MAY 1979

HEWLETT-PACKARD JOURNAL

HEWLETT-PACKARD JOURNAL
Technical Information from the Laboratories of Hewlett-Packard Company

MAY 1979 Volume 30 • Number 5

Contents:

3 **A Precision, Programmable Pulse Generator,** *by Werner Hüttemann, Lutz Kristen, and Peter Aue* *Get the pulse parameters needed without stopping to adjust the pulse generator.*

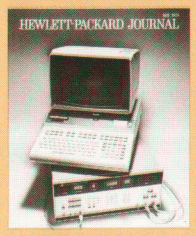
11 **Extending Possibilities in Desktop Computing,** *by Sandy L. Chumbley* *Here's a powerful midrange computer/controller with enhanced BASIC and assembly languages.*

13 **Processor Enhancements Expand Memory,** *by Damon R. Ujvarosy and Dyke T. Shaffer* *A new NMOS LSI chip and larger memory chips extend memory capacity.*

16 **Designing to Meet Electromagnetic Interference Requirements,** *by John C. Becker* *Eliminating electromagnetic pollution is assuming increasing importance around the world.*

18 **Assembly Programming Capability in a Desktop Computer,** *by Robert M. Hallissy* *Although efficient, assembly languages aren't usually easy to use. This one is.*

In this Issue:



The two products featured in this issue fit so comfortably together that we put them both on the cover. One is a pulse generator, Model 8160A, that supplies precisely controlled voltage pulses for testing all kinds of electronic devices. It generates pulses as large as 20 volts at rates up to 50 million per second, and it generates them accurately—within 2% of the amplitude and rate asked for. It's also remotely programmable, so it can respond to commands from the controller of an automatic test system.

That's where the other product comes in. Model 9835A/B Desktop Computer (also known as System 35 of the 9800 Series) is designed to act as the controller in automatic test systems made up of various stimulus-generating and response-measuring instruments, including pulse generators. The 9835A/B tells the 8160A and other instruments what to do by means of the HP Interface Bus, or HP-IB, which is HP's implementation of a widely accepted standard for communication among such devices.

Model 8160A's major contribution is its combination of accuracy and programmability. Programmable pulse generators have been available for some time, but with loosely specified accuracy, which means that their output had to be measured and adjusted to the needed accuracy before a test could proceed. With the 8160A, the controller calls for certain pulse parameters and gets them accurately enough without adjustment.

System 35's contributions include larger memory capacities than earlier desktop computers in its class, and two languages, enhanced BASIC and assembly language. Its BASIC is compatible with other HP computers, and its assembly language capability is a first for desktop computers.

-R. P. Dolan

A Precision, Programmable Pulse Generator

This 50-MHz instrument shortens setup times either on the bench or in automatic systems by generating pulses so accurately there is no need to interrupt a test to monitor the pulse waveform and make corrections.

by Werner Hüttemann, Lutz Kristen, and Peter Aue

WITH THE EVER-INCREASING COMPLEXITY of today's circuits, the only feasible way to test device characteristics is to use an automatic system. But even with automatic systems, obtaining accurate pulse stimuli has often required the test programmer to insert iterative loops that enable an operator to observe a pulse on a monitoring instrument and make corrections to the pulse parameters before continuing with the test sequence. This absorbs a considerable amount of test time.

By allocating complex instrument functions to microprocessor management, a new 50-MHz pulse generator obtains higher levels of pulse accuracy than have previously been available in programmable instruments, eliminating the inconvenience of interrupting a test sequence for operator verification every time pulse parameters are changed. In this instrument (Fig. 1), actual pulse period differs from the value specified by less than 2% in most cases, and never more than 3%. Pulse amplitude differs from that specified by less than 2%, width and delay by less than 1%, and transition times by less than 3%. Consequently, there is no need to interrupt a test sequence to examine pulse parameters on an oscilloscope. Overall test times are thus shortened significantly. Furthermore, up to nine pulse setups can be stored and recalled whenever needed, enabling pulse parameters to be changed quickly, a useful feature for bench applications as well as automatic systems.

The new pulse generator, Model 8160A, has a maximum repetition rate of 50 MHz and a maximum pulse amplitude of 10 volts into a 50Ω load when using the matched 50Ω source impedance, or 20 volts when using the 1-kΩ source impedance. Pulse leading- and trailing-edge transition times are variable, and the minimum transition time is 6 ns. The instrument has double-pulse and burst modes and an option gives it a second channel with independent control of all pulse parameters except repetition rate. The two

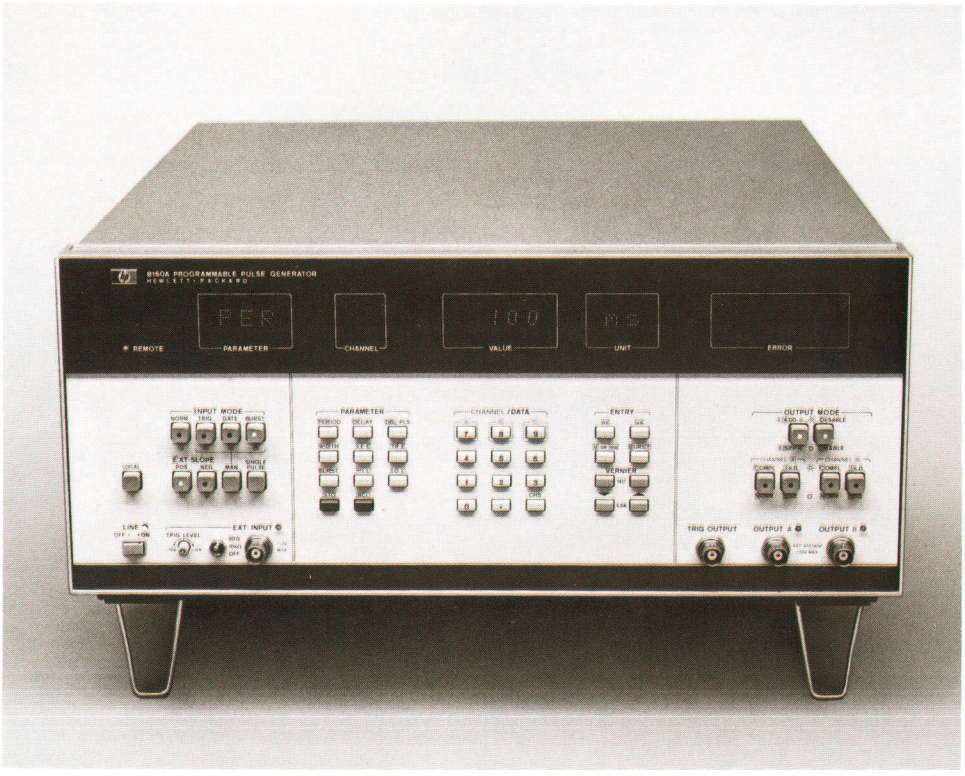


Fig. 1. *Model 8160A Programmable Pulse Generator produces pulses whose characteristics are within 1-3% of the values entered through the front panel or by way of the HP Interface Bus. Period range is 20.0 ns to 999 ms, amplitude range is 0.10 to 9.99V (into 50Ω from a 50Ω source impedance), and transition times are variable from 06.0 ns to 9.99 ms. The instrument shown here has the optional second channel that enables two-phase or three-level waveform generation.*

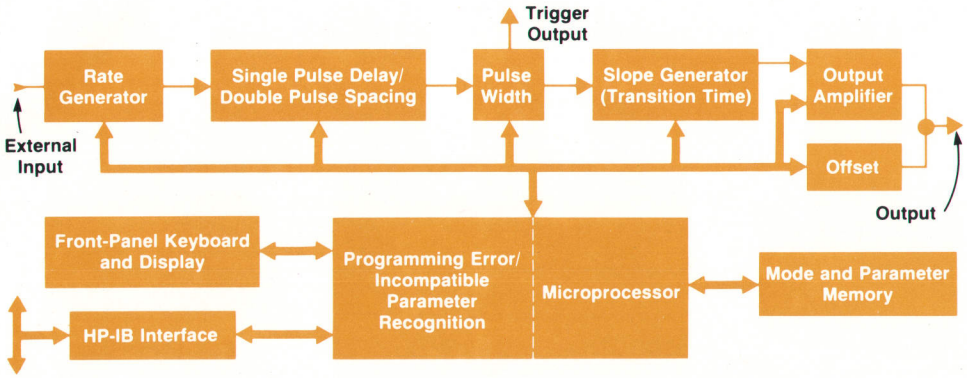


Fig. 2. *Block diagram of the 8160A Programmable Pulse Generator. All pulse parameters and operating modes are controlled by the microprocessor whether the instrument is operated manually or remotely through the HP Interface Bus.*

channels can be used for such dual-channel purposes as two-phase clocks or they may be combined into a single output to produce three-level and other complex waveforms.

Microprocessor control simplifies operation of the instrument. Pulse parameters are entered with three-digit resolution through the front-panel keyboard or through the HP Interface Bus,* and are displayed on numeric LEDs. However, the value of any parameter can be rapidly incremented or decremented manually without numerical entry by the use of pushbutton verniers. The instrument has the intelligence to recognize demands for incompatible pulse parameters—for example, pulse width greater than pulse period—and provides error indications to prompt the operator. A learn mode speeds test program development by allowing the operator to experiment with pulse parameters using the front-panel keyboard, and then transfer the selected setup to the controlling computer with a single command.

Instrument Organization

A block diagram of the Model 8160A Programmable Pulse Generator is shown in Fig. 2. As can be seen, the settings and control of all pulse parameters and modes are controlled by the 8-bit microprocessor (a 6800).

