 3: If a crash occurs or when the car is shifted into high gear, this switch shorts out the 1MF capacitor through the 10K resistor and abruptly discharges it.

128. SC: Changes with setting of accelerator pot from ≈2v to ≈3v.
129. SC: Changes with setting of accelerator pot from ≈1v to ≈2v. Drops LO when CRSTP rises HI.
130. SC: Changes with setting of accelerator pot from ≈15v to ≈2v. Responds more quickly when car is in low gear.
131. SC: Pulse occurring every 63.5 µs. Width dependent on setting of accelerator pot.
132. SC: Pulse occurring every 16.6 µs. Width dependent on setting of accelerator pot.
133. LP: Goes HI when accelerator is set beyond a certain point.
134. SC: Pulses LO when car crashes provided TP133 is HI.
135. SC: Goes HI if car crashes or when player shifts into high.
136. SC: Goes HI when accelerator is set beyond a certain point.
137. SC: Goes HI for 0.62 sec. after a crash occurs.
138. SC: Randomly occurring pulses during the time TP137 is HI.
139. SC: Erratic sawtooth.
140. SC: Randomly occurring pulses.
141. SC: Filtered engine RPM noise waveform.
142. SC: TP141 added to TP138 and amplified.

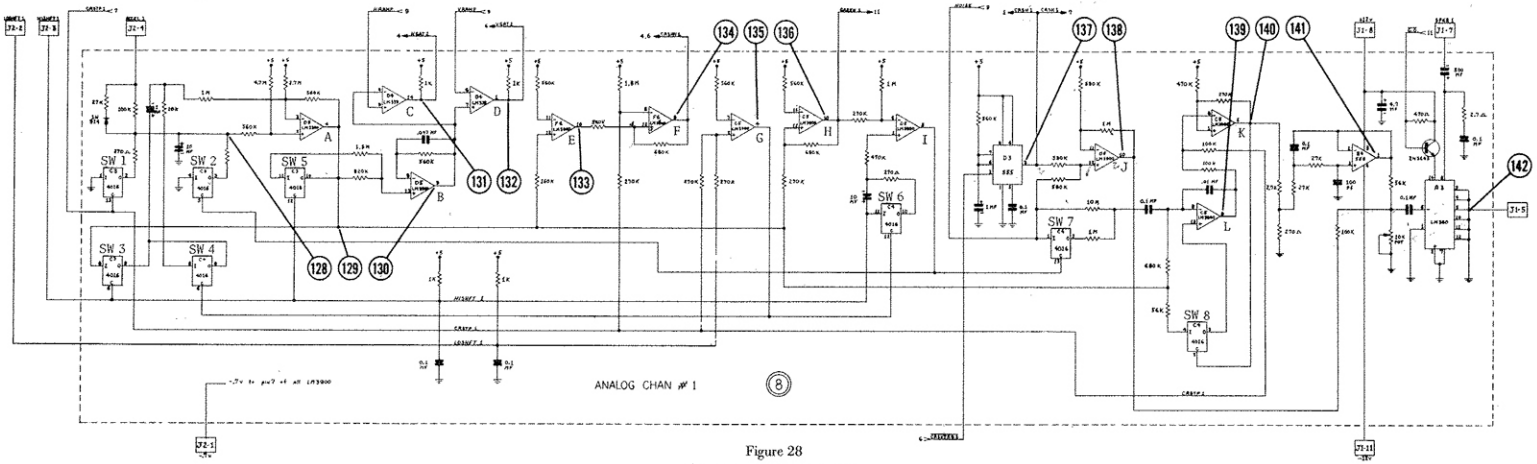


Figure 28
Analog Channel 1 Schematic

SWITCH 4: Switches 4 and 5 are both directly controlled by HISHFT 1. When in low gear, Switch 4 causes Opamp A to act like an integrator which charges the 1MF capacitor in a linear fashion to provide a time lag between when the setting of the accelerator pot is changed and when the car responds. The result is smoother and more realistic acceleration.

When in low gear, HISHFT 1 is HI and the switch closes so that the 1.5M and the 820K resistors are wired in parallel through the analog switch. This permits the integrator to charge the capacitor more quickly when in low gear producing quicker acceleration in this gear. When the player shifts into high, HISHFT 1 drops LO and cuts the capacitor out of the circuit altogether.

AMPLIFIER A: As mentioned before, this amplifier acts either as an opamp or an integrator depending on how the switches are set and its output is varied according to the setting of the accelerator pot and the conditions created by the analog switches. The output from Opamp A feeds several other circuits including integrator B.

INTEGRATOR B: The acceleration-related voltage from Opamp A enters the (+) input of B. Since B is wired up as an integrator, the .047MF capacitor is charged through B to the potential at the (-) input. A certain amount of time is required to charge the .047MF and this is controlled by Switch 5 which we will get to in a minute. In any case, the voltage from the capacitor enters the (-) input of B and produces a linear or smoother output which operates the (-) input of C and the (+) input of D.

