

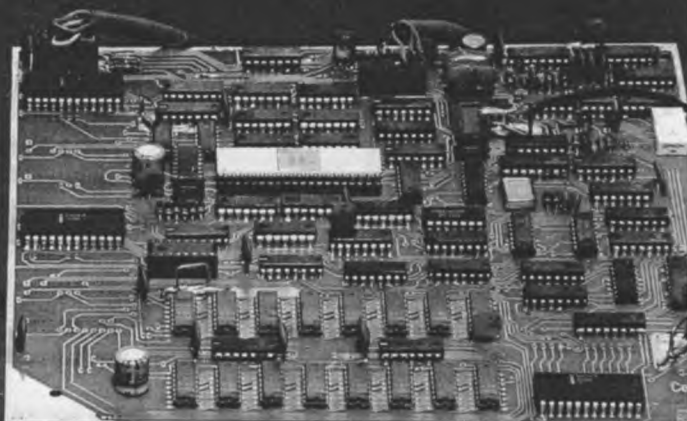
NEW DBX SYSTEM - PLL FOR CB

\$1.00 ■ APR. 1977

Radio-Electronics

THE MAGAZINE FOR NEW IDEAS IN ELECTRONICS

build this
2650 COMPUTER
on one board
experiment with
FUNCTION GENERATOR
new Exar IC
for your phone :
BUILD TELESWITCH
and turn things on
IC data sheet :
RETICON SAD-1024
bucket-brigade
installing
ANTENNA TOWERS
can be easy



SIGNETICS 2650 MICROPROCESSOR
ASCII ENCODED KEYBOARD
80-CHARACTER BY 16-LINE VIDEO DISPLAY
CASSETTE INTERFACE--KANSAS CITY STANDARD AT 300 BAUD
SUPERVISOR PROGRAM
1024 BYTES ROM EXPANDABLE TO 4K
2048 BYTES RAM
I/O PORTS EXPANDABLE TO 256
SINGLE 5-VOLT 3-AMP POWER SUPPLY
BROUGHT TO YOU EXCLUSIVELY BY
RADIO-ELECTRONICS MAGAZINE
AND JEFF ROLOFF

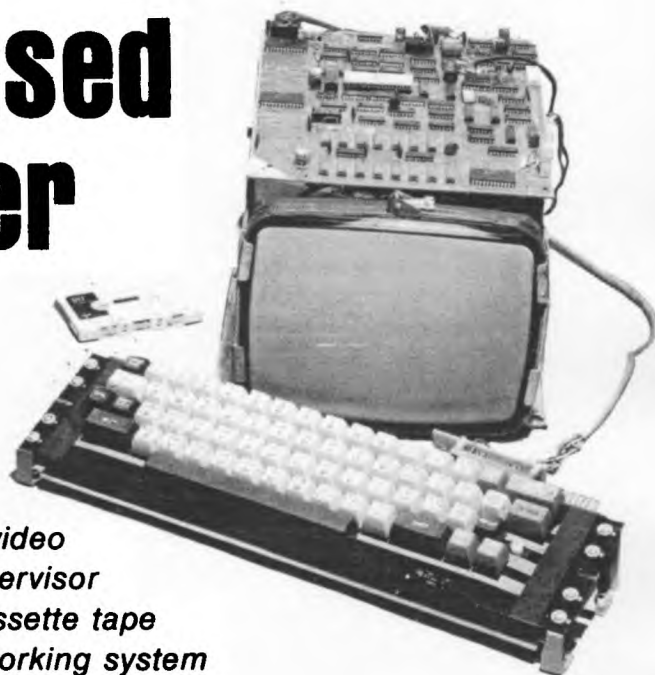


PLUS:

- ★ Komputer Korner
- ★ Jack Darr's Service Clinic
- ★ Lab-Tested Hi-Fi Reports
- Yamaha CT-800 Tuner & Kenwood KR7600 Receiver
- ★ State-Of-Solid-State ★ Equipment Reports ★ Reader Questions

GERNSBACK
PUBLICATION

Build 2650-Based Microcomputer System



Part 1. Built on a single printed-circuit board, this 2650 microcomputer contains a video and cassette tape interface and resident supervisor program. Add a keyboard, video monitor, cassette tape recorder and power supply for a complete working system

JEFF ROLOFF

AN EASY-TO-USE COMPUTER SYSTEM MUST have essentially four major sections. There is, of course, the actual processing section that usually (at least for small users) contains a microprocessor IC. There must also be input and output sections so that the processor can communicate with the outside world. The last section that is essential for a computer's operation is the memory. This is used to store programs and data, and it can be either a memory that needs to be accessed very quickly when running a program or a memory used to store information for long periods of time, such as cassette tape.

The 2650 computer project presented here has all of the above circuits mounted on a single printed-circuit board. In addition, there is a versatile supervisor program that allows you to easily create your own programs. The heart of the unit is the Signetics 2650 microprocessor. Basically, it is a simple to use microprocessor with a powerful instruction set. The input device can be any ASCII-encoded keyboard, which simply hooks to an input port of the computer. The output device is a high quality 80-character by 16-line video display generator which is fully controlled by the processor. There are also parallel input and output ports (one each) to allow you to communicate with any other peripheral device, such as a line printer or a floppy disc. The

SPECIFICATIONS

DISPLAY GENERATOR

Display Format: 16 lines of 80 characters. User programmable character generator selected by setting bit 6 of the stored data to logic-1 level. Bit 7 available for any special user application.

Method of Accessing: The processor writes correct RAM location to place a character on the screen.

Output: Composite video, 1 volt P-P nominal, 75 ohms, 7.1 Mhz.

Cursor: Written by processor just like any other character. Supervisor uses a square dot centered in character field.

Screen Blanking: During horizontal or vertical retrace, between character lines and whenever the processor is accessing the display memory.

Display Addressing: Starts (upper left hand corner) at address H1000, and ends at H14FF. As addresses are incremented, the display position moves downward.

Additional User RAM: A total of 768 bytes starting at address H1500. Actually a part of the display memory, but this is transparent to the user.

PROCESSOR

Type: Signetics 2650 microprocessor.

Buffering: All processor signals buffered for TTL fan-out of 10.

Memory: 1K bytes of PROM which contains supervisor program. 768 bytes of RAM unused by display is available for user programs. PROM is on-board expandable to 4K bytes.

Control Lines: Pause and stop-clock lines allow single stepping of programs. Disable-lines for address data, and control buses to allow DMA (Direct Memory Access) or dual processor operation.

CASSETTE INTERFACE

Recording Format: 300 baud, 1200/2400-Hz Kansas City Standard.

Output Voltage: 100 mV P-P.

Misc. Size: 8 1/2-inch square printed-circuit board.

Power Consumption: 3 amps at 5 volts.