The microprocessor system is similar to that in the HP Model 8165A Programmable Signal Source.[1] It communicates with the analog pulse-forming circuits by way of a 16-bit instrument bus. Each analog circuit has a digital acceptor circuit that accepts only data that is addressed to it.

*Hewlett-Packard's implementation of IEEE 488-1975 and ANSI MC1.1.

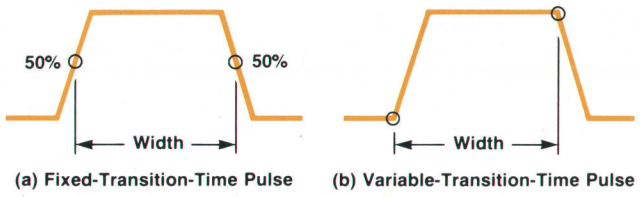


Fig. 3. *Definitions of pulse width. The traditional way defined the width of pulses with fixed transition times as the time between the 50% points of the leading and trailing edges (a). Current practice for pulses with variable transition times (b) defines pulse width as the time from the start of the leading edge to the start of the trailing edge (10% and 90% points). This measure is unaffected by a change in transition time of either edge.*

Accepted data is latched in the acceptor circuit and used to control the analog circuit.

The instrument's operating routines are stored in seven 2K-byte ROMs. The 1K-byte random-access (read-write) memory (RAM) is divided into a 768-byte, battery-supported, non-volatile section, that retains the pulse parameters and modes, and a scratchpad section.

An important part of the microprocessor's control function is error recognition. Highest priority is given to checking the syntax of instructions entered by way of the HP Interface Bus. If a wrong mnemonic is entered, the system refuses it completely.

When a valid parameter is entered, either by way of the front panel or through the HP-IB, it is checked to see if it is

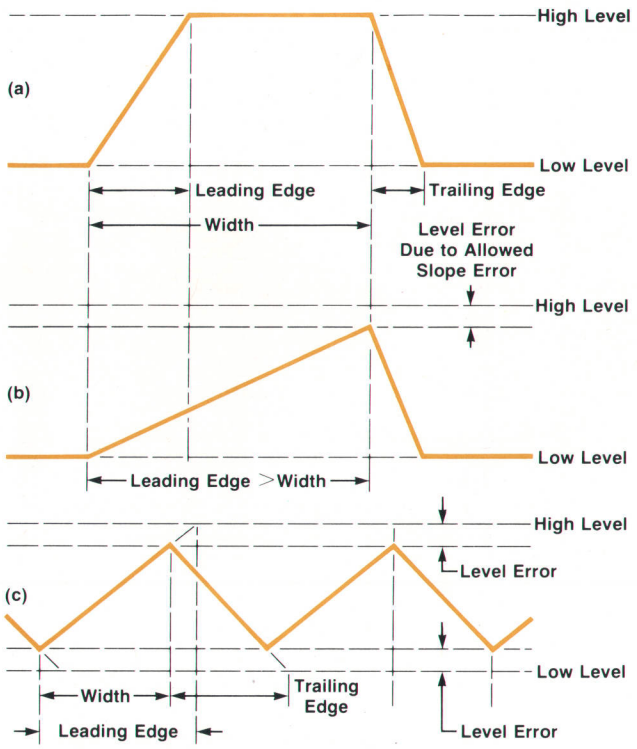
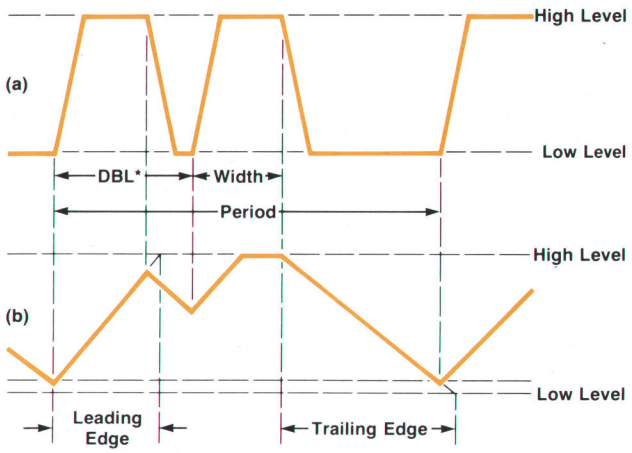


Fig. 4. *If the leading-edge transition time exceeds pulse width, the amplitude (high level) will be less than that programmed (b). If the pulse width plus the trailing edge transition time exceeds the pulse period, the pulse baseline (the low level) will be higher than that programmed. Both errors are allowed by the 8160A and can occur simultaneously (c).*



*DBL is Double-Pulse Spacing. DBL + Width Must Always Be Less Than Period.

Fig. 5. *Errors caused by double pulses running into each other are also allowed by the 8160A, enabling generation of complex waveforms like that shown in (b).*

within the allowed high and low limits, and then to determine whether or not it is compatible with the other parameters selected. If the entry is not acceptable, the output remains unchanged and the type of error is displayed on one of the four front-panel error-recognition indicators (TIMING, SLOPE, LEVEL, and PARAMETER).

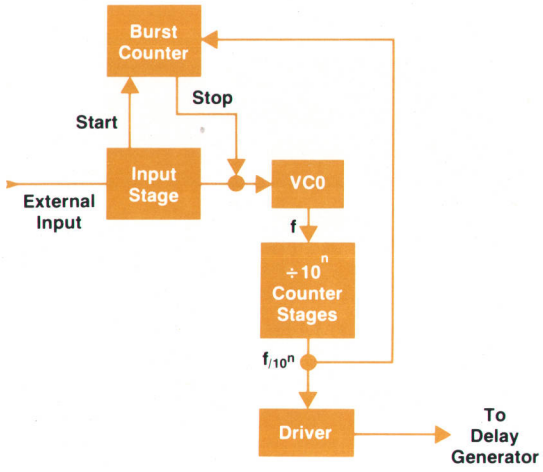


Fig. 6. *Block diagram of the rate generator. When operating in the burst mode, the VCO (voltage-controlled oscillator) is inhibited until a START pulse occurs. The preset burst counter then counts output pulses until the selected number has been counted at which time it stops the VCO. The number of pulses supplied in the burst is determined by the counter and remains unchanged if the pulse repetition rate is changed. This mode is useful for stepping counters to a predetermined count during test procedures.*

Allowable Errors for Waveform Flexibility

As users of conventional pulse generators can testify, setting up compatible pulse parameters is not always as straightforward as it might seem. For example, will the

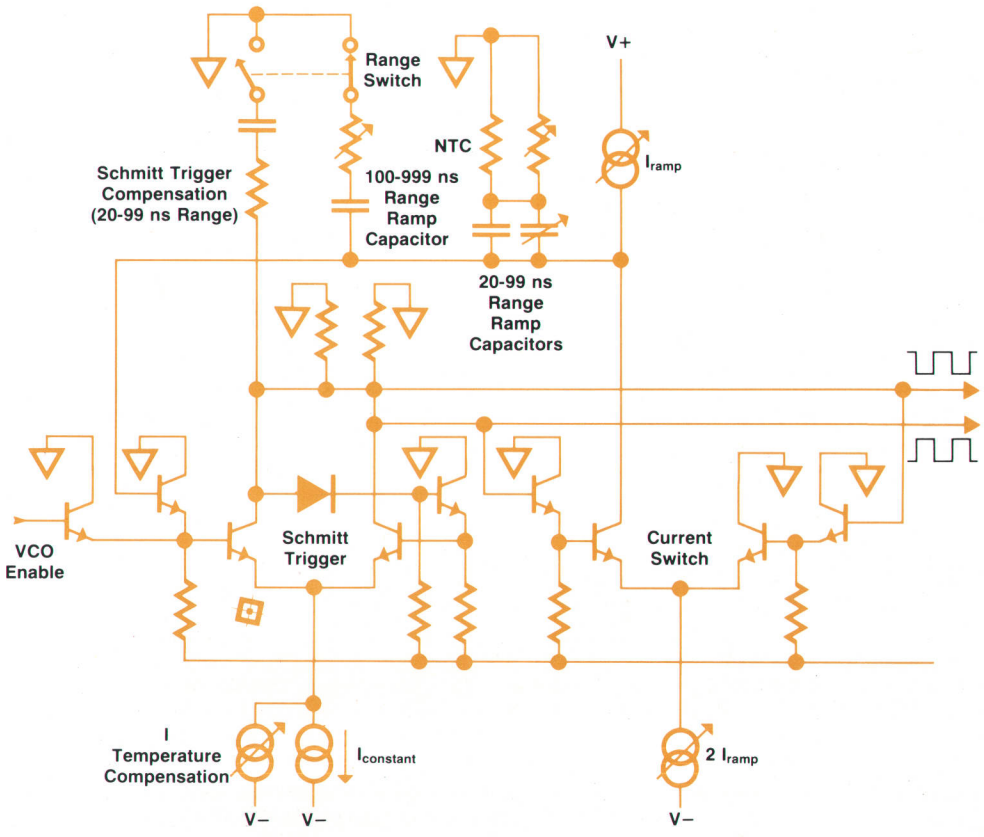


Fig. 7. *Voltage-controlled oscillator operates by capacitance charge and discharge. When the current switch is off, I_{ramp} charges the ramp capacitors until the ramp voltage reaches a level that trips the Schmitt trigger, turning on the current switch. Current $2I_{ramp}$ then discharges the capacitors, while also absorbing I_{ramp}, until the ramp drops below the Schmitt trigger's lower hysteresis level, turning off the current switch. A current controlled by the circuit's temperature varies the Schmitt trigger hysteresis to compensate for temperature-induced variations in propagation delay.*

pulse width still be compatible with the pulse period when a change in transition time stretches the pulse? Is it necessarily an error if lengthening the transition times causes the pulse to become a triangle, reducing the pulse amplitude?