COMPARATOR C: The output of B varies according to the setting of the accelerator pot and the instantaneous voltage of any point along the charge curve of the capacitor is entered at the (-) input of C. Therefore, the sawtooth waveform of H RAMP must rise to the value of the instantaneous voltage before the output of C will go HI. In other words, H GAT 1 will go HI according to how the pot is set and when the H RAMP waveform reaches an equal value. This analog-to-digital interface is necessary so these waveforms can control digital devices.

COMPARATOR D: The operation of this device is quite similar to that of C except V RAMP must fall to the value of the instantaneous volatge before V GAT 1 will go HI.

AMPLIFIER E: The (-) input is tied HI through a 560K and the (+) input is controlled by the output of accelerator amplifier A. The 560K connected to the (-) input allows about 8.9µA through this input so the same amount of current must flow into the (+) input before the output will go HI. Approximately 1.35 volts are required from Opamp A to produce a HI from amplifier E.

AMPLIFIER F: When the player has accelerated, the output from Opamp A reaches the value required to produce a HI from E which controls the (+) input of F. Since there is a 560K resistor in the line between the output of E and the (+) input of F, there is about 8.9µA entering the (+) input of F. Since the output of F is LO at this point, the 680K pulls the (+) input down by about 7.35µA so the net current available is 1.55µA.

Before a crash occurs, CRSTP 1 is LO which pulls the (-) input LO. However, when a crash does occur, CRSTP 1 rises HI and since there is already 2.8µA flowing through the 1.8M resistor, the HI from CRSTP 1 increases the total current available at the (-) input to 20µA. Therefore, the crash sequence is disabled unless the car is moving.

AMPLIFIER G: G is tied HI through a 560K which establishes a voltage at at (-) input. CRSTP 1 and LOSHFT 1 are connected to the (+) through two 270K resistors. CRSTP 1 rises HI when the car crashes and LOSHFT 1 is HI only when the car is in high gear. Wired in this way, these two inputs act like an OR gate connected to the (+) input. When either or both of the signals go HI (when the car has crashed and/or while in high gear), the output also goes HI. If both are LO, the output is also LO. This output is then used as the control input for Switches 3 and 6.

AMPLIFIER H: The (-) input of this opamp is also tied HI through a 560K and the (+) input is controlled by the output from Opamp A. Since Opamp H is wired with positive feedback like F, the gain is increased and a much smaller signal can trip it. So, when acceleration has reached the predetermined point, the output of H goes HI. Therefore, GREEN 1 only rises HI after the accelerator has been depressed a certain amount and then only if conditions at the switches which affect Opamp A are right.

AMPLIFIER I: When HISHFT 1 is HI (car in low gear), the 10MF capacitor is charged through the 470K resistor which produces a changing voltage at the (+) input as the capacitor is charging. The peak charge of the capacitor is reached 4.7 seconds later and the output of Opamp I goes HI. But, since Switch 6 is controlled by the output of Opamp G, if the player shifts into high gear or a crash occurs, the capacitor is abruptly discharged by the 270Ω resistor.

In other words, when the player shifts into low gear and accelerates, 4.7 seconds of acceleration are required to reach the low gear maximum RPM plateau after which point the car will no longer accelerate. However, if the player shifts gears or a crash occurs, the acceleration position of the car is reset back to zero.

THE NOISE CIRCUITRY: The remainder of the circuitry in this section (the area to the right of and including the 555 timer) generates noises to accompany the control functions produced by the other circuitry.

THE TIMER: The output, CRSH 1, is the signal which enables the crash sequence, the duration of which is controlled by the 555 timer. CRSH TRG 2 is connected to the trigger input of the timer and this signal drops LO whenever a crash occurs producing a pulse from the timer. Since the timer is connected in the monostable mode, the output rises HI when active and the values chosen for the external RC network limit the pulse duration to 0.62 seconds.

AMPLIFIER J: Opamp J turns on the crash sound during the crash sequence. When a crash occurs, the timer outputs a HI pulse which balances the currents at the (-) and (+) inputs and NOISE is allowed through and amplified by Opamp J. NOISE is the random white noise output of Analog Timing and its unrefined waveform is used to directly generate the raucous crash sound. When enabled, NOISE is sent directly to the audio amplifier without further processing.

Switch 7 is controlled by Opamp I which rises HI only when the RPM plateau has been reached. Even when it is LO, some NOISE is allowed into Opamp I through the 10M resistor, but when the control line to Switch 7 goes HI, eleven times more NOISE is allowed through Switch 7 and is AC-coupled to the (-) input of L. The result is greater noise amplitude at maximum RPM.