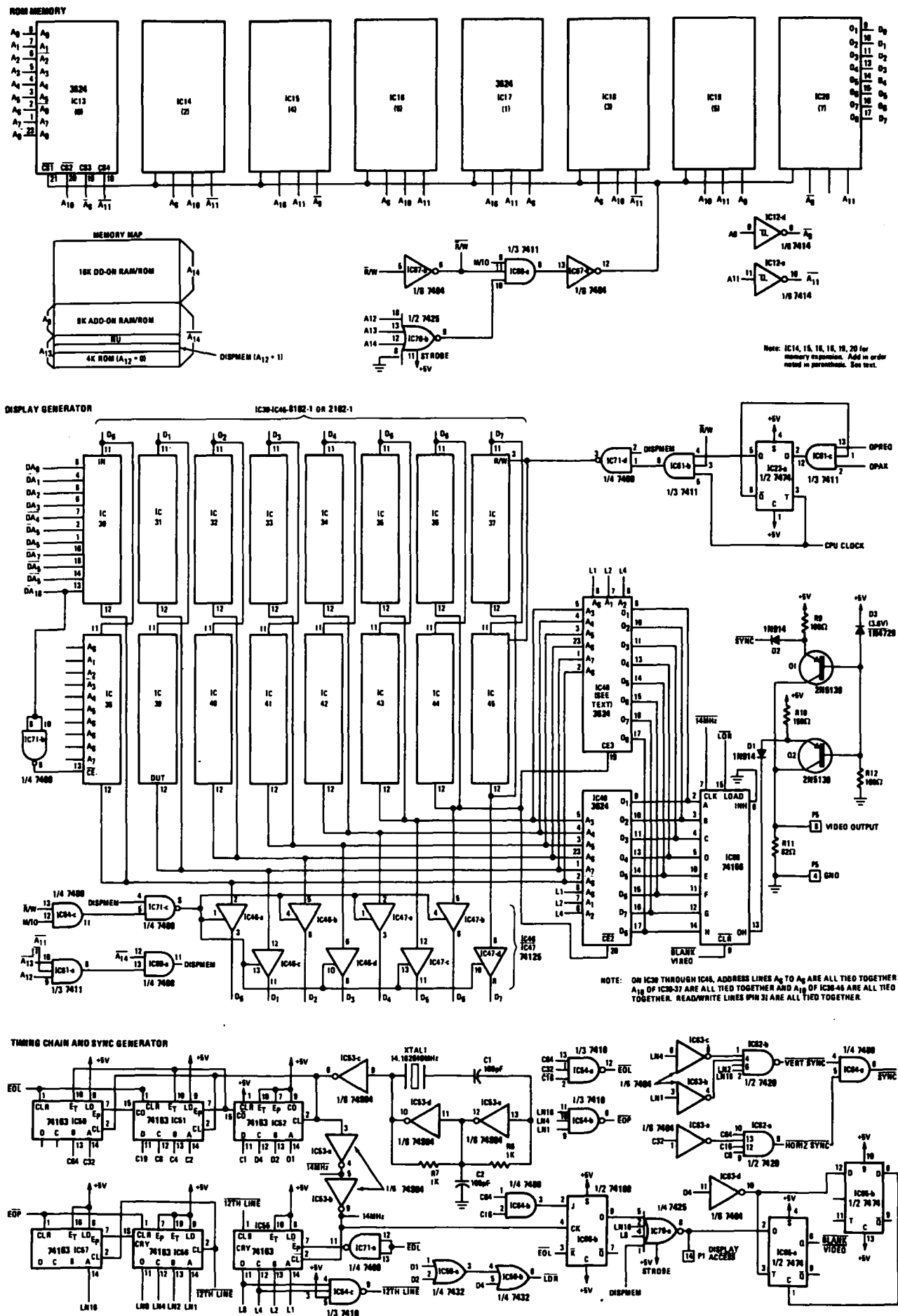
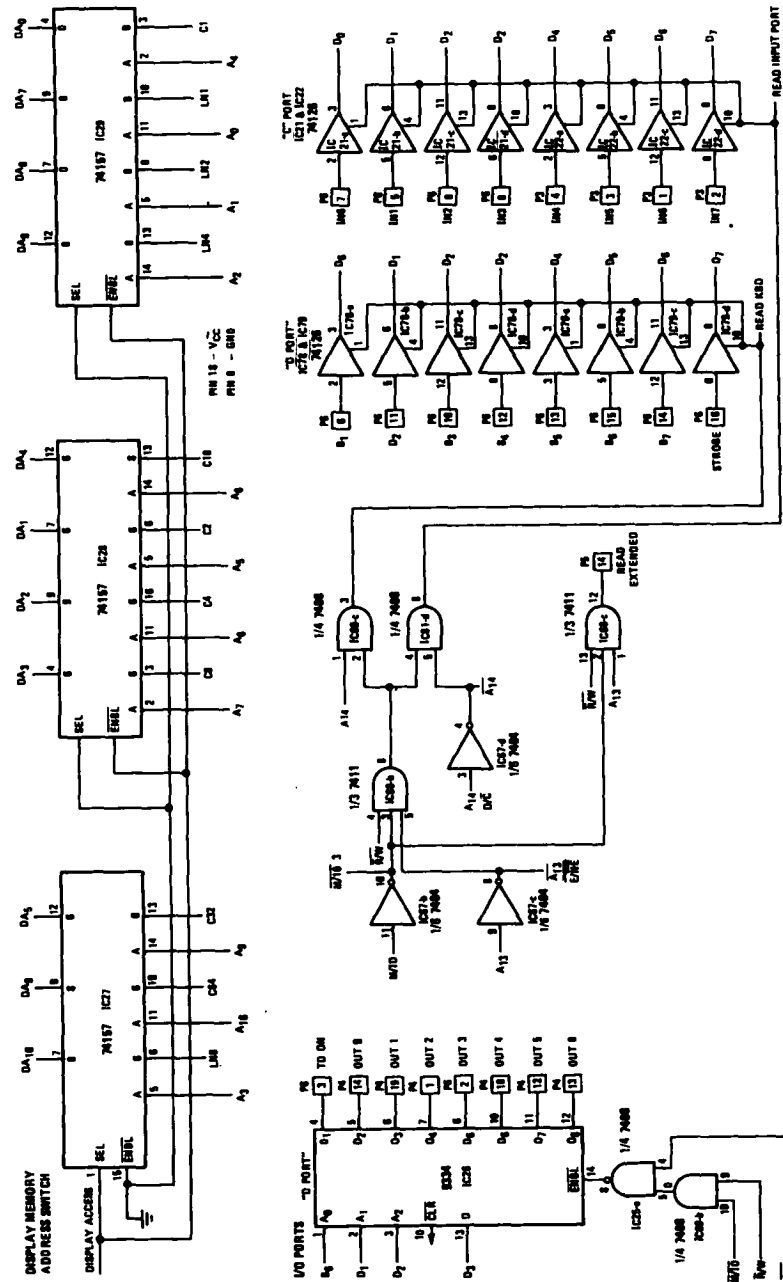


FIG. 1—COMPLETE SCHEMATIC of 2850-based microcomputer. All circuitry, including cassette-tape interface and video display generator, mount on single PC-board. Connection to peripheral devices is made via six IC sockets.

PROCESSION AND BUS DRIVERS



CASSETTE TAPE INTERFACE



PARTS LIST

All resistors 1/4 watt, 5%.

R1-R6, R14, R18-10,000 ohms

R7, R8, R16, R17-1000 ohms

R9-330 ohms

R10-150 ohms

R11-82 ohms

R12-100 ohms

R13-620 ohms

R15-470 ohms

R19-20,000 ohms

R20-68,000 ohms

R21, R22-20,000-ohm trimmer potentiometer

C1, C2, C11-100 pF, disc

C3, C14-100 μ F, 16 volt, electrolytic

C4, C8-0.01 μ F, disc

C5, C6, C7, C9, C15-C29-0.1 μ F, disc

C10-0.0022 μ F

C12, C13-0.056 μ F, $\pm 10\%$ polyester film

D1, D2-1N914 diode

D3-1N4729 Zener

D4, D5-1N4148 diode

Q1, Q2-2N5139 transistor

IC1-2650 microprocessor (Signetics)

IC2-IC10, IC21, IC22, IC78, IC79-74126

IC11, IC64, IC69-7408

IC12-7414

IC13, IC17-3624, pre-programmed

PROM containing supervisor program

IC14-IC16, IC18-IC20, IC48-3624,

PROM (see text)

IC23, IC24, IC65, IC75, IC77-7474

IC25, IC71, IC76-7400

IC26-9344, 4-bit by 2-bit multiplier

(Fairchild)

IC27-IC29-74157

IC30-IC45-2102-1, RAM (Signetics)

IC46, IC47-74125

IC49-3624, pre-programmed PROM

character generator, upper case

IC50-IC52, IC55-IC58-74163

IC53-74S04

IC54-7410

IC59-7432

IC60-74166

IC61, IC68-7411

IC62-7420

IC63, IC67-7404

IC66-74109

IC70-7425

IC72-555, timer

IC73-74123

IC74-CA3130

XTAL1-14.192640 MHz series-resonant crystal

MISC.—One 40-pin DIP socket for IC1, six 16-pin DIP sockets and printed-circuit board.

The following parts may be ordered from: Central Data Company, P.O. Box 2484, Station A, Champaign, IL 61820.

IC49-3624, pre-programmed PROM character generator, upper case, \$27.

IC13, IC17-3624, pre-programmed PROM containing supervisor program, \$27 each.

PC board, predrilled and etched, \$30.

An assembled and tested microcomputer board, \$325.

number of ports can be easily expanded to 256.