To distinguish between illegal parameter entries that may be desirable and those that are definitely wrong, the concept of "allowable" slope error was introduced. It allows the transition times to be extended such that a rectangular or trapezoidal shape no longer exists, resulting in an amplitude error that nevertheless may be tolerable. This then enables the generation of ramps, triangles, and other irregular waveforms. The instrument will indicate that a slope error exists, but it produces a waveform according to the parameters entered.

To understand what effect the allowable error has, a review of the definition of pulse width* is in order. Fig. 3a illustrates the classic half-amplitude definition for pulses that have fixed transition times. This definition, however, causes some confusion with the setting of pulse width where variable transition times are involved, so present practice is to measure pulse width from the start of the leading edge to the start of the trailing edge (actually the 10% and 90% points), as shown in Fig. 3b. This measure is unaffected by transition time as long as the leading-edge transition time is less than the pulse width.

If the leading edge transition time exceeds the pulse width, then a ramp or triangle is generated and the apex lies below the programmed level (Fig. 4b). Similarly, if the sum of pulse width and trailing edge time exceeds the pulse period, then the trailing edge runs into the leading edge of the next pulse at a point above the programmed low level or baseline (Fig. 4c). Both of these errors are allowed with the new pulse generator because waveforms with these shapes may be desired. Also, when the instrument is in the double-pulse mode, the trailing edge of the first pulse could run into the leading edge of the second pulse, or the trailing edge of the second pulse could run into the leading edge of the first pulse in the next pair. These errors are also among those permitted in the 8160A, enabling generation of complex waveforms like that shown in Fig. 5b.

Accurate Rate Generation

Among the major design goals of the Model 8160A were high accuracy, excellent tuning linearity, and low jitter in the rate generator. However, the need for synchronous gating in the burst, external trigger, and gate modes precluded the use of an oscillator with an LC tank circuit or feedback loops that would prevent instant starting.

The solution was to use a high-frequency, constant-current, capacitor-charge/discharge function-generator circuit for the voltage-controlled oscillator (VCO), and operate it over a limited range to assure the performance qualities desired. The other ranges are obtained by dividing down the VCO output frequency. A block diagram of the rate generator system is shown in Fig. 6 and the VCO is diagrammed in Fig. 7.

The VCO has only two ranges, 20-99 ns and 100-999 ns per period, avoiding a proliferation of range capacitors. For longer periods, it operates in the 100-999-ns range and the output frequency is divided down to obtain the desired

*Although the term "pulse duration," has been defined as the standard, "pulse width" is used almost universally throughout the industry.

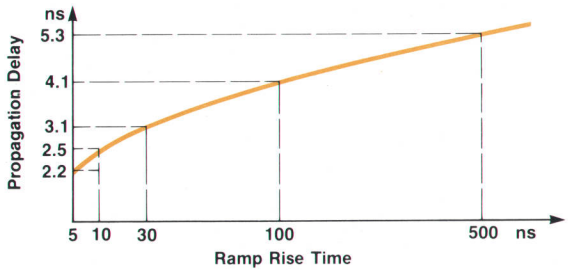


Fig. 8. *Propagation delay in an uncompensated VCO Schmitt trigger varies with the rise time of the ramp and therefore varies as the pulse period is changed.*

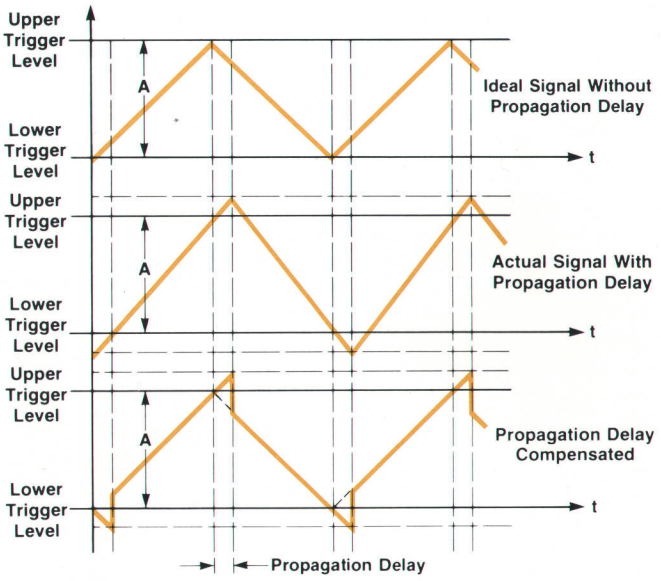


Fig. 9. *The effect that propagation delay has on pulse period is shown by the middle waveform. A resistor in series with the ramp capacitor compensates for the propagation delay as shown in the lower waveform.*

	20ns	50ns	100ns	500ns	1μs
A	50%	25%	14%	4%	2%
B	12%	8%	0%	−0.3%	−0.3%
C	0%	−1%	0%	−0.1%	0%

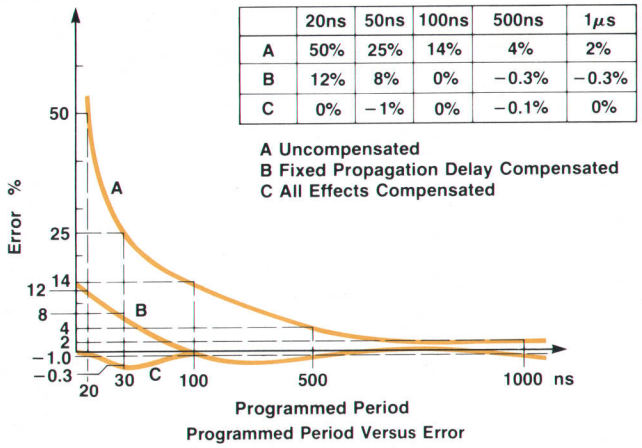


Fig. 10. *Graphs show the effects that the compensation techniques have on pulse period accuracy in the Model 8160A Programmable Pulse Generator.*

period. The microprocessor determines the values of the tuning signal and the integer n for the 10^n division factor for each period selected.

To achieve good tuning linearity in the VCO, four effects had to be accounted for. The most important of these are the propagation delay in the Schmitt trigger and the current switch, and the change in propagation delay in the Schmitt trigger as a function of ramp slope, and hence of pulse period (Fig. 8). Compensation for the propagation delay in the Schmitt trigger and current switch is often obtained in function and pulse generators by adding a small resistor in series with the ramp capacitor. This introduces a step in the ramp capacitor waveform every time the current changes direction (Fig. 9). This compensation is sufficient for period settings greater than 100 ns. For shorter periods, the Schmitt trigger hysteresis is narrowed. These measures make it possible to meet the 2% accuracy specification in the 100-ns-to-999-ns pulse period ranges and the 3% accuracy specification in the 20-99.9-ns pulse-period range, a significant improvement when compared to the 50% nonlinearity at the high end of the frequency range of an uncompensated circuit (Fig. 10).

The other two effects to be accounted for result from changes in temperature. Without compensation, there would be a ±5% change in propagation delay over the allowed temperature range (15-35°C) at high repetition rates, and a 0.2%/°C change in Schmitt trigger hysteresis. The change in hysteresis is compensated for by sensing the chip temperature and altering the temperature-compensation current accordingly (Fig. 7). The resistor in series with the ramp capacitor for the 20-99-ns range has a negative-temperature-coefficient to compensate for temperature-induced changes in propagation delay.

Delay and Width

The delay and width functions are obtained in identical time-interval stages, a switch determining whether a time-interval stage operates as a delay generator or a width generator, as shown in Fig. 11.

The pulse width is derived in the RS flip-flop (Fig. 11) which is set by the trigger pulse and reset by the delayed pulse. The very short delays (0.00 to 49.9 ns) are produced by switched delay lines, either printed-circuit board microstrip line or sections of 50Ω coaxial cable. This approach gives ±1% accuracy and low jitter, and the delay time is not restricted by the repetition rate.

The longer delays originate in an astable multivibrator that has a 25-ns delay line in a feedback path to give it a period of 50 ns with better than 1% accuracy. At the beginning of a delay interval, a counter, preloaded with an appropriate number, starts counting the multivibrator pulses. When it overflows, the counter produces the delayed pulse and stops the multivibrator. With this arrangement, delays are generated with better than 1% accuracy and the jitter is less than 0.005% of delay for delays longer than 10 μs. This compares to the 0.1% jitter specification of earlier instruments.

Slope Generator

The slope generator provides an accurate baseline and pulse top as well as providing the variable transition-time capability. To avoid degradation of the delay and width settings, the slope generator must have minimum roll-off effect. Roll-off is a significant problem in a programmable pulser because it causes a change in delay and width as the transition times are changed.

