CB ROUNDUP – DIGITAL CLOCKS – IC APPLICATIONS

$1.00 ■ SEPT. 1976

Radio-Electronics

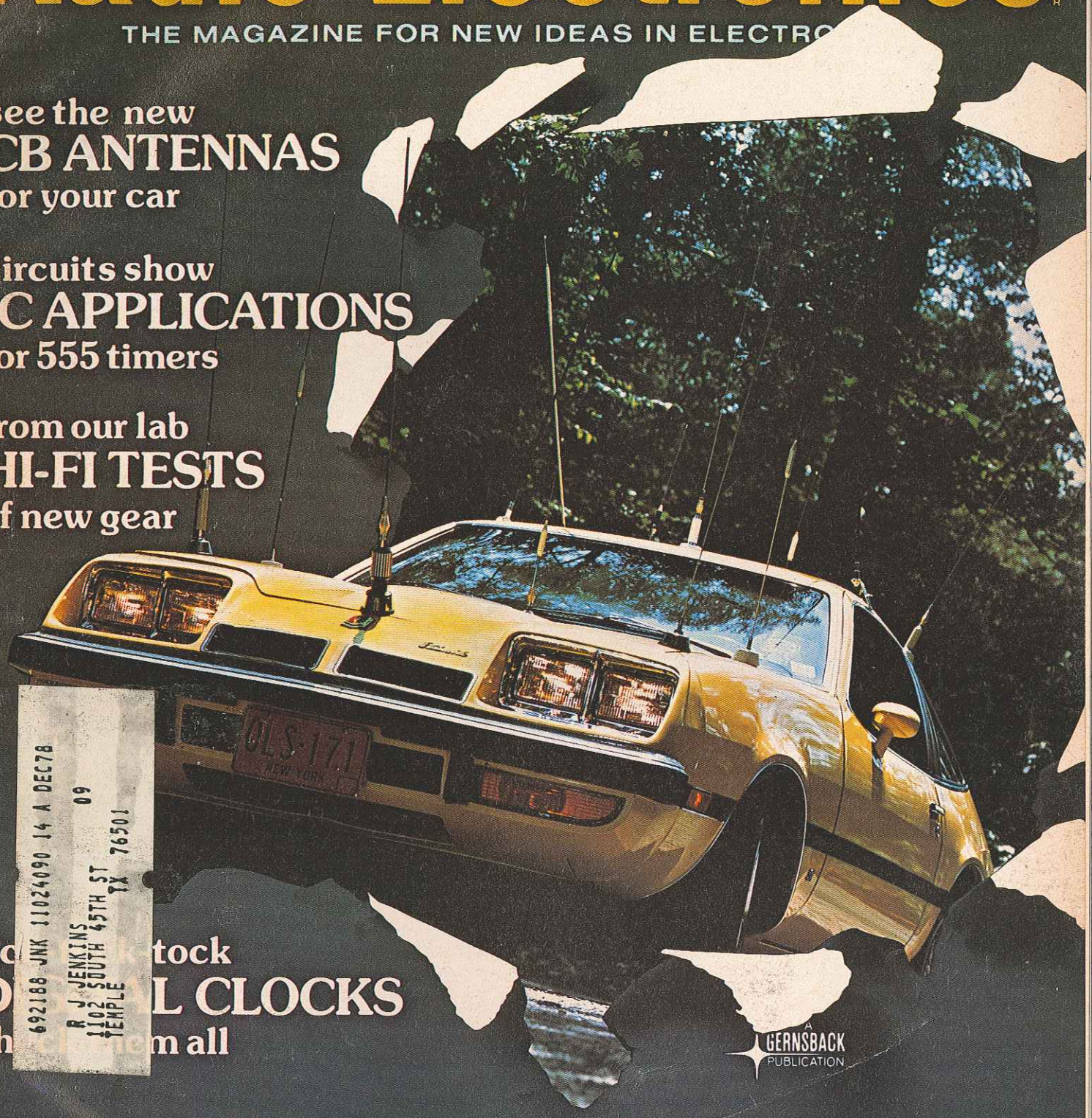
THE MAGAZINE FOR NEW IDEAS IN ELECTRO

see the new
CB ANTENNAS
for your car

circuits show
IC APPLICATIONS
for 555 timers

from our lab
HI-FI TESTS
of new gear

tic tock
D L CLOCKS
ch m all



A GERNSBACK PUBLICATION

PLUS – Bucket Brigade Time Delay ★ Selecting & Installing TV Antennas ★ Mindpower: Alpha ★ State Of Solid State ★ Digital Countdown Timer ★ Step-By-Step Troubleshooting ★ Jack Darr's Service Clinic ★ Komputer Korner

KOMPUTER KORNER

Substituting microcomputer software for hardware

JOHN TITUS, PETER RONY, and DAVID LARSEN *

A READER WHO FOLLOWS THE CURRENT LITERATURE on microcomputers will frequently encounter phrases such as "hardware/software tradeoffs" or "substitution of software for hardware." These phrases are strongly indicative of anticipated applications for microcomputers in the near future and do much to explain why industry is so excited about them. In this month's column, we would like to discuss how it is possible to "substitute" microcomputer software for hardware.

Let us first recall the definitions for *software* and *hardware*:

hardware—The mechanical, magnetic, electronic, electromechanical, and electrical devices from which a system is fabricated.

software—The programs and routines used to extend the capabilities of computers, such as compilers, assemblers, narrators, routines and subroutines.[1]

In our specific case, software represents the machine-language program that is stored within the memory of a microcomputer, and hardware represents the specific devices that store, manipulate, receive or transmit digital information. The microcomputer itself is included in our definition of hardware. The purpose of this month's column can be simply stated as follows:

Through skillful programming it is possible to substitute machine-level routines and subroutines for specific hardware devices that store, manipulate, transmit, or receive digital information. This activity is called "the substitution of software for hardware."