The on-board memory consists of 1024 bytes of PROM used to store the supervisor program along with 2048 bytes of RAM used for the video-display generator and for program storage. The external storage device is a cassette-tape recorder. An on-board cassette-tape interface IC is provided that uses the 1200/2400 Hz, 300-baud standard to store the data. When the cassette interface is not being used, data lines are available for serial I/O (Input/Output) data transfer. The computer operates off of a single +5-volt supply. When it comes right down to it, all that you need to get rolling in microcomputers is the board described in this article, plus an ASCII-encoded keyboard, a video monitor, a cassette-tape recorder and a power supply.

Something more should be said about the supervisor program at this time since it pulls all of the hardware together to form a simple to use computer system. The program allows you to have cassette-tape input or output to or from any memory block (in a standard format, in any length that you want), display or change memory, set a software breakpoint (stop) address, inspect and set the CPU registers, and jump to any memory location to execute your program.

Theory of operation

Figure 1 shows the schematic for the 2650 microcomputer system. Because of the architecture of the 2650 microprocessor, the Processor and Bus Drive circuit is very simple. All of the output lines of the 2650 are buffered by tri-state buffers, and the data bus is buffered in both directions, with only one set of buffers being enabled at a time.

Figure 2 shows the timing relation-

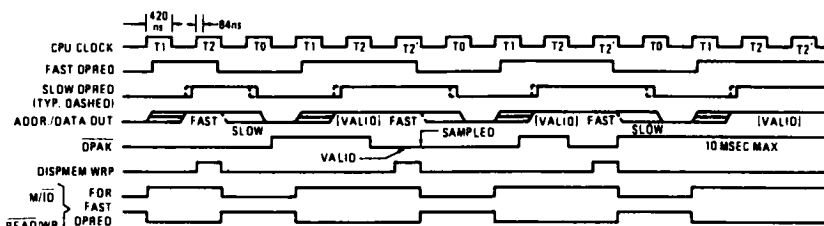


FIG. 2—2650 MICROPROCESSOR timing relationships.

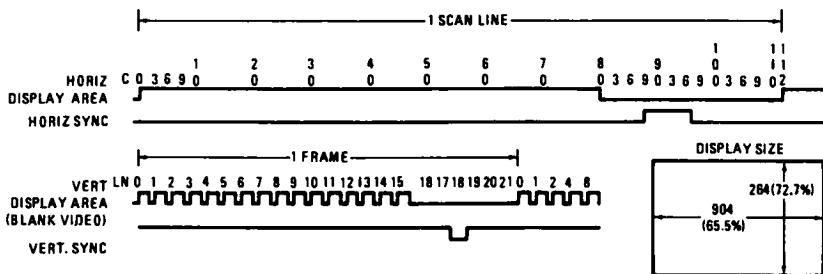


FIG. 3—TIMING DIAGRAM of horizontal, vertical and sync signals supplied to video monitor.

ships for the 2650 microprocessor. Note that both OPREQ (operation request) and the address lines become stable somewhere in an interval of about 600 ns, depending on the individual 2650, the temperature and the power supply voltage. The 1.18 Mhz TTL clock for the 2650 is derived from the output which is a division of the 14-MHz master oscillator used in the display unit.

Because the microprocessor and the display unit use the same RAM (unless you expand the RAM), a priority arrangement for the memory had to be devised. Since the processor could be in the middle of an access when the display needs the memory again, the display checks to see if the processor is accessing RAM, and if it is, the display waits. After the processor is finished accessing the memory, the processor is locked out from accessing the memory again until the display is through. The

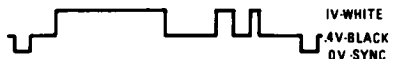


FIG. 4—COMPOSITE VIDEO SIGNAL supplied to video monitor.

display releases the memory to the processor whenever it is doing inter-line or vertical blanking, which accounts for about 53% of the time. Therefore, the processor runs at about half-speed when continually accessing the display memory, which is very seldom. When the display is using the memory, pin 36 (OPAK) of the 2650 is high.

Data is transferred between the processor and peripheral devices via the parallel input and output ports. The keyboard should be hooked to the input port by connecting it to the correct pins of plug P6. The data from the keyboard is read by the READ DATA command. The other input port is read by a READ CONTROL command. The bit settable

output port is accessed by the WRITE DATA instruction.

The Timing Chain and Sync Generator circuit divides the 14 Mhz clock in several stages to form the various signals needed to interface with a video monitor. Derived from these divisions of the master clock are the signals HORIZONTAL SYNC, VERTICAL SYNC, and BLANK VIDEO (composite blanking). Figure 3 shows the relation between the vertical and horizontal timing and the sync pulses.

The Display Memory circuit consists of sixteen 2102-1 RAM IC's and up to two 3624 PROM's. The 3624 PROM's are used for both the character generators and the supervisor program (as opposed to 1702's) so that the whole system could run off of a single supply. The use of PROM's for character generators also allows you to create your own characters, symbols or limited graphics. If bit 6 of the ASCII data is low, it selects one of the character generators (IC49), while if it is high it selects the other (IC48). The outputs of the character generators are fed into a parallel-to-serial converter where they are sent out at a 14 Mhz rate. They are then mixed with the sync pulses to form the composite video signal that is sent to the monitor. The video output voltages are shown in Fig. 4.

Since the processor and the display share the display memory, the address lines of the display memory must be switched between the processor address bus and the display timing generator. The Display Memory Address Switch circuit does this. When the Display Generator is not writing characters (and therefore not accessing memory), the DISPLAY ACCESS line is low, which selects the processor address bus to be gated to the RAM. When the display is not blanking the video, however, the Timing Chain and Sync Generator circuit selects the addresses for the RAM in an ordered fashion.

There is a special pattern by which the display accesses the RAM. This pattern allows all locations of memory not used by the display section (there are 768 of these bytes) to be confined to one memory block, not just several bytes here and there scattered throughout memory. A memory map for the display is shown in Fig. 5. When the address of the RAM is incremented, the

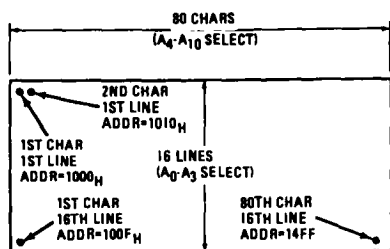


FIG. 5—RANDOM ACCESS MEMORY is simultaneously used for the video display and program storage.

PIN NO.	P1	P2	P3	P4	P5	P6
1	ADDR DISABLE	A ₁₁	IN6	OUT 2	M/I/O	GND
2	CTRL DISABLE	GND	IN7	+5V	A ₁₄	OUT 3
3	SERIAL INPUT	+5V	IN5	GND		TD-ON
4		A ₁₀	IN4	D ₇	GND	+5
5	RUN/WAIT (OUT)	A ₉		D ₆	VIDEO OUT	IN1
6	PAUSE	A ₆		D ₀	A ₁₂	IN3
7	DB DISABLE	A ₅		D ₁	+5V	IN0
8	STOP CLOCK	A ₇	GND	D ₂	+5V	IN2
9	SERIAL OUT	A ₈	TAPE RECORDER EARPHONE	D ₃	GND	B ₁
10	RESET OUT (SQUARED)	A ₁	TAPE RECORDER MICROPHONE	D ₄	WRP	B ₃
11	+5V	A ₂		D ₅	A ₁₃	B ₂
12	+5V	A ₉		OUT 5		B ₄
13	CPU CLK	A ₄	GND	OUT 6	R/W	B ₅
14	DISPLAY ACCESS	+5		OUT 0	READ EXTENDED	B ₇
15	RESET IN	A ₉		OUT 1	OPREQ	B ₆
16		A ₃		OUT 4	A ₁₃	STROBE

FIG. 6—PIN CONNECTIONS of DIP sockets used to connect microcomputer to external devices.

character on the video display moves down one line. When it gets to the bottom line, the next character is on the top line, one character to the right of the present one. Therefore, when the display memory is filled, every memory location from address 1000 to 14FF is filled with data. If you wish to write characters across a line, you must increment the address by 16, thus moving you to the right one position. So from locations 1500 to 17FF you may put your own programs and data, just as if the video display wasn't there.