Roll-off in a slope generator results from the use of limiters to define the pulse top and bottom levels. The limiters are needed to eliminate the step that occurs with a conventional slope generator at the start of a transition. As shown in Fig. 12, the leading edge of a pulse is generated by using a positive current source to charge ramp capacitor C up to the voltage that turns on diode D1, clamping the waveform at the clamp level, V_{C+}. The trailing edge is formed by switching off the positive current source and switching on the negative current source, discharging capacitor C until its voltage reaches a level that turns on diode D2, clamping the waveform at the lower level, V_{C-}. Because of the difference between the turn-on and turn-off voltages of the clamp diodes, each transition ramp begins with a small step. The

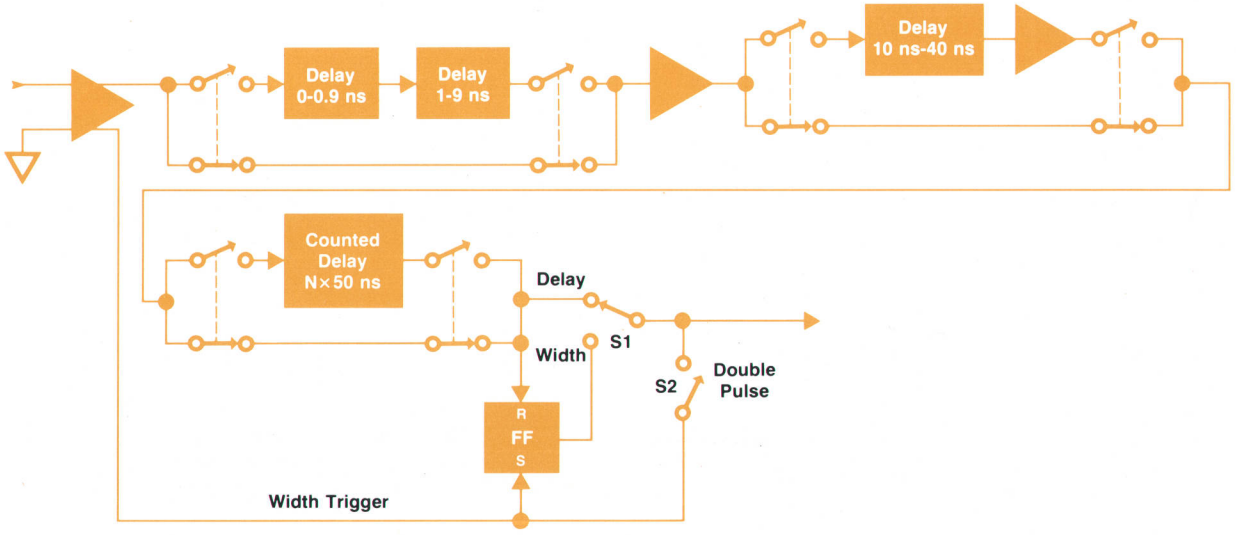


Fig. 11. *Basic time-interval circuit functions as either a delay or a width generator depending on the setting of switch S1. For double pulse operation, switch S2 passes both the trigger pulse and the delayed trigger to the width circuit following.*

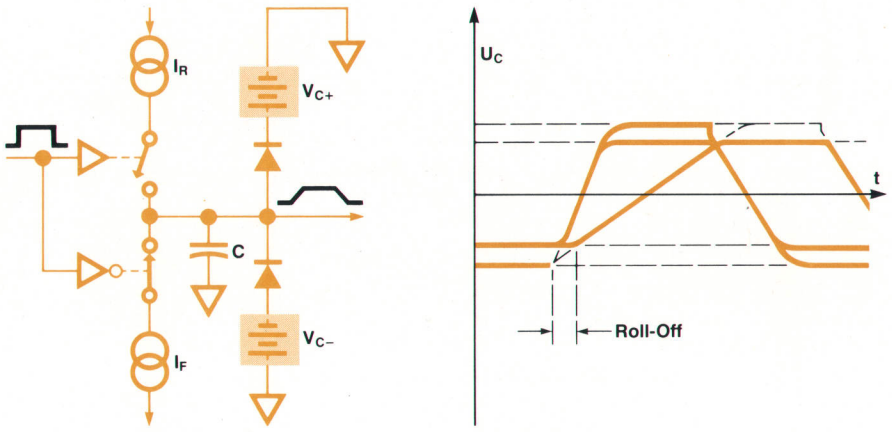


Fig. 12. *Conventional slope generator is followed by a limiter stage to remove the step at the start of each transition. Changing the leading edge slope thus affects the start time of the pulse. This is the roll-off effect.*

step is reduced by limiter stages that follow. However, as shown by the waveforms in Fig. 12, the point at which the leading edge starts then varies with the slope of the ramp.

The solution to this problem is to switch off the active clamp circuit when the current sources switch. The circuit that does this in the Model 8160A is described in Fig. 13 and the result illustrated in Fig. 14.

Expanded Amplitude Range

To be able to test CMOS devices adequately, the design objectives for the Model 8160A called for an 8-ns, 20Vp-p pulse positionable anywhere within a ±20V window. At the time of instrument design, no transistors with the requisite combination of speed and voltage ratings were available.

The solution to the problem was to design a fast, 20V amplifier powered by a floating supply. To move the pulse up or down in the window, the ground return of the amplifier's supply is simply shifted above or below ground potential. A level shifter between the slope generator and the output amplifier shifts the amplifier's input signal level a corresponding amount.

As shown in Fig. 15, the output stage also includes a switched 20-dB attenuator, an electronic vernier attenuator, and a switched second output stage that doubles the current supplied to the output. The amplitude range is up to 20 V

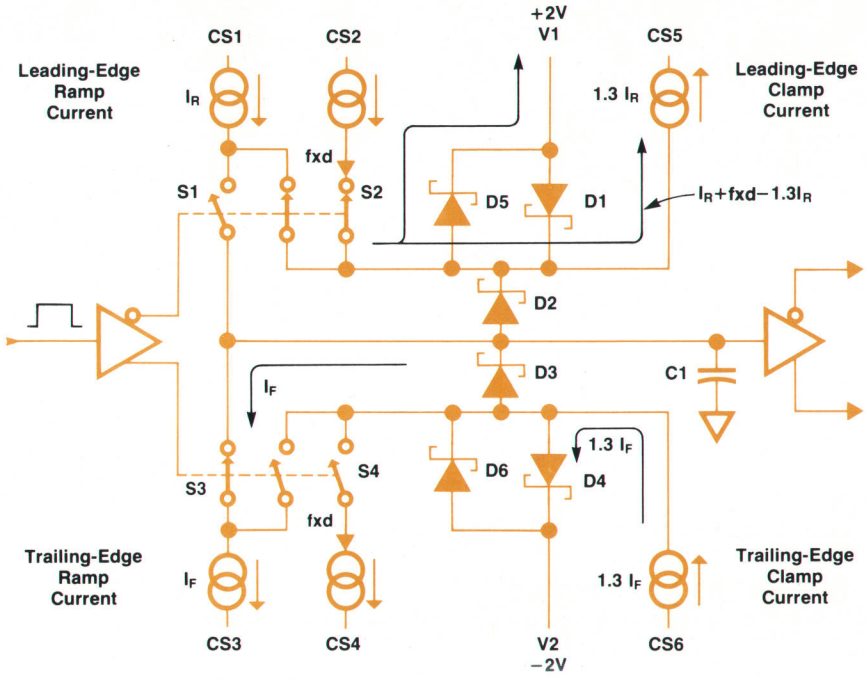


Fig. 13. *Leading and trailing edge slopes are generated by charging and discharging capacitor C through the current sources CS1 and CS3. As shown here, switch S3 is closed and current source CS3 sinks the trailing edge ramp current I_F. Diode D3 is turned off so clamp source V2 sinks a current equal to 1.3I_F, which is supplied by current source CS6. When the pulse voltage drops to a level that turns on D3, a current equal to I_F flows from CS6 through D3 to current source CS3. A current equal to 0.3I_F then flows to clamp V2 and the waveform is clamped at −2V. When a positive-going transition is applied to the buffer amplitude, switch S3 turns off and S4 turns on (S1 turns on and S2 off to start the leading edge). Current sources CS3 and CS4 then sink all the CS6 current, thus turning off diodes D3 and D4.*

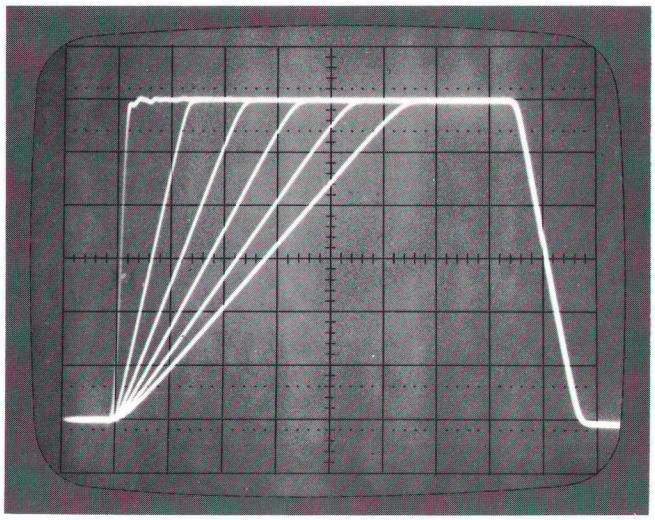


Fig. 14. *Minimizing the roll-off effect makes the start of the pulse leading edge virtually independent of the transition time, as shown by this multiexposure photo.*

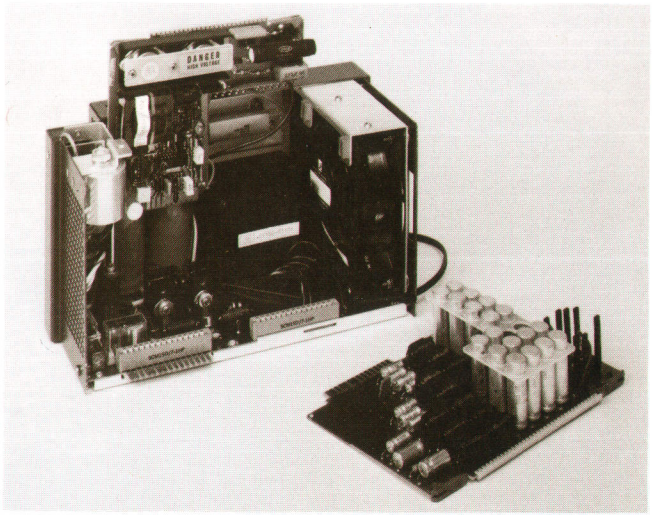


Fig. 16. *Power supply with one circuit board removed and another raised in position to disclose details. Although the supply is compact, cooling air is able to flow freely over heat-dissipating surfaces.*

(driving a 1 kΩ load from a 50Ω source impedance). Offset is added at the amplifier output so the full range of offset (±20V) is always available. The user, however, need never be concerned with determining the offset required to position the pulse top and bottom where he wants them—he simply enters the desired levels for the top and bottom of the pulse, and the microprocessor determines the necessary gain and offset required.