Opamps K and L also transform the unrefined NOISE waveform into timed bursts of engine noise where the burst rate is relative to acceleration or, in other words, to engine RPM.

The NOISE signal at the (-) input of L discharges the .01MF capacitor through L and, when sufficiently discharged, the output of K rises HI. This HI closes Switch 8 which pumps a charge into the (+) input of L, recharging the capacitor and producing a burst of noise. The recharge time is controlled by the output from accelerator Opamp A so that the time between bursts is relative to acceleration and the engine RPM then increases as the car accelerates.

The engine sound from Opamp K is filtered by the 558 and associated components to produce a cleaner sound. The filtered waveform enters the LM380 integrated audio amplifier along with the crash sound. The amplitude of the volume is controlled by the 10K volume pot which pulls off an amount of sound voltage proportional to the setting of the pot.

The transistor connected to the power pin (pin 14) of the LM380 enables the crash and engine sounds only during the play mode. When the game ends, G.O. drops LO and pulls the base of the transistor down which turns off the transistor and shuts off power to the amplifier.

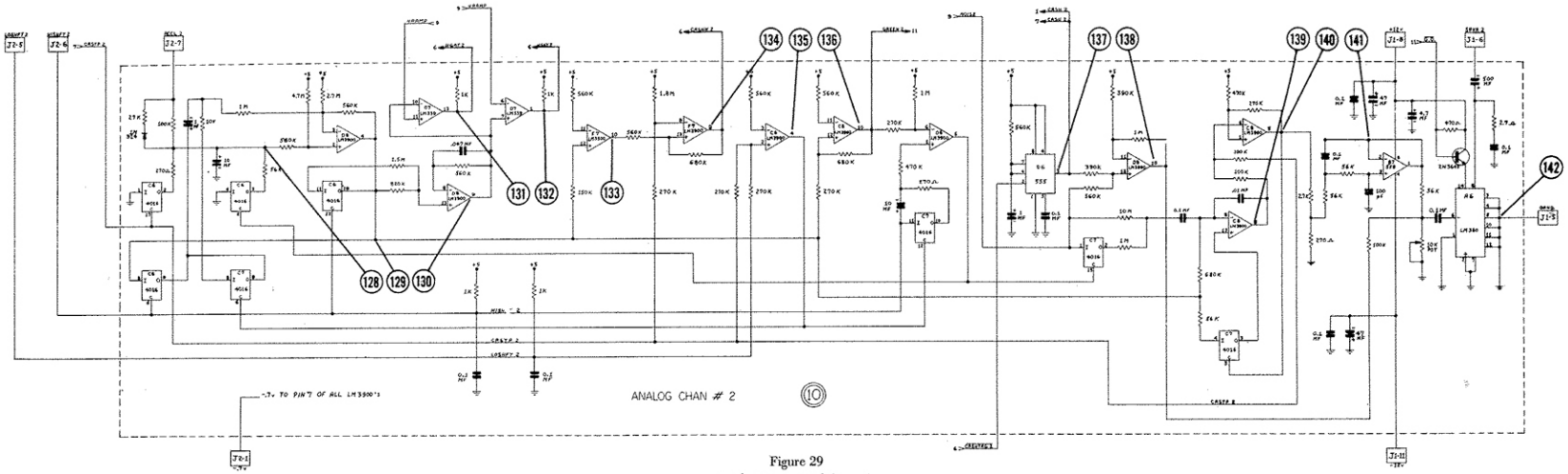


Figure 29
Analog Channel 2 Schematic

ANALOG CHANNEL 2

The second player's Analog Channel is absolutely identical to the first player's with the exception of two resistors. Both resistors are in the line to the (+) input of B7 (B7-3) and they simply provide a slightly different tone for the second player's engine sounds. In all other respects, the circuits are identical and require no further elaboration. See TP128 through TP142 when testing this circuit.

ANALOG TIMING

Analog Timing consists of four miscellaneous sub-circuits which are used to generate signals needed in other parts of the computer.

1. V GATE/V RAMP: The 555 timer is triggered with $\overline{\text{V RST}}$ which starts the 1MF capacitor charging. Since this capacitor is charged by a constant current source produced by the transistor, the ramp waveform is linear. The position of the SET TOP SPEED pot determines the amount of current through the diode which establishes the amount of current at the base of the transistor. When the capacitor has charged to 2/3 of Vcc, the output of the timer (pin 3) drops LO and it stays LO until the next $\overline{\text{V RST}}$ pulse. Therefore, as the pot is turned, the output stays LO for a longer or shorter period of time.

When viewed with the video probe, the signal appears as a white band at the top of the CRT (because $\overline{\text{V RST}}$ has reset the 555 output HI). The bottom edge of the band represents the point in time when the capacitor has charged to 2/3 Vcc. The video probe reveals that the waveform is LO until $\overline{\text{V RST}}$ occurs again since the area of the CRT underneath the white band is dark.