It may help to show what would happen if the memory addressing scheme was worked out another way. If the addresses were incremented as the characters were being printed on a line, then addresses 80 to 127 would be left unused, along with 207 to 254 and so on to form 16 groups of 48 characters each. All of these groups are separated by 80 characters, and programs cannot be written in them unless they are less than 49 bytes long! The addressing format can be ignored by someone programming the unit, for it is transparent to the programmer. It was thought that it would be appropriate, however, to bring it out here for a more hardware oriented person.

The PROM memory consists of up to eight 3624 4K-bit PROM's that are selected as a function of the upper six address lines. The first two IC sockets (IC13 and IC17) on the PC board are usually filled with the supervisor program. It should be pointed out that other programs could be written that would allow the board to have a special purpose such as an intelligent terminal or a process controller. These programs could either replace the supervisor program or be added to the empty sockets.

The last circuit is the Cassette Interface, which connects to the serial input and output pins of the 2650 processor. The modulator consists of an oscillator and gating circuitry to allow either 1200 or 2400 Hz go to the tape recorder's microphone jack. The demodulator consists of a limiter (IC74) and two Monostable Multivibrators (IC73-a and IC73-b). The output of monostable IC73-a goes low if the input frequency is not fast enough to keep retriggering it (somewhere between 2400 and 1200 Hz). This output is latched by IC75-b, and the other monostable (IC73-b) takes care of bias and distortion problems. This output is sent to the serial input (sense) line of the 2650.

Installation

Connecting the circuit board to external devices is accomplished through the use of DIP plugs and sockets. Figure 6 shows the pin connections of these sockets.

Connection to your video monitor is made using plug 5, pins 4 and 5. Pin 4 is the ground, and pin 5 is the 1 volt P-P composite-video signal. Plug 6 is used to connect to your keyboard. The pin numbers that correspond to the ASCII data bits can be found in Fig. 6.

The strobe signal must be low going, and should ideally be active as long as the key is depressed. Note that the supervisor program will not accept a very short strobe pulse, since the software is used to detect the strobe. If your keyboard strobe is small (less than 1 ms), you can add the circuit shown in Fig. 7 to be sure that the program will catch every keypress.

Most keyboards have a strobe that lasts as long as a key is depressed, so it is then simply a matter of hooking up the

continued on page 90

BUILD A COMPUTER

continued from page 35

wires to the correct pins of the DIP plug. If you have a keyboard with tri-state outputs, they can be always enabled. Then their outputs can be wired to the inputs of the on-board buffers as in a normal keyboard.

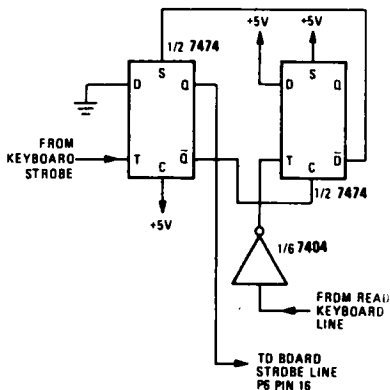


FIG. 7—DURATION OF STROBE pulse from keyboard is increased using this add-on circuit.

To hook up your cassette-tape unit you use plug 3 of the board. Pin 10 is for the record jack and pin 9 is for the earphone jack. Pin 14 of plug 3 can be used to turn off and on the tape recorder (using the auxillary control line) by simply hooking up a relay and switching transistor to the circuit as shown in Fig. 8. You may need to adjust the volume and tone controls on the

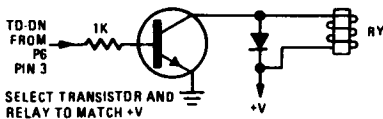
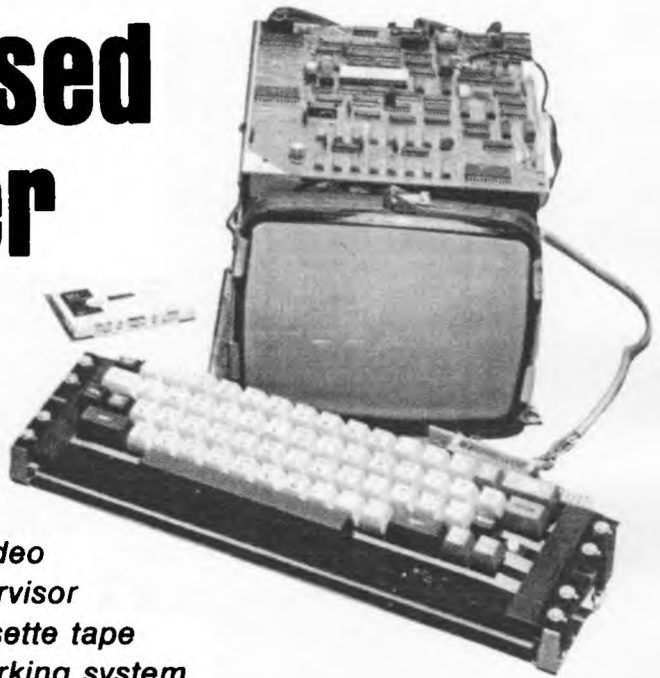


FIG. 8—AUTOMATIC CASSETTE TAPE operation is obtained by adding driver transistor and relay.

tape recorder in order to get perfect data storage (no errors on loading).

continued next month

Build 2650-Based Microcomputer System



Part II. Built on a single printed-circuit board, this 2650 microcomputer contains a video and cassette tape interface and resident supervisor program. Add a keyboard, video monitor, cassette tape recorder and power supply for a complete working system

JEFF ROLOFF

LAST MONTH, A DESCRIPTION INCLUDING specifications and schematic appeared.

This month, an in-depth look at the troubleshooting procedures plus the component placement diagram is given.

Troubleshooting

After applying power to the PC board, the first thing to check is the power-supply voltage. If the output of an adequate (5 volts, 3 amperes) power supply is pulled down by connecting it to the board, there is either an IC put in backwards (if the supply is not pulled completely to ground) or else the power-supply lines are shorted somewhere on the board. If an IC is plugged in backwards, it will become very warm immediately after the power is applied. If the supply is actually shorted to ground, the process is much more tedious since you must inspect both supply lines to discover the point of the short. If quality IC's were purchased, the majority of problems are caused by solder or etch shorts. For this reason, it is a good idea to check the whole board for little solder spatters.

Logic and display circuits

The logic section of the board is the next part of the board to test. You need a scope for this part, and its bandwidth should be above 10 MHz. (You can also send the board into Central Data to

have a trained technician test it for \$15.00 an hour.) Connect the scope to the composite video signal that appears on plug 5, pin 5, of the board. Observe the sync pulses that appear at the zero-volt level. There should be one short sync pulse (down from 0.5 volt) every 63.5 μ s, and one long pulse every 16.667 ms. If either of these two trains of pulses are not there, or have the wrong frequency, you must check the Timing Chain and Sync Generator section of the circuitry.

The Timing Chain and Sync Generator circuit contains a crystal-oscillator as the basic timing source. (A complete schematic appears in last month's issue of *Radio-Electronics*.) This is a standard oscillator with the 14.192640-MHz crystal used as the main feedback element. The signal from this oscillator is buffered using three other inverters from IC53. The first divider network that this signal goes to is formed by IC50, IC51 and IC52. These three IC's are used to divide the master clock down to a signal that has a period of 63.5 μ s. The three counters are all fed by the same clock, so that all of the outputs change state at the same instant. The enable pins of the two low-order counters are fed from the previous counter stage(s) in an arrangement that forms a 10-bit synchronous binary counter. Pins 1, 2 and 13 of IC54 monitor the states of this counter. When all three of

the monitored lines are high, the clear line is activated which resets all of the counters.

The D1, D2 and D4 outputs of IC52 provide the first three binary divisions of the master clock. Since the characters are eight-dots wide, output D4 has the width of one character (it is the divide-by-eight output). The last output of this counter, along with all of the outputs of IC50 and IC51 (designated the 'C' outputs) provide a count of the current character position on the display. While it is counting from 0 to 79, the characters are actually appearing on the screen. While counting from 80 to 112 (during which the counters are reset), the display is in the process of horizontal blanking.