Compact Power Supply

The new pulse generator's high accuracy calls for extensive use of analog circuits that demand power. Space considerations put a limit of about 4500 cc (0.159 cu. ft.) on the volume that could be occupied by a power supply. With a load of 275 W, 40 percent of which is needed for the output amplifiers, the power supply requires high efficiency and a minimum number of components.

Good regulation, independent of load and supply changes, is achieved by a flyback-regulated, switched-mode power supply. It has a switching frequency of 25 kHz so a ferrite-core transformer of small size and a reduced number and volume of secondary filter components can be used. The energy delivered to the transformer core is controlled by switching the primary current at duty cycles up to 50%. A sensing winding provides feedback to a control circuit that converts the sensed level to a switching drive signal. Transformer coupling between the driver and switching stages provides the high base current for the switching transistors, which are specially fabricated to withstand the voltage and current stresses. The sensing winding also provides a supply voltage to the control circuit so that line transformers are not needed. At switch-on, when no energy is available from the sensing winding, the control circuit is supplied from a start circuit.

To allow operation from either 110/120V or 220/240V lines, the rectifier circuit acts as a voltage doubler for the low range and as a full-wave rectifier for the high range. A current limiter restricts the surge current in the doubler/rectifier capacitors to prevent damage to the rectifiers and line switch when the instrument is turned on.

This compact power supply has ten outputs totaling 275W and an efficiency of 65%. Accessibility and hence serviceability are good thanks to modular design with functionally-separated plug-in boards (Fig. 16).

Acknowledgments

Helmut Rossner designed the switching power supply

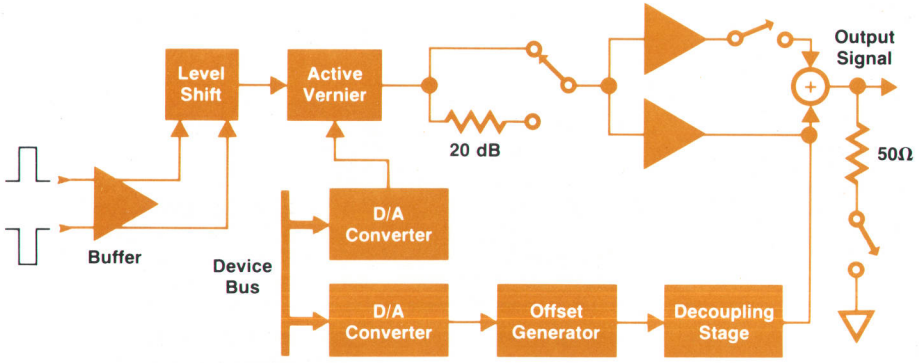


Fig. 15. *Organization of the output stage. With the 50Ω termination switched in, the output stage absorbs reflections from external impedance mismatches. With it switched out, the effective source impedance is 1 kΩ.*

SPECIFICATIONS

HP Model 8160A Programmable Pulse Generator

(Specifications are for 50 Ω source and load resistances unless stated otherwise.)

PERIOD
RANGE: 20.0 ns to 999 ms (see Table 1).
RESOLUTION: 3 digits (100-ps resolution limit).
ACCURACY: 3% of programmed value ±0.3 ns (period <100 ns).
2% of programmed value (period ≥100 ns).
MAXIMUM JITTER: 0.1% of programmed value + 50 ps.

WIDTH
RANGE: 10.0 ns to 999 ms (see table 1).
RESOLUTION: 3 digits (100-ps resolution limit).
ACCURACY*: 1% of programmed value ± 1 ns.
MAXIMUM JITTER: 0.1% + 50 ps (width ≤999 ns).
0.05% (999 ns <width ≤9.99 μs).
0.005% (width >9.99 μs).
Width is programmable to 94% of period value −30 ns.
(0.94 period −8 ns for width <50 ns).

DELAY
RANGE: 0.00 ns to 999 ms (measured from 50% point of leading edge of trigger output).
RESOLUTION: 3 digits (100-ps resolution limit).
ACCURACY*: 1% of programmed value ±1 ns (see table 1).
MAXIMUM JITTER: 0.1%+50 ps (delay ≤999 ns).
0.05% (999 ns <delay ≤9.99 μs).
0.005% (delay > 9.99 μs).
Delay is programmable to 94% of period −30 ns. Delays less than 50 ns can be programmed without being limited by the period value.

DOUBLE PULSE
RANGE: 20.0 ns to 999 ms (spacing between leading edges of double pulse).
RESOLUTION: 3 digits (100-ps resolution limit).
ACCURACY: 1% of programmed value ±1 ns.
MAXIMUM JITTER: 0.1%+50 ps (double-pulse spacing ≤999 ns).
0.05% (999 ns <double-pulse spacing ≤9.99 μs).
0.005% (double-pulse spacing >9.99 μs).
PULSE WIDTH: 98% of double-pulse spacing less 30 ns (width ≥50 ns).
98% of double-pulse spacing less 8 ns (width <50 ns).

TRANSITION TIMES (10 − 90% amplitude)
LEADING EDGE: 06.0 ns to 9.99 ms (see table 1).
TRAILING EDGE: 06.0 ns to 9.99 ms (see table 1).
Leading and trailing edge transition times are independently programmable within a common range. Ranges are overlapping as follows:

06.0 − 99.9 ns	05.0 μs − 99.9 μs
050 ns − 999 ns	050 μs − 999 μs
0.50 μs − 9.99 μs	0.50 ms − 9.99 ms

RESOLUTION: 3 digits (100-ps resolution limit).
ACCURACY: 3% of programmed value ±1 ns (see table 1).
LINEARITY: 3% for transition times longer than 30 ns.
PROGRAMMABILITY (without loss of amplitude):
LEADING EDGE: 70% of width
TRAILING EDGE: 70% of (0.94 period less width).

Output Mode	Period min	Width min	Delay accuracy	Leading, Trailing edges min	Leading, Trailing edges accuracy
A SEP B, 50 Ω	20 ns	10 ns	1%±1 ns	6.0 ns	3%±1 ns
A SEP B, 1 kΩ	25 ns	12.5 ns	1%±2.5 ns	8.0 ns	3%±2 ns
A ADD B, 50 Ω	50 ns	25 ns	1%+6/−0 ns	15 ns	3%±4 ns
A ADD B, 1 kΩ	50 ns	25 ns	1%+6/−0 ns	15 ns	3%±4 ns

TABLE 1: Output Modes/Timing (into 50 Ω load)

*With fastest transition times.

OUTPUT LEVELS (into 50 Ω load)

Output Mode	Typical Source Impedance	High Level min/max	Low Level min/max	High/Low accuracy	AMPLITUDE min/max
A SEP B 50 Ω	50 Ω/25 pF	−9.89 V/ +9.99 V	−9.99 V/ +9.89 V	1%±1% of amplitude ±50 mV	100 mV/ 9.99 V
A SEP B 1 kΩ	1 kΩ 25 pF	−19.7 V/ +19.9 V	−19.9 V/ +19.7 V	1%±1% of amplitude ±100 mV	200 mV/ 19.9 V
A ADD B (Opt 020) 50 Ω	48 Ω/60 pF	−9.89 V/ +9.99 V	−9.99 V/ +9.89 V	2×(A SEP B, 50 Ω) −2.5%	100 mV/ 19.5 V
A ADD B (Opt 020) 1 kΩ	500 Ω/60 pF	−19.7 V/ −19.9 V	−19.9 V/ +19.7 V	2×(A SEP B, 1 kΩ) −5%	200 mV/ 20.0 V

OUTPUT LEVEL RESOLUTION: 3 digits (10 mV).
PRESHOOT, OVERSHOOT, RINGING: 5% of amplitude ±10 mV.
SETTLING TIME: 40 ns to specified accuracy.

REPEATABILITY (all parameters): 50% of specified accuracy.

A ADD B: Adds Channel A and B outputs (option 020).
OUTPUT FORMAT: Normal or complement.
OPERATING MODES
NORM: Continuous pulse stream.
GATE: External signal enables rate generator. First output pulse is synced with leading edge. Last pulse is always complete.
TRIG: Each input cycle generates a single output pulse.
BURST: Each input cycle generates a programmable number of pulses (0 to 9999). Minimum time between bursts is 1 period.
MAN: Simulates external signal when EXT INPUT is switched OFF.
SINGLE PULSE: Provides a single pulse independent of input and period settings.

EXTERNAL INPUT
TRIGGER LEVEL: +10 to −10 V.
MAXIMUM INPUT: ±12 V into 50Ω, ±20 V into 10 kΩ.
MINIMUM AMPLITUDE: 500 mVpp.
SLOPE: Positive or negative.
MINIMUM PULSE WIDTH: 3 ns.
TYPICAL INPUT RESISTANCE: 50 Ω or (also in OFF) 10 kΩ.
DELAY FROM TRIGGER INPUT TO TRIGGER OUTPUT: 90 ns ±10 ns.