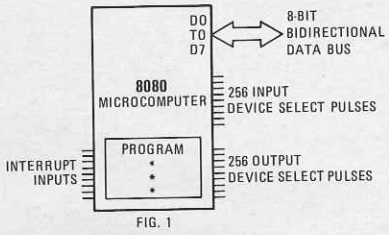


FIG. 1

TABLE 1—SUBSTITUTION OF HARDWARE BY SOFTWARE.

SUBSTITUTED HARDWARE FUNCTION	NEW HARDWARE CIRCUIT	SOFTWARE EMPLOYED
1. PULSER, 500 ns MONOSTABLE	DS 0	OUT 0
2. LOW DUTY-CYCLE CLOCK (500 ns PULSES)	DS 1	OUT 1 … JMP — TIMING LOOP
3. DEBOUNCED PULSER, MONOSTABLE, LOGIC SWITCH	DS 2, DS 3, SET 7474 CLEAR, Q	OUT 2 … OUT 3 — TIMING LOOP
4. VARIABLE DUTY-CYCLE CLOCK	DS 4, DS 5, SET 7474 CLEAR, Q	OUT 4 … TIMING LOOP; OUT 5 … JMP — TIMING LOOP
5. ONE BIT OF CONTROL INFORMATION FROM A MECHANICAL SWITCH	D0, DS 6, D 7475 Q, ENABLE	MVI A CTRL; OUT 6
6. EIGHT BITS OF CONTROL INFORMATION FROM A SET OF EIGHT MECHANICAL SWITCHES	D7 … D0, DS 7, 8212 DSI, CONTROL WORD	MVI A CTRL; OUT 7

Typical Substitutions

Typical hardware that is replaced includes knobs, buttons, pulsers, switches, logic switches, clocks, and small memories as well as TTL integrated circuits that perform

(continued on page 24)

* This article is reprinted courtesy of American Laboratories. *Dr. Rony, Department of Chemical Engineering, and Mr. Larsen, Department of Chemistry, are with the Virginia Polytechnic Institute & State University. Mr. Titus is president of Tychon, Inc., a microcomputer consulting firm in Blacksburg, Virginia.*

KOMPUTER KORNER

(continued from page 22)

digital functions such as debouncing, sequencing, shifting, adding, subtracting, comparing, and logic operations on multi-bit digital words. Hardware that is *not* usually replaced includes simple TTL IC's such as inverters, flip-flops, gates, latches, three-state buffers, and counters.

Tradeoffs

Figure 1 illustrates the basic "tools" that you would employ in the substitution of software for hardware, namely:

1. Programming.
2. The use of synchronized data appearing on the bidirectional 8-bit data bus, D0 through D7.
3. Input and output synchronizing pulses called *device select pulses.*
4. *Interrupts* to the microcomputer.

In an 8080-based microcomputer, you are able to generate 256 different input and 256 different output synchronizing-pulses. If you need more pulses, you can always employ memory I/O techniques. You therefore have an unlimited number of synchronizing pulses with which to coordinate the behavior of almost any type of digital electronic circuit. As you substitute software for hardware, your main tradeoff will be speed of operation. It is useful to remember the following rules when doing this:

In the substitution of software for hardware, the key tradeoff is speed of operation. The execution of any computer instruction takes time; the more instructions that are used, the longer it will take to execute them.

This tradeoff is not as serious as it may seem. Present 8-bit microcomputers are very fast, and future microcomputers will be at least ten times faster. The majority of existing electromechanical machines are slow by digital electronic standards. The human senses cannot participate in activities that require millisecond resolutions; i.e., in an input/output sense, we are very slow machines.

Practical examples

Table 1 summarizes some of the more commonly encountered situations where hardware such as debounced pulsers, switches, logic switches and clocks is replaced by simple wire connections, latches, flip-flops and inverters. We have provided abbreviated versions of the software required. The reader is referred to reference 2 for details on the generation of the OUT n pulses, where n is an octal number ranging between 000_8 and 377_8.

A "timing loop" is a short microcomputer subroutine that generates a precise time delay, typically greater than one hundred microseconds. As can be seen from the table, the replacement can be accomplished in most cases by the use of one or two different device-select pulses. A pair of OUT n instructions that bracket a timing loop are sufficient, when applied to an SN7474 flip-flop, to produce a monostable pulse of precise time duration. The addition of a second timing loop and a jump instruction changes the output of the flip-flop to that of a variable duty-cycle clock, the duty cycle being controlled by the relative time delays of the two timing loops.

Of particular interest is entry 6 in the table, in which an eight-position mechanical switch or eight individual mechanical switches are replaced by an 8-bit control word that is strobed into an 8212 IC from the accumulator with the aid of a device-select pulse. This control word is latched by such an action and can subsequently influence the behavior of a rather sophisticated digital circuit. The 8212 IC therefore functions as a *control register* for the circuit. We have directed your attention to this principle because it is now being widely used in an exciting new generation of *interface* chips that reduce the number of wire connections that need to be made between a microcomputer and an external device.

Table 1 provides only a few examples of how hardware can be replaced by simple software with the aid of device-select pulses. Omitted from the table are the more obvious hardware substitutions: arithmetic logic units (SN74181), digital comparators (SN7485), and shift registers (SN74194, SN74198, SN74199). Such IC's are replaced by microcomputer instructions that add, subtract, compare, and shift the 8-bit contents of the accumulator register. **R-E**

References

1. *Microprogramming Handbook* (Santa Ana, CA: Microdata Corp., 1971).
2. *Bugbook III* Derby, CT. E&L Instruments, Inc., 1975).