The horizontal sync signal is generated by IC62-a. The sync signal is eight character-spaces wide, which is about 4.5 μ s. Also, the three dot signals that come from IC52 (D1-D4) are all OR'ed together to form the LOR signal that is used in the Display Generator circuit to load the data from the character generator into the shift register once for every character. The LOR signal has a duty cycle of $\frac{1}{8}$ (negative duty cycle, that is), and it goes low for this one dot period at the very beginning of each character.

Now that the signals are defined for all of the areas of each scan line, the up and down position on the screen must be defined. The three other counters

(IC55-IC57) serve this function, IC55 is used to count down by 12 and the other two are connected to count down by 21. The input that enables the first counter is the $\overline{\text{EOL}}$ signal that reset the previous counters at the end of a scan line. The output of this counter is monitored at pins 3 and 4 of IC54, to determine when 12 scan lines have been counted. When this event has occurred, its output at pin 6 drops and resets the counter. The reason that it counts to 12 is that there are 12 scan lines for each character line on the screen (i.e., vertically there are 12 dots for each character). The L1, L2, L4 and L8 outputs of this counter are used to determine which scan line of a character line is currently being displayed. From lines 0 to 7, the display section is actually displaying a character (since the actual character size is eight-dots tall), while from lines 8 to 11, the display is blanking between lines.

The reset line for IC55 is used to clock the next two counters that determine the character line being displayed. The counters (IC56 and IC57) are both reset by IC54-b when a count of 21 character lines is reached. From character lines 0 to 15, the display is actually showing characters on the screen (or at least it can be), while from character lines 16 to 21, the vertical blanking is in effect. Vertical sync is generated in this time period by IC62-b. Pins 4 and 5 of IC64-a are inputs to the gate that mixes the vertical and horizontal sync signals to form the composite sync signal.

The video-blanking signal is composed of a horizontal blanking and vertical blanking signal. The horizontal blanking signal is generated by IC64-b and IC66-b. The output of IC66-b is high when C64 and C16 are both high (at character position 80) and later changes back to low when $\overline{\text{EOL}}$ is present. Any of the four inputs to NOR gate IC70 indicate that the screen should be

blank when they are high. So there are three other signals that indicate that the screen should be blanked and all of them are OR'ed together to form the master blanking signal. The vertical-blanking area is indicated by LN16. Any one of the four scan lines between character lines is indicated by L8. The DISPMEM line is high whenever the processor is accessing the display RAM. Wrong character patterns would appear when the display RAM is accessed, so we just blank the screen during this small interval to cover up the few out of place dots. The DISPLAY ACCESS signal is low whenever the screen is to be blanked. This signal is delayed one more character time (to make up for the delay in accessing the RAM and character generator in the Display Generator circuit) by IC65, using pins 2 and 5. If the processor is accessing display memory, pin 8 of IC65 is low and is a one character-time delayed version of the DISPLAY ACCESS signal. This in turn keeps BLANK VIDEO (pin 5 of IC65) low a little after DISPLAY ACCESS disappears to make up for the recovery time required by the RAM. The BLANK VIDEO signal goes to the output shift register and keeps the output data low by clearing the register.

If these circuits are operating correctly, a good video signal should be present that can be connected to a video monitor. Figure 1 shows the composite-video

resistor (R9) in the video output stage (such as 220 ohms) and lower the value of the 100-ohm resistor hooked to the base of the two transistors. This will adjust the sync level and the total video-output level to suit your monitor. If you have a standard monitor, you will not have to do this since the output signal is standard.

If the characters on the screen waver, there is still a problem with the Timing Chain and Sync Generator circuit. Check this circuit out again, looking for any imperfections in the signals that indicate a bad IC or a shorted line. This probably won't happen, since a fault in the circuit usually causes major problems and not just a little wavering on the screen.

Now that a pattern appears on the screen, it must be identified. If it is a prompting character (a period) and the cursor appears in the upper left corner of the screen, the whole system is probably working. All that is needed is to hook up an ASCII keyboard and tape recorder to the correct plugs and see if commands can be entered into the monitor. (The procedure for testing the cassette interface is given later.) If you have either random characters on the screen or just 16 solid horizontal bars, the processor section of the board is probably not working. This prevents the supervisor program from being read in and thus the screen is not cleared.



FIG. 1—COMPOSITE VIDEO SIGNAL supplied to video monitor.

signal. If the sync signal is correct, the monitor should immediately lock onto the video signal and display a stable picture. If the picture does not lock-in, try lowering the value of the 330-ohm

Processor

The Processor and Bus Driver circuit divides the master clock by 12 to obtain the CPU clock. The divide-by-6 counter IC58 starts its count at 10 and counts up to 15. After the count of 15, inverter IC53-a connected to the carryout line and load input is used to load the 10 and start the division again. Flip-flop IC66 divides this carry signal by 2 to obtain the square CPU clock. The

PARTS LIST

All resistors 1/4 watt, 5%.

R1-R6, R14, R18—10,000 ohms
R7, R8, R16, R17—1000 ohms
R9—330 ohms
R10—150 ohms
R11—82 ohms
R12—100 ohms
R13—620 ohms
R15—470 ohms
R19—20,000 ohms
R20—68,000 ohms
R21, R22—20,000-ohm trimmer potentiometer
C1, C2, C11—100 pF, disc
C3, C14—100 μ F, 16 volt, electrolytic
C4, C8—0.01 μ F, disc
C5, C6, C7, C9, C15-C29—0.1 μ F, disc
C10—0.0022 μ F
C12, C13—0.056 μ F, \pm 10% polyester film
D1, D2—1N914 diode
D3—1N4729 Zener
D4, D5—1N4148 diode
Q1, Q2—2N5139 transistor

IC1—2650 microprocessor (Signetics)
IC2-IC10, IC21, IC22, IC78, IC79—74126
IC11, IC64, IC69—7408
IC12—7414
IC13, IC17—3624, pre-programmed PROM containing supervisor program
IC14-IC16, IC18-IC20, IC48—3624, PROM (see text)
IC23, IC24, IC65, IC75, IC77—7474
IC25, IC71, IC76—7400
IC26—9344, 4-bit by 2-bit multiplier (Fairchild)
IC27-IC29—74157
IC30-IC45—2102-1, RAM (Signetics)
IC46, IC47—74125
IC49—3624, pre-programmed PROM character generator, upper case
IC50-IC52, IC55-IC58—74163
IC53—74S04
IC54—7410
IC59—7432
IC60—74166
IC61, IC68—7411

IC62—7420
IC63, IC67—7404
IC66—74109
IC70—7425
IC72—555, timer
IC73—74123
IC74—CA3130
XTAL1—14.192640 MHz series-resonant crystal
MISC.—One 40-pin DIP socket for IC1, six 16-pin DIP sockets and printed-circuit board.
The following parts may be ordered from: Central Data Company, P.O. Box 2484, Station A, Champaign, IL 61820.
IC49—3624, pre-programmed PROM character generator, upper case, \$27.
IC13, IC17—3624, pre-programmed PROM containing supervisor program, \$27 each.
PC board, predrilled and etched, \$30.
An assembled and tested microcomputer board, \$325.

frequency of this clock is about 1.183 MHz, and if the faster version of the 2650 is put into the board, the counter's inputs can be changed to increase this frequency. The CPU clock signal is gated through IC11-a with the **STOP CLOCK** signal that allows you to externally stop the clock by bringing the line low. The timing relationships of the microprocessor is shown in Fig. 2.

Pin 16 of the 2650 is the reset input, and the **RESET** signal is generated using a Schmitt trigger and an R-C time constant. Capacitor C3 is initially discharged and the **RESET** line is therefore high. As the capacitor charges through resistor R6, the output of the inverter will change to a low level allowing the processor to operate—

starting at address zero in memory. All of the outputs of the processor, except for **RUN/WAIT**, are buffered with tri-state drivers or an inverter in the case of the serial output line.