TRIGGER OUTPUT
AMPLITUDE: ≥2.5 V into 50 Ω, ≥5 V into open circuit.
TYPICAL SOURCE RESISTANCE: 50 Ω.
TYPICAL PULSE WIDTH: 8 ns (period <100 ns),
40 ns (100 ns <period<1 μs),
400 ns (period≥1 μs).

HP-IB CAPABILITY

Code	Interface Function
SH1	Source Handshake Capability.
AH 1	Acceptor Handshake Capability.
T6	Talker (basic talker, serial poll, unaddress to talk if addressed to listen).
L4	Listener (basic listener, unaddress to listen if addressed to talk).
SR1	Service Request Capability.
RL1	Remote/Local Capability (including Local Lockout to prevent interference with a running program).
PP0	No Parallel Poll Capability.
DC0	No Device Clear Capability.
DT1	Device Trigger Capability (Trig, Burst modes).
C0	No Controller Capability.

All modes and parameters can be programmed. EXT SLOPE POS/NEG programming can simulate Gate mode. TRIG LEVEL adjustment and 50Ω/10kΩ/OFF switch are not programmable.

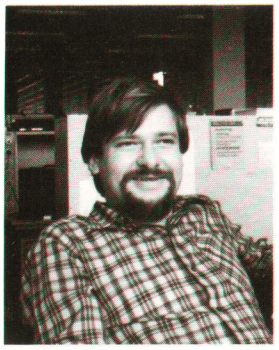
PROGRAMMING TIMES (typical)
PERIOD, DELAY, DOUBLE PULSE SPACING, WIDTH: 140 ms.
TRANSITION TIMES: 110 ms.
OUTPUT LEVELS: 150 ms.
BURST, INPUT MODES: 100 ms (existing burst will be interrupted when programming new burst).
OUTPUT MODES: 200 ms.
LISTENER DATA TRANSFER TIMES (typical)
INPUT MODES: 50 ms
PARAMETERS: 90 to 140 ms
OUTPUT MODES: 5 ms (EN/DISABLE,NORM/COMPL),
70 ms (A SEP/ADD B, 50 Ω/1 kΩ).
DEVICE TRIGGER: 30 ms (EXT TRIG), 80 ms (BURST).
TALKER DATA TRANSFER TIMES
STATUS: 1 byte (indicates nature of programming error), <5 ms typical.
LEARN: 11 lines (18 in option 020) up to 14 characters plus CRLF, 10 ms/line av.
MEMORY: 9 addressable locations plus one for existing operating state.
CAPACITY: 1 complete operating state per location.
ACCESS TIME: <20 ms (store). <1.2s (recall).
POWER-OFF STORAGE: Batteries maintain all stored data for up to 2 weeks with instrument switched off. Hardwired addressable location contains a fixed operating state for confidence check (standard parameter set).

POWER: 115/230V + 10%, −22%; 48-66 Hz; 675 VA max.
ENVIRONMENTAL
TEMPERATURE RANGE: 15−35°C for operation within specifications.
ACCURACY DERATING FACTORS: 0−15°C or 35−50°C:
DELAY, WIDTH, DOUBLE PULSE: 0.07%/°C
PERIOD, HIGH LEVEL, LOW LEVEL: 0.14%/°C
LEADING EDGE, TRAILING EDGE: 0.21%/°C
WEIGHT: 20.8 kg (46 lbs).
DIMENSIONS: 178 mm H, 426 mm W, 430 mm D (7×16.8×17 in).
OPTIONS
001: Rear Panel Input and Outputs
020: Second Channel, Includes delay, width, double pulse, transition times, and output amplifier.
PRICE IN U.S.A.: $11,000; Opt 001, no charge; Opt 020, $5160.
MANUFACTURING DIVISION: Böblingen Instrument Division
Herrenbergerstrasse 110
D-703 Böblingen, Germany

and accomplished the task of getting it into a small enclosure. Rolf Hoffman implemented the output amplifier, including the output control card, as well as the series regulator. Special thanks are also due to Rainer Eggert who did the mechanical design, and to Dieter Kible who wrote the HP-IB software. Many helpful ideas were provided by lab section leader Reinhard Falke and product manager Robin Adler.

Reference

1. T. Schad, D. Kible, and P. Brünner, "1-mHz-to-50-MHz Signal Source Combines Synthesizer Accuracy, Multimode Operation, and Easy Programming," Hewlett-Packard Journal, December 1978.



Werner Hüttemann

A Diplom-Ingenieur graduate of the University of Aachen, Werner Hüttemann joined HP's Böblingen Instrument Division in 1973. Werner contributed to the design of the delay generator and slope generator of the 8160A before becoming project leader of the instrument. In his spare time, he enjoys photography, table tennis, and playing the piano.

Lutz Kristen

A native of Berlin, Lutz Kristen joined HP's Böblingen Instrument Division shortly after receiving his Diplom-Ingenieur from the Technische Universitat Karlsruhe. Lutz contributed to the design of the digital control and wrote the software for the 8160A Pulse Generator. Lutz and his wife recently moved into a new house and spend much of their spare time doing the finishing work on it. Lutz also enjoys hiking and working on his digitally-controlled FM tuner.

Peter Aue

Peter Aue joined HP's Böblingen Instrument Division in 1976 shortly after receiving his Diplom-Ingenieur from the University of Stuttgart. Peter was responsible for the design of the repetition rate generator and burst generator for the 8160A. Peter is married, has two daughters (ages four and one), and when he's not working on his new home, he spends his spare time doing woodworking, taking photographs, and building model railroads.

Extending Possibilities in Desktop Computing

This midrange computer's large memory capacity, two languages (enhanced BASIC and assembly language), low radiated interference, and powerful input/output facilities suit it especially well for computation, control, and data acquisition applications.

by Sandy L. Chumbley

IN THE TRADITION OF DESKTOP COMPUTERS, HP's new Series 9800 System 35, or Model 9835A/B (Fig. 1), places as much computational power as possible into a small, integrated package and makes it extremely easy and natural to use. This new desktop computer features the largest memory capacity in its class, plus assembly language programming capability and enhanced BASIC.

The 9835A/B is a midrange, large-memory, scientific and engineering desktop computer designed for computation, control, and data acquisition applications. It features expanded read/write memory capacity of up to 256K bytes, unified mass storage, a tape cartridge directory in read/write memory, and a "bad memory" error detect message system. High-speed control and data acquisition applications are facilitated by standard, plug-in interface cards, direct memory access, and 15-level interrupt. The interface cards include the Hewlett-Packard Interface Bus (HP-IB), 16-bit parallel, RS-232C, and BCD. A real-time clock interface adds real-time reference and time-related control capabilities.

As is characteristic of most desktop computers, many peripherals have been integrated into System 35, including interactive keyboard, alphanumeric display (a 24-line CRT for the 9835A, a single-line display for the 9835B), and an internal cartridge tape drive with a capacity of 217K bytes per tape. An optional 16-character thermal strip printer is also available for users who require low-cost permanent copy for such applications as data logging or program debugging.

The 9835A/B is similar in hardware design to Model 9825A.[1] In language and performance, however, it resembles System 45 (Model 9845A).[2] It provides most of the high-level capability of System 45 and adds significant new capabilities. It extends and allows for further extension of memory, it provides a low-level language with a refreshing ease of use, and it meets new standards for electromagnetic interference.

High-Level Capability

System 45 was notable for its enhanced BASIC language and ease of use.[2] System 35 uses essentially the same firmware and thus provides most of the same capability. For example, it supports enhanced BASIC with 15-character

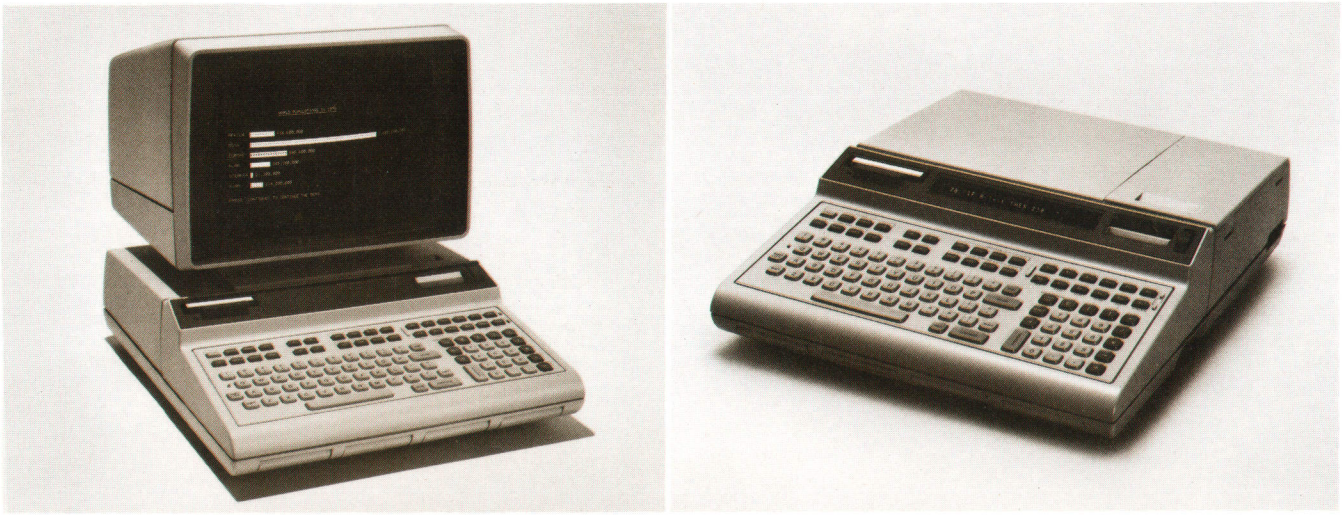


Fig. 1. *Model 9835A/B Desktop Computer's large memory capacity and powerful I/O system make it especially well suited for control and data acquisition applications in addition to scientific computation. Model 9835A has a 24-line CRT display. Model 9835B has a single-line display.*

variables, labeled GOTOs, extended string and array capability, subprograms with local environments, and so on. It also has the same interactive CRT, unified mass storage, and PRINT USING capabilities. It does not support the CRT graphics option of System 45.