2. TIMGAT: $\overline{\text{FSTCLK}}$ is generated in the Game Control circuit (Section 7) by the following process: When the game starts, flip-flop B10 is cleared and stays cleared until $\overline{\text{MSDT}}$ and $\overline{\text{D8}}$ both go LO, causing A11-6 to go HI producing a HI at B10-1. On the next H TRG pulse, the HI is clocked through and the $\overline{\text{Q}}$ output drops LO to generate $\overline{\text{FSTCLK}}$. FSTCLK is simply a signal which drops LO at a predetermined time; the reason it has been named "fastclock" is that its function is to speed up the rate at which the time display is decremented.

$\overline{\text{FSTCLK}}$ is connected to the inverting input of the opamp. When this signal is HI, the opamp output will drop down an amount determined by the position of the adjustment pot. The gain of the opamp is set at approximately 2. The output of the first opamp runs through a 560K resistor to the non-inverting input of the second opamp. This input is also tied to the wiper of the TIMSET adjustment pot through a 560K resistor. As this signal moves down, the value that V RAMP must reach before the output of the second opamp goes LO also changes. Therefore, the result is a signal which is wider when $\overline{\text{FSTCLK}}$ is HI that when it is LO. Please be aware that, since TIMGAT is used in the generation of $\overline{\text{TA1}}$ and $\overline{\text{TA2}}$ which are in turn used in the Course Speed circuit, the adjustment of the TIMSET pot also affects the course velocity.

3. H RAMP: H RAMP is used to determine the normal horizontal position of the C CARs. The H RAMP control adjusts the position of the C CARs on the CRT so that the cars can be positioned in the desired area.

The two transistors form a simple current-mirror scheme where the setting of the pot controls the current through the first transistor and, assuming that both transistors have approximately the same β, an equal amount of current will appear in the other leg. This charges the .001 MF capacitor in a linear fashion and, when H GATE drops LO, the 2N3644 transistor is turned on and discharged the capacitor. The resulting H RAMP signal appears as a series of spikes because it is being charged and discharged at about the same rate.

4. NOISE: First, a very low current (18 microamps to be specific) is established through the transistor which limits the amount of current through the Zener to a correspondingly low amount. The Zener produces noise because, although an avalanch voltage can barely be reached, enough current is drawn off to halt the avalanch immediately after it begins. This continues in a repetitive fashion, generating random "white noise".

This noise signal is amplified by CD-9, and at this point, the amplified signal goes to two places for separate processing: (A) It is filtered by the 30K resistor and the two .047 MF capacitors before entering the inverting input of the other amplifier. Here it is further amplified and the resulting noise signal becomes available to other circuits of the computer. (B) The unfiltered noise waveform is amplified by the other transistor and inverted by P10 which tends to square up the waveform to make it more acceptable to other digital devices.

143. LP: LO & PULSING.
VP: Light bar at top of screen which changes with the setting of the SET TOP SPEED pot.
144. LP: HI & PULSING.
VP: Dark bar at top of screen which changes width according to SET TOP SPEED & TIMSET pots.
145. SC: Sawtooth waveform where the trailing edge is adjustable by C CAR position (H RAMP) pot.
146. SC: Random white noise.
147. SC: "Digitized" random noise.

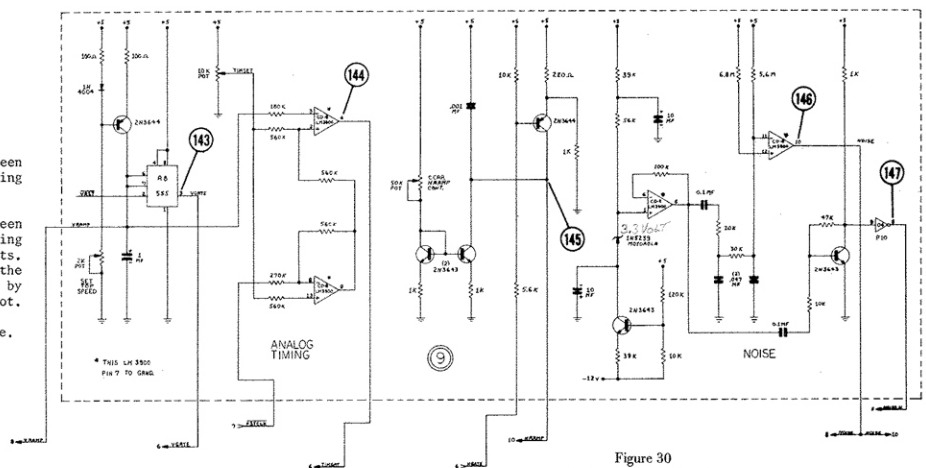


Figure 30
Analog Timing Schematic