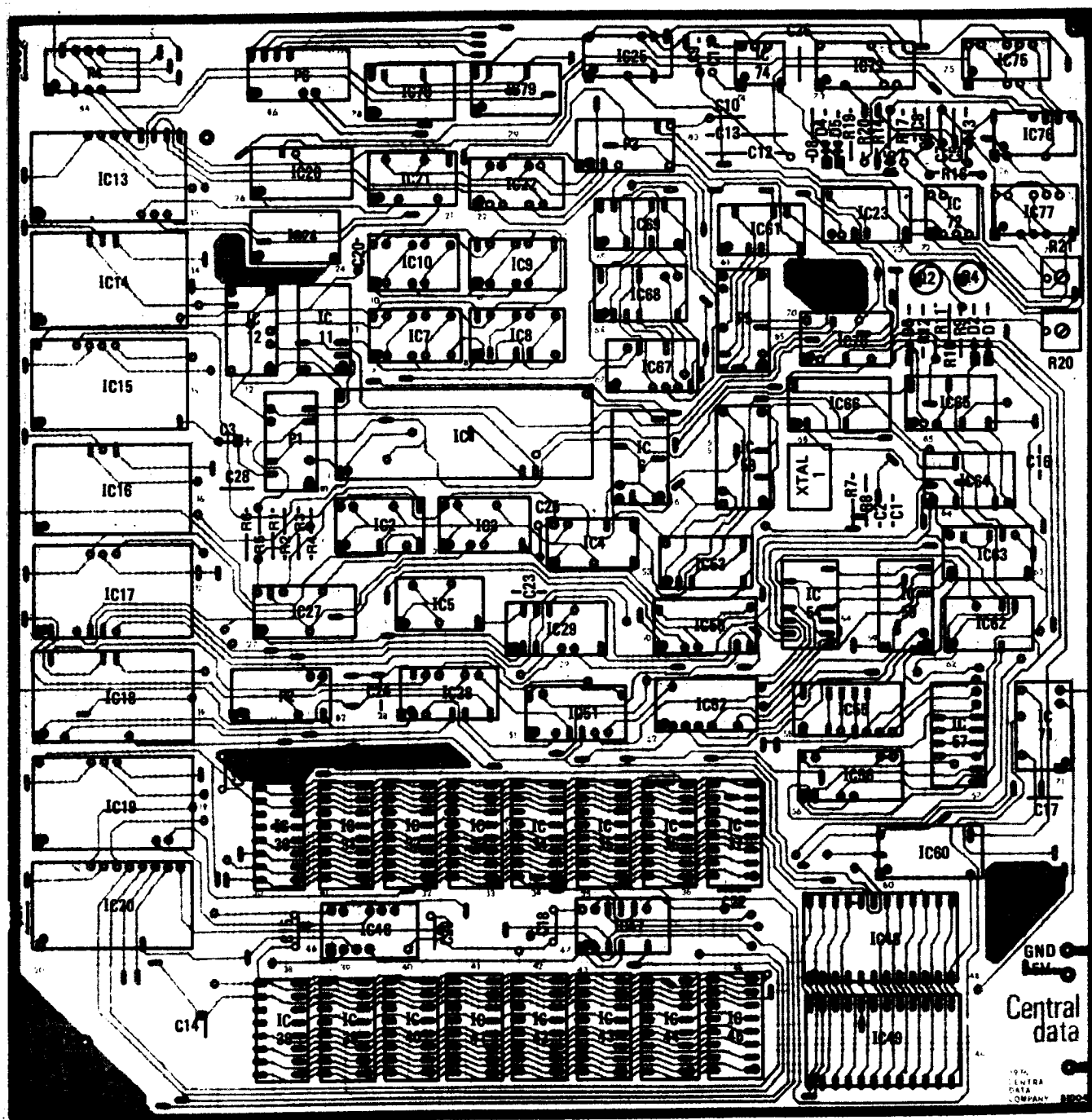
The tri-state drivers have their enable pins available for data bus override operations. For instance, bringing the **ADDR DISABLE** line low causes all of the address buffers to be disabled. This allows another device to take control of the address bus. The same is true for the data and control buses. So, by bringing these three enable lines low, you can effectively disconnect the 2650 from any external memory.

The **SERIAL INPUT** and **SERIAL OUTPUT** lines come directly from pins on the processor that can be inspected or set by

special processor instructions. This allows the system to get by without a UART and also allows high flexibility as far as output rates and formats.

The data bus buffers are tri-state buffers, with either the in or the out buffers being enabled at one time. If the **R/w** line is low, the processor is reading data, so the input buffers are enabled. If this line indicates that the processor is outputting data to the data bus, then the output buffers are enabled. Although buffers are not needed, they were put in so that any memory added to the system would not need to have its own set of buffers.

In the area of expansion, Central Data offers a S100 compatible bus board. It generates signals derived from



COMPONENT PLACEMENT diagram

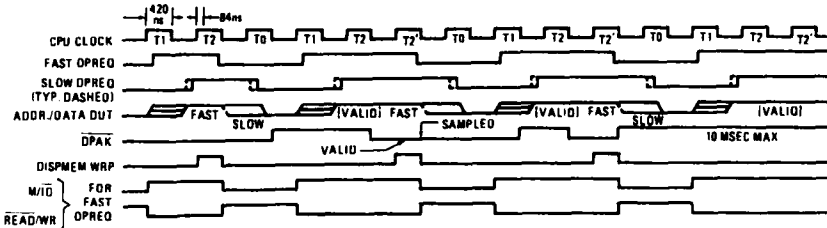


FIG. 2—2650 MICROPROCESSOR timing relationships.

the 2650 board that will interface directly with any standard 8080 static RAM board or I/O board. The board has room for five edge connectors, and is plug expandable with extender boards.

Locating a problem in the Processor and Bus Driver circuit requires inspection all of the address and data-bus lines to be sure that all of the signals are at standard TTL levels. If one line is stuck either high or low, carefully clip the pins of the IC's that this line runs to until the problem clears. The last pin that was clipped is the one holding the line, so replace the IC. Before cutting the IC pins, check the power supply on all of the IC's that the line runs to. Also, if any of the IC's are tri-state buffers, check to be sure that they are not gating data onto the line at the wrong time (an enable pin may be stuck in the enable state.)

If the processor's bus checks out, then you check to be sure that the RAM interface to the processor is working since that would also cause the random characters to appear on the screen. Initially, when the supervisor program is executed (right after reset), it clears the screen by sending the ASCII code for a blank (H20) on the data bus to the RAM, and then sequentially accessing all of the display positions and writing the data. Therefore, right after a reset, the DISPMEM line should undergo many transitions. If it doesn't, check out gates IC61-a and IC69-a that are used to generate this signal from the address lines. Note that the DISPMEM line indicates what page the processor is accessing—if it is the 4K page where the display RAM resides, this line is high.

If the DISPMEM line checks out, then look at the write-pulse generating circuitry. IC61-a (pins 1,2,12 and 13), along with flip-flop IC23-a, is used to generate a signal that is high at pin 5 of IC23-a for the first CPU clock cycle following the low to high transition of the OPREQ-signal. This is gated with R/w line and the CPU clock by IC61-b. This signal finally goes through one last gate that turns on the write line of the RAM's if the processor is accessing the display memory page. This write line should have high activity right after the power is tuned on.

If these signals check out, then check the data-bus drivers IC46 and IC47. They are used to gate data from the

display memory onto the data bus so that the processor can read the contents of the RAM. It is helpful to have a dual-trace scope for this, so that the input to the buffers and the output can be observed at the same time. Then check to see that the input and output of the buffers are the same when the enable is high.

There is almost nothing that can go wrong with the character generators, their operation is simple. IC49 is the one that is used for the normal operation of the board, with IC48 being an optional character generator for lower case, graphics, or other characters. Therefore, when supervisor program is the only thing that is running, it only selects character generator IC49. For this reason, the chip enable on IC49 (pin 20) should be low whenever the video is not blanked. This is easily checked with a dual-trace scope. If the other character generator is selected, and it is absent, a block of all on dots will appear on the screen. So, if you have several of these selections in a line, you will have a big horizontal bar on the screen rather than characters. Two of the output lines of the RAM may also be shorted, which would cause some characters not to be displayed. Again, a dual-trace scope is needed to check the data-out lines against one another to see if any look shorted.

The output shift register is IC60. It is loaded with the character generator's outputs at the first horizontal-dot position of each character, and then sends the eight-bit word out serially at the master-clock rate. Note that the clear line on IC49 is driven by the BLANK VIDEO line from the Timing Chain and Sync Generator circuitry.