The use of HP enhanced BASIC on both System 35 and System 45 greatly simplifies the exchange of data and programs between the two machines. Because of their common language, Systems 35 and 45 share an extensive library. Programs available for the System 35 include a utility pack (with plotter graphics), basic statistics and data manipulation, regression analysis, numerical analysis, nonlinear regression and statistical plotter graphics.

Enhanced BASIC on the dual-processor System 45 was converted to System 35 by running the language processor unit[3] firmware and peripheral processor unit[3] firmware in series on System 35's single processor. In many cases the speed penalty is minimal.

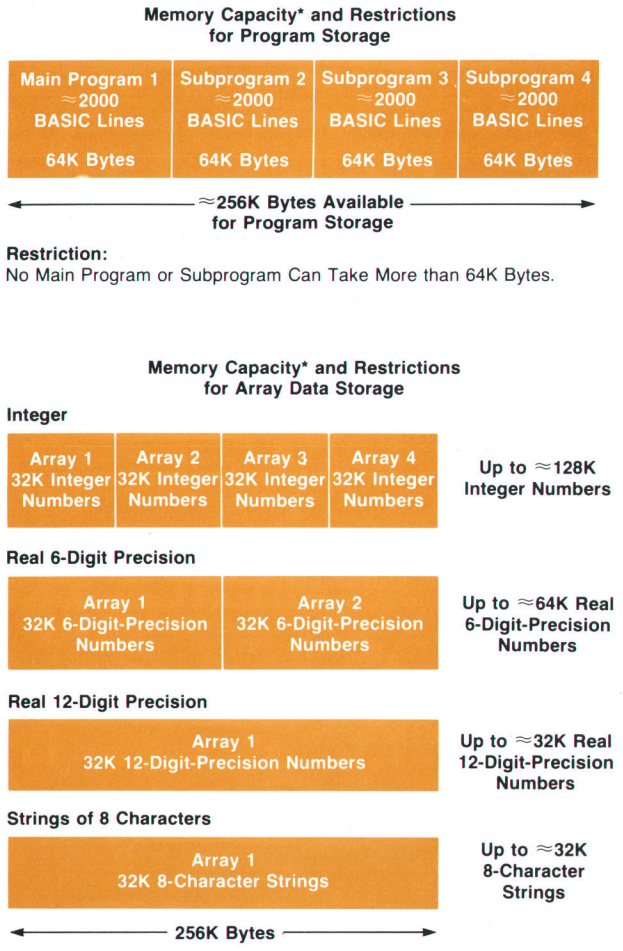


Fig. 2. *Model 9835A/B can have up to 256K bytes of read/write memory and 240K bytes of read-only memory. The read/write memory can be used for programs or data as shown.*

Memory Extension

A significant extension to the high-level capabilities of the 9835A/B has been the expansion of read/write memory (RAM) up to 256K bytes and system read-only memory (ROM) up to 240K bytes. The read/write memory space can be used for program or data as shown in Fig. 2.

Low-Level Language

Assembly language is provided in the 9835A/B to obtain additional speed for critical routines. These routines can be written, edited, debugged, and run much as a BASIC-language subprogram would be.

Assembly language was chosen as a second language because it allows the experienced user to get at the absolute maximum speed of the machine, and because this language is still a very good fit for routines requiring bit manipulations, I/O drivers, and various user-required primitive operations.

Although assembly language programming is available on many computer systems, the 9835A/B is thought to be the first desktop computer to offer it. A real contribution over virtually every other assembly language system is the 9835A/B's ease of use. It is truly refreshing to have the computer work for you by checking syntax on entry, allowing easy and immediate edits, giving clear run time error messages, and providing single-step, breakpoint, and other debug facilities.

New Standards for Electromagnetic Interference

It was considered important that System 35 meet new standards for electromagnetic interference. We had seen real needs in several applications. Government regulations were being enforced more rigidly and were becoming more severe. European users in particular demanded it. As a result, System 35's radiated interference levels are approximately 20 dB lower than those of its predecessors, enabling it to meet the standard of VDE radiated interference level A (see article, page 16).

Sandy L. Chumbley

Sandy Chumbley was project manager for 9835A/B Desktop Computer hardware. With HP since 1965, he has served as production test technician, lab technician, design engineer for 9821A software and the NMOS II processor chip set, and project leader for the 9885A flexible disc drive. He's named inventor on two patents related to the NMOS II and 9885A projects. Sandy was born in Louisville, Kentucky. In 1972 he received his BSEE degree from Colorado State University after four years of full-time school combined with a full-time career. He's married, has five children, and enjoys life in Colorado, including backpacking, cross-country skiing and running, tennis, and swimming. He also does woodworking and is active in religious education.

SPECIFICATIONS

HP Model 9835A Desktop Computer

DYNAMIC RANGE: -10^{99} to -10^{-99}, 0, $+10^{-99}$ to $+10^{99}$.
INTERNAL CALCULATION RANGE: -10^{511} to -10^{-511}, 0, 10^{-511} to 10^{511}
SYSTEM 35 READ/WRITE MEMORY
STANDARD: 49,962 bytes
OPT. 201: 115,402 bytes
OPT. 202: 180,842 bytes
OPT. 203: 246,282 bytes
The standard read/write memory contains 65,536 bytes, 49,962 directly available to the user.
TAPE CARTRIDGE
CAPACITY: 217K bytes.
ACCESS: Directory, file-by-name.
SEARCH SPEED (bidirectional): 2286 mm/s (90 in/s).
AVERAGE TRANSFER RATE: 1480 bytes/s.
CARTRIDGE SIZE: 63.5 × 82.5 × 12.7 mm (2.5 × 3.25 × 0.5 in).
CRT
SCREEN SIZE: 261 × 193 mm (10.3 × 7.6 in). 310-mm (12.2-in) diagonal.
SCREEN BRIGHTNESS: manually adjustable 12-30 ft-lamberts.
REFRESH RATE: 60 Hz (independent of line frequency).
TUBE PHOSPHOR: P31.
SCREEN CAPACITY: 25 lines × 80 characters (2000 characters).
RASTER SCAN SIZE: 215 × 135 mm (8.48 × 5.3 in).
CHARACTER GENERATION: 7 × 9 character font in a 9 × 15 character cell.
STANDARD CHARACTER SET: 128 ASCII characters.
OPTIONAL CHARACTER SETS: French, Spanish and German.
CURSOR: Blinking underline.
OPERATING TEMPERATURE: 5°C to 40°C (ambient).
STORAGE TEMPERATURE: −40°C to +65°C.
RELATIVE HUMIDITY: 5% to 80% at 40°C.
THERMAL LINE PRINTER
PRINT SPEED: up to 190 lines/min.
PAPER WIDTH: 57 mm (2.25 in), 16 characters/line.
SIZE: HWD 376 × 384 × 495 mm (14.8 × 15.1 × 19.5 in).
WEIGHT: 11.8 kg (26 lb).
PRICE IN U.S.A.: 9835A base price, $9,900.
MANUFACTURING DIVISION: DESKTOP COMPUTER DIVISION
3404 E. Harmony Road
Ft. Collins, Colorado 80525 U.S.A.

Acknowledgments

Much of the concept of the product is owed to Tom Haswell. Doug Clifford and Ed Muns made this highly-leveraged system happen, wisely overcoming the tendency to invent the new rather than add to what had come before. Jack Cooley, Dan Oseky and Tom Lane made the mainframe code conversion. Bob Hallissy, Sam Sands and Paul Morrison designed the refreshing assembly addition. Bob Tinnen, Bill Boles, and John Becker provided much of the design of the hardware, and were very innovative in leveraging what had come before. Damon Ujvarosy, Bob Hallissy, and Dyke Shaffer made the LSI extensions look easy and set some new standards with chips and hybrids going into production with first masks. There are many others, too numerous to mention here, who contributed greatly.

References

1. D.E. Morris, C.J. Christopher, G.W. Chance, and D.B. Barney, "Third-Generation Programmable Calculator Has Computer-Like Capabilities," Hewlett-Packard Journal, June 1976.
2. W.D. Eads and J.M. Walden, "A Highly Integrated Desktop Computer System," Hewlett-Packard Journal, April 1978.
3. J.C. Keith, A.K. Vogen, and L.T. Schulte, "System 45 Hardware Design," Hewlett-Packard Journal, April 1978.

Processor Enhancements Expand Memory

by Damon R. Ujvarosy and Dyke T. Shaffer

THE 9835A/B DESKTOP COMPUTER offers the user very large memories, formerly available only in big computers. This large-memory capability was achieved by using 16K dynamic RAMs and 64K ROMs and by adding a new NMOS II[1] chip to the existing BPC processor chip set.[2]

A major objective of the 9835A/B was to maintain language compatibility with the 9845A Desktop Computer, also known as System 45.[3] This objective was realized by adapting the System 45 operating system for use in the 9835A/B. Two major hardware challenges had to be overcome to minimize the modifications. The first was that System 45 has two processors and the 9835A/B has only one. The second was that memory extension already existed in System 45,[4] but did not allow for read/write memory expansion beyond 64K bytes.

The single processor of the 9835A/B is used as if it were two pseudo-processors. Although the two pseudo-processors cannot run simultaneously as the two processors of the System 45 do, they are separate and distinct entities within the software. Each pseudo-processor needs to have its own home block of ROM and base-page read/write area (upper 512 words of address space). The new address extension chip makes this possible.

Address Extension Chip

The address extension chip (AEC) expands upon the memory address extension scheme of System 45 by adding three registers to the three already defined in System 45 and extending the length of all six registers from two bits to 16 bits. Fig 1 is a microphotograph of the AEC.

The fetch and subsequent execution of an instruction

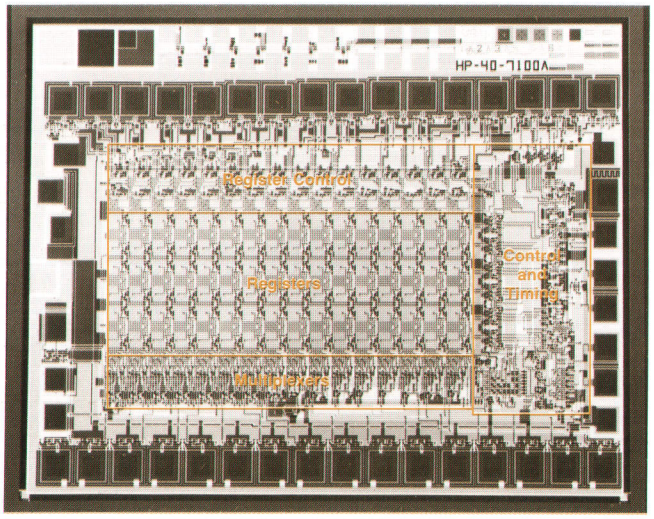


Fig. 1. *The address extension chip was added to the System 35 processor chip set to provide a means of managing larger memories. The AEC adds registers to those defined in System 45 and extends these registers to 16 bits. This provides more flexibility and extends the addressing capability to 2^{16} = 65,536 blocks of 64K bytes each.*

require one or more memory accesses. These memory accesses are divided into two groups: those that generally access ROM, which are accesses of instruction space, and those that generally access read/write memory, which are accesses of data space. Memory is subdivided into 64K-byte blocks, and the processor can directly address 128K bytes. Two blocks of memory can be accessed in each space, instruction and data. This makes four different blocks that can be directly accessed by the processor, as shown in Fig. 2.

Six 16-bit registers are provided in the AEC. Each can be loaded with a block select code (BSC) to indicate which block of memory should be enabled for each memory access. Four of the registers define the upper and lower instruction and data space blocks. The fifth defines the base-page read/write block and the sixth is used whenever DMA

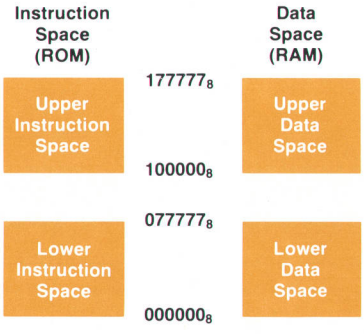


Fig. 2. *The processor can directly address two 64K-byte blocks of read-only memory and two 64K-byte blocks of random-access (read/write) memory. Each of these four blocks has a corresponding 16-bit register on the address extension chip. The contents of the register corresponding to a particular block defines which of the 65,536 possible blocks is to be used as that block. Two other AEC registers are for base page read/write and direct memory access.*

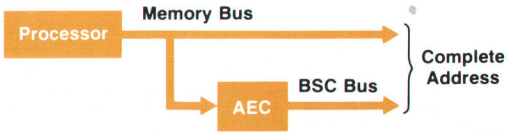


Fig. 3. *At the start of each memory access the address extension chip places the contents of one of its six registers on the BSC bus. This block select code and the address from the processor form the complete address.*

is taking place. Since each block is 64K bytes long and the BSC is 16 bits long, the total addressing capability is extended to 2^{16} = 65,536 blocks, with a capacity of $2^{16}\times$64K bytes or four gigabytes.

The contents of one of these six registers must be placed on the BSC bus at the start of each memory access along with the address from the processor to form the complete address, as shown in Fig. 3. By placing the proper register's contents on the BSC bus, any block can be defined as the home block (lower instruction block), working block, or base-page read/write block. In general, which register's contents is put on the BSC bus is dependent on:

1. Type of memory access, that is, processor instruction fetch, processor memory access, or DMA.
2. The instruction being executed.
3. Which memory access within the execution of the instruction this is.
4. The address.

Information regarding the type of memory access is available before the start of the memory access. The instruction being executed can be decoded during the instruction fetch. The AEC keeps track of the instruction execution by counting the number of memory accesses that have taken place before the current access. All of this information influences which register's contents will be placed on the BSC bus and is available before the start of the memory access. The address that the processor puts out is also used to determine which register's contents is placed on the BSC bus.

Since the AEC cannot put the correct BSC on the bus until

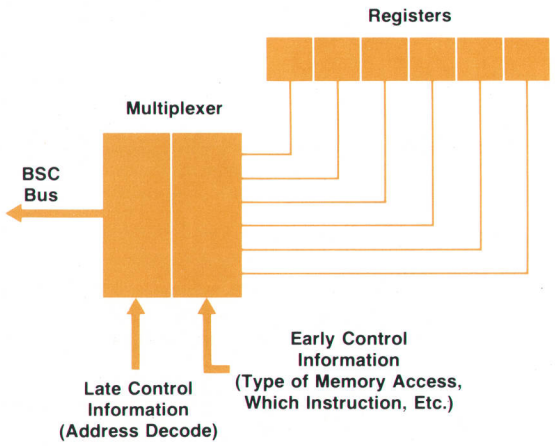


Fig. 4. *Multiplexer speeds memory block selection by making use of early information to reduce the number of possible block select codes to three or fewer.*

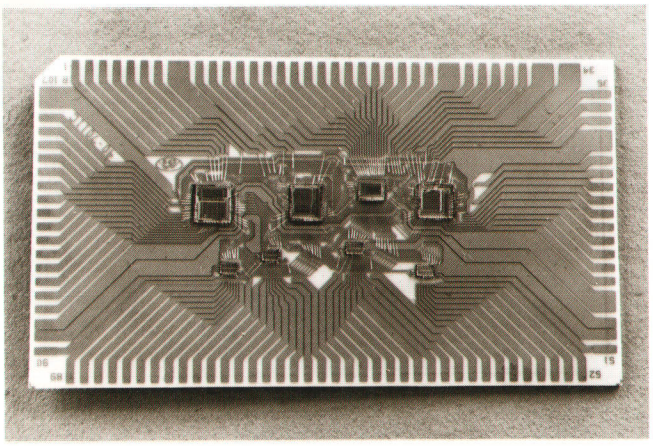


Fig. 5. *Existing 9800 Series processor hybrid was modified to accommodate the new address extension chip.*

the address from the processor is valid and the total address is not valid until the BSC is valid, the time from the instant the address is valid until the BSC bus is valid is critical. To minimize the time necessary to get the BSC bus valid, a parallel register structure combined with a two-section multiplexer is used, as shown in Fig. 4. The contents of all six registers are available to the multiplexer all the time. The first section of the multiplexer uses the information that is available before the start of the memory access to reduce the number of BSCs possible to at most three. The second section of the multiplexer uses the address to make the final selection of which BSC is actually placed on the BSC bus.

Hybrid

The natural place for the AEC to reside was within the hybrid package already containing the BPC processor chip set (Fig. 5).

A new metallization mask was laid out for the 7.6×4.3-cm ceramic substrate already in use in the 9825A and 9845A Desktop Computers. Two major obstacles were overcome to package the new chip. First, room had to be made on an already crowded substrate to provide the necessary bonds to connect the AEC into the processor chip set. Reducing the substrate layout design rules to 120-μm minimum conductor width and 80-μm minimum space between conductors provided the flexibility necessary to add the new IC.

Second, a means of increasing the pinout by 25 signals had to be developed. An assymetric pad arrangement at the periphery of the substrate was designed using the existing conductive elastomer gasket. All signals were brought out on pads 1 mm wide on 1.8-mm centers. Power supply connections were made via interspersed pads 2.8 mm wide. Thus the number of pads was increased to 107 to include the AEC while minimizing the impact to the production procedure by merely changing the substrate metallization mask. Sensitivity to misalignment of the substrate and the printed circuit board that occurs during normal production remains the same as with previous designs since the distance between pads was not changed from the 0.8-mm spacing used previously.

Acknowledgments

Of course no IC can be designed without the help of many people. Our sincere thanks go to all who helped. Special thanks go to Bob Hallissy and Bill Thayer who contributed to the definition and development of the AEC.

References

1. J.E. DeWeese and T.R. Ligon, "An NMOS Process for High-Performance LSI Circuits," Hewlett-Packard Journal, November 1977.
2. W.D. Eads and D.S. Maitland, "High-Performance NMOS LSI Processor," Hewlett-Packard Journal, June 1976.
3. W.D. Eads and J.M. Walden, "A Highly Integrated Desktop Computer System," Hewlett-Packard Journal, April 1978.
4. J.C. Keith, L.T. Schulte, and A.K. Vogen, "System 45 Hardware Design," Hewlett-Packard Journal, April 1978.

Damon R. Ujvarosy

Damon Ujvarosy helped design the address extension chip for the 9835A/B Desktop Computer. He's been with HP since 1976, when he received his BSEE degree from the University of Wisconsin. Now attending Colorado State University, he expects to complete his MSEE studies this year. A native of the state of Wisconsin, Damon is single and lives in Loveland, Colorado. He's an amateur radio operator and is active in church affairs.

Dyke T. Shaffer

Dyke Shaffer did the substrate design and final test hardware for the processor hybrids of the 9825/35/45 Desktop Computers. Born in West Virginia