The addresses for the display RAM are multiplexed by the Display Memory Address Switch. If the DISPLAY ACCESS line is low, the processor's address lines are selected to address the display RAM. This is the case when the display is being blanked (and thus the processor tries to access the display memory, raising the DISPMEM line and dropping the DISPLAY ACCESS line). The other case is when the display RAM is addressed by the Timing Chain. The signals C1 to C64 are gated in, along with LN1 to LN8, to select the correct address.

If the gating circuit for the ROM is not working properly, the supervisor program will not run. The output at pin

12 of IC-67 goes low whenever the ROM is selected. When using the supervisor program, this line will always be low when the DISPMEM line is also low. In other words, the supervisor only has two places to operate from, it will always be selecting one of these places.

The I/O ports 8 are also enabled using a good deal of gating circuitry. If the data bus doesn't have the right signal levels, check the two groups of data-bus buffers to be sure that they are not gating data onto the bus. This could happen if the enable pins are locked in a high state. Either a write data or a write control instruction will access the output port.

If the circuitry checks out, the supervisor program must be working for all of the operations except the tape routines. It is a simple matter to set up the cassette-tape interface and once it is set, it will stay perfectly aligned until it is changed. The potentiometer for the 555-timer should be adjusted so that the output of the 555 is a 4800-Hz square-wave. This is divided by two by the first flip-flop (pins 3 and 5 of IC77), and again by the second half of this same IC. The serial output line is clocked through part of IC75, and then this output is used to gate on one of the two frequencies. The R-C network at the output of IC76 (pin 11) is used to decrease the output-voltage level and change it into a more rounded signal. The demodulator section is just a monostable and operational amplifier.

continued next month

NOAA updates list

The National Oceanic and Atmospheric Administration (NOAA) National Weather Service has updated its list of stations to about the end of last year. The network is the sole government operated system for communicating weather, disaster or other warnings direct to the public. Weather receivers are available at most electronic stores.

The broadcasts are at 162.40, 162.475, or 162.55 MHz. A listing (by state, call letter and frequency) of the stations in service available free by writing to the National Weather Service, Silver Spring, MD 20910.

Zenith gets a patent for improved electron gun

Zenith's EFL (Electrostatic Focus Lens) has received Patent No. 3,995,194, granted to Zenith research and development engineers Allen Blacker and James W. Schwartz.

The EFL gun extends the focusing action over a longer distance by using four electron lens elements, instead of only two as in most electron guns. This concentrates the beam to produce a spot size as much as 60 percent smaller than in conventional gun systems. The new in-line gun is housed in the narrow neck of the company's 19-inch 100-degree picture tube.

R-E

Build 2650-Based Microcomputer System



Part III. Built on a single printed-circuit board, this 2650 microcomputer contains a video and cassette tape interface and resident supervisor program. Add a keyboard, video monitor, cassette tape recorder and power supply for a complete working system

JEFF ROLOFF

THE FIRST TWO PARTS OF THIS ARTICLE appeared in the April and May issues and provided the construction details and an in-depth look at how the circuit works.

This month, the article concludes with a look at the software associated with the 2650 microcomputer and a look at how it's programmed.

Using the supervisor

In all, there are nine basic functions of the supervisor program. First, you can alter or display any position of memory. (You cannot, obviously, alter data that is in ROM.) This will allow you to enter and inspect your own programs in the RAM. After entering and checking your program, you can use the supervisor to jump to your program and execute it. When your program returns control to the supervisor (by a branch instruction), it saves the contents of the CPU registers so that you can inspect them. You can also set the CPU registers before you jump to your program. When the program is finished and you want to turn off the microcomputer, the program can be transferred to cassette tape in blocks of two-256-bytes and then transferred back to the microcomputer at a later time. There is also a command to turn on the tape recorder so that you can manually rewind it, etc. To troubleshoot the program, a breakpoint (a point in the program where

processing will be interrupted) can be set. When this address is reached, a message is written on the screen and the CPU registers or any memory location can be inspected to see what they were immediately before the breakpoint address. You can also clear this address if you wish to change it. Another command permits verification of what is on tape against any block of memory.

The specific instructions for the operation of the supervisor are provided. In the examples, all underlined characters are ones entered by the operator. Everything else is printed on the screen by the supervisor program. A period (.) indicates that the supervisor program is ready for a command. An A indicates that it is waiting for you to type in an address. At any time the supervisor is looking for a keyboard input, you can press ES (escape) which will terminate the present command and wait for a new one.

To alter or display memory, depress the A on the keyboard. It will then ask for an address, which should be entered in hexadecimal form. The address and the data then appear on the next line of the video monitor. You can now do one of three things: depress the ES key to quit the alter/display routine, enter C to change the data at that location, or depress the space-bar to display the next memory location. If you decide to alter the memory, the supervisor will wait for

you to type in two hex characters to fill the memory location. The following is an example of this routine:

```
A A100A
100A 05      data is 05, space
              indicates go on
100B 10C3B   data is 10, change to
              3B
100C 38ES    data is 38, press es-
              cape to terminate
              routine
```

To execute a program, type an E for the command. The supervisor will then ask for the address that it should start executing at. It will then jump to the address and start executing instructions:

```
E A163B     execute at 163B, press
              space to start
```

If the program returns to the supervisor (by a branch instruction), all of the CPU registers are saved, and then it asks for a new command.

If you did return from your program by a branch instruction, or because of a breakpoint, you can inspect the memory using the alter routine, or you can inspect the CPU registers entering I. It will then ask you to type in a register number corresponding to the register that you want, as follows:

Enter	For
0	Register 0
1	Register 1, Bank 0

- 2 Register 2, Bank 0
- 3 Register 3, Bank 0
- 4 Register 1, Bank 1
- 5 Register 2, Bank 1
- 6 Register 3, Bank 1
- 7 Program Status Word,
Lower
- 8 Program Status Word,
Upper

The microcomputer will then display the data that was in this register right before the program returned to the supervisor. Similar to the alter/display routine, you now have three options: to stop by depressing the **ES** key, to change the register value by entering **C**, or to inspect another register by depressing the space-bar:

```

1
R3 2CC 02 register 3, bank 0
           has 2C, change to a
           02
R4 C3     space to go on
R8 B7ES   escape to quit

```

To transfer your program to tape, enter a **D**. The supervisor will then ask for the beginning address and the length (in bytes—up to 256) of the data to be transferred. Remember that everything must be entered in hexadecimal for the supervisor to interpret it correctly. The supervisor actually dumps one-byte more than the length that is entered, so that a length of **FF** (255 in decimal) will cause a dump of 256 bytes. Also, a length of zero indicates that this is the last block that the load routine should read in, and will cause any load of this data to be completed. This allows the load routine to load multiple blocks without having to re-enter the **L** (load command) and allows it to stop itself automatically when all the data has been loaded. Therefore, a block with length of zero should be inserted after all of your data blocks have been transferred to the cassette tape:

```

D A10DB LFF dump 256 bytes
           start at 10DB
D A11DB L10 dump the next 17
           bytes
D A0000 L00 dump an end of file
           block

```

After all of the data has been transferred, the supervisor will automatically ask for a new command.

If you wish to check the data that has been transferred to the cassette tape, use the verify (**V**) command. After entering **V**, the supervisor will then ask for an address. After this has been entered, the supervisor will start the tape recorder and will look for a block starting with this address. When the block is found, the data in the block is compared with the actual data in memory at the time of the verify. If the data is not the same as what is on the tape, an error occurs. Also, if the first block on the tape has an address different than the one that you

typed in you will get an error message.

It should be noted that the dump routine transfers the data along with the address and the length of the block:

V A1000 be sure that the first block on tape is for address 1000, and that the data is correct

The verify routine returns to ask for a new command if the verify was all right.

When using the cassette tape routines, the supervisor takes care of turning the recorder off and on. To implement this feature, you must hook the auxiliary control wires of the tape recorder to a relay, and drive this relay with the **TD ON** line from the board. You must be sure to have the recorder in the correct mode (i.e., record or play).

To load data from a tape, simply enter **L** for the command and be sure the recorder is in play mode. All of the data is recorded on the tape along with the address to load it at and the length of the load. The supervisor will ask for a new command when it is done loading the tape:

L load from tape

Recorded on tape are sumcheck characters also. Their purpose is to check against errors while recording or playing back data. The first sumcheck is sent after the address and length, while the second is sent after the block of data. Therefore, you can receive an error indication while loading or verifying in either of two places.

To set a breakpoint address in your program, enter a **B** as the command. It will then ask you for the address of the breakpoint:

B A1703 set breakpoint address
to be 1703

When this address is reached in the program, the supervisor will save all of the registers and wait for a new command. It signifies that the breakpoint address has been reached by writing the message:

BP 1703 indicates breakpoint address was reached

The registers and memory can now be examined as you see fit. After the breakpoint has been executed, it is cleared and the program will be allowed to run past the point next time through.

If you decide that a breakpoint that you set was at the wrong address, you must clear the breakpoint address by entering **C**. If you do not do this, the program will still have a supervisor inserted instruction and will not operate correctly:

C 1703

The supervisor responds by typing the address that the breakpoint was set at. Note that you must set breakpoints in an address position where an instruction would begin. In other words, you cannot set a breakpoint to be executed at an address which is the second or third byte of an instruction.

To run the tape recorder (to rewind the tape, etc.) enter **R**. Pressing escape will return you to the supervisor.

Subroutines

The supervisor program includes many useful subroutines that can be used by branching to them. The more useful ones are shown in Table 1.

All registers used are in the bank currently selected.

More information about the 2650 microprocessor and its language can be found in the *2650 Microprocessor Manual*, which is available from Signetics.

TV typewriter

Now that your system is finished and you know how to use the supervisor program, what can you do with it? One obvious use is for a TV typewriter display, which is also quite simple to do. Table 2 has the listing for the TV-typewriter program that accepts any printable character along with the backspace code and carriage return. The first thing that the program does is branch to

TABLE I

Address	Mnemonic	Description
0396	WCHR	writes the character that is in R3 on the screen and updates the cursor position.
0024	LFCR	moves the cursor to the leftmost position of the next line.
030F	KBIN	inputs one ASCII character from the keyboard and puts in R3.
006A	HXOT	takes the binary data in R2 and displays it as two hex characters.
01B6	INHX	inputs two hex characters and converts them to binary in R3.
0083	RETU	branch to this address to return to the supervisor and save the register values.

the keyboard input routine (KBIN) with a branch-to-subroutine instruction (BSTA). This subroutine receives one character from the keyboard and prints it on the display, if it is not a control character. After the character has been printed (if it is printable), the subroutine returns to the program to check if it was a backspace or a carriage return. If it was either of these two, the result of the respective compare instruction will be to clear the condition code. Then the branch instructions immediately following the compare instructions check the condition code to see if the compare was equal. If it was equal, the program branches to the correct subroutine. The carriage-return subroutine is simply a branch to the line feed-carriage return (LFCR) subroutine in the monitor, while the backspace routine is contained in the TV typewriter program.

The backspace routine simply takes the cursor pointer and decrements it. It also writes a space at the present cursor position and writes a new cursor at the new position.

The RAM positions 17FE and 17FF are used to store the present address of the cursor. To store a character in the cursor position, indirect addressing is used. This causes the processor to read what is in 17FE and 17FF and use this data as the actual address where it should do the operation.

Tape format

The cassette tape routines take care of all data encoding and decoding needed to interface with your tape unit, but if data is to be transferred between two different types of machines, you must have the format of the tape. That format is as follows:

Character	Description
1	colon indicating the start of a block
2	high order address byte for load
3	low order address byte for load
4	length of data block
5	sumcheck character for bytes 1-4
6 to n-1	data
n	sumcheck for data

Character 4 is the length of the data block. If it is zero, it represents the fact that this is the last block and that the load routine can stop. If it is from 1 to 255 (H01 to HFF), it is one less than the length of the data field. This allows transferring data blocks of exactly 256 bytes.

All characters are 8 bits wide, with one start and two stop bits. The least significant bit is recorded first, with the other bits following in order.

The sumcheck is generated by feeding each data byte into an EXCLUSIVE-OR gate with the sumcheck character and then rotating the resulting byte to the left one bit. The sumcheck is cleared

TABLE II—TV TYPEWRITER PROGRAM						
Line	Address	Instruction	Label	Operation	Operand	Comments
1	0000		LFCR	EQU	0024	ADDRESS OF LINE-FEED ROUTINE
2	0000		KBIN	EQU	0309	ADDRESS OF KEY-BOARD INPUT ROUTINE
3	0000			ORG	1600	START AT ADDRESS 1600 IN HEX
4	1600	75 08	TVT	CPSL	08	SET OPERATIONS WITHOUT CARRY/BORROW
5	1602	3F 03 09		BSTA,3	KBIN	GET KEYBOARD INPUT NOTE THAT KBIN ALSO WRITES THE CHAR
6	1605	E7 08		COMI,R3	08	COMPARE THE CHARACTER TO A BACKSPACE
7	1607	18 07		BCTR,0	BACK	IF A BACKSPACE, DO BS ROUTINE
8	1609	E7 0D		COMI,R3	0D	COMPARE THE CHARACTER TO A RETURN
9	160B	3C 00 24		BSTA,0	LFCR	IF A RETURN, DO CARRIAGE RETURN ROUTINE
10	160E	1B 70		BCTR,3	TVT	JUMP BACK TO BEGINNING—GET NEW CHAR
11	1610	07 20	BACK	LODI,R3	20	ASCII FOR A SPACE
12	1612	CF 97 FE		STRA,R3	I17FE	STORE THE SPACE AT THE CURSOR LOCATION
13	1615	0F 17 FF		LODA,R3	17FF	LOAD THE LOW ORDER CURSOR ADDR INTO R3
14	1618	A7 10		SUBI,R3	10	SUBTRACT ONE CHAR POSITION FROM IT
15	161A	CF 17 FF		STRA,R3	17FF	STORE THE NEW CHARACTER
16	161D	77 08		PPSL	08	OPERATIONS NOW WITH CARRY/BORROW
17	161F	0F 17 FE		LODA,R3	17FE	HIGH ORDER CURSOR ADDRESS
18	1622	A7 00		SUBI,R3	00	SUBTRACT BORROW FROM PREVIOUS SUBTRACT
19	1624	CF 17 FE		STRA,R3	17FE	STORE THE NEW HIGH ORDER ADDR
20	1627	07 5C		LODI,R3	5C	CODE FOR THE CURSOR
21	1629	CF 97 FE		STRA,R3	I17FE	STORE THIS IN THE NEW CURSOR POSITION
22	162C	1B 52		BCTR,3	TVT	JUMP BACK—DO NEXT CHARACTER
23	162E			END		

before data is started. When read back, each byte (including the sumcheck) goes through this routine. If no errors have occurred, the ending sumcheck character should be zero. Each block has two sumchecks and they are totally independent of one another.

Loading a program

After you have written a program, how do you load it? To many people who have been around microcomputers, the answer is obvious: use the alter

routine that the supervisor provides. To people who are having their first computer experience, this solution may not be so clear.

Recall that the alter routine allows you to change the data contained in any RAM memory location. Thus, by using this routine to change all of the memory locations that your program needs, you can enter your program into the system.

A question comes up immediately: At
continued on page 84

BUILD A COMPUTER

continued from page 49

which memory location should I start the program? The answer is that your program can be put in any RAM area, so you can start your program anywhere from address H1510 to H17E9. This is the usable RAM space that the board provides you. If you expand the RAM you could, of course, put your programs in that space also.

For an example, suppose you have the following program and want to load it into the system:

07 00	LODI,R3	00
06 10	LODI,R2	10
1F 00 00	BCTA,UN	0000

If you choose to load this program at the start of RAM, then you would alter location 1510 first. The supervisor displays the contents of 1510, and permits you to change it. Once it is changed, the next byte is displayed (1511) and you are again allowed to change it. This process continues until the entire program is loaded.

TABLE III

<u>A A1510</u>	
1510 00C07	
1511 00C00	
1512 00C06	
1513 00C10	
1514 00C1E	***note that this line has a mistake on it
1515 00e	
<u>A A1514</u>	
1514 1EC1F	
1515 00C00	
1516 00C00	
1517 00e	***done!

(e) represents the pressing of the escape key.

If you should make a mistake, simply press escape, and alter the location with the mistake and continue on. The way the screen would look for program would appear on the video monitor is shown in Table 3.

R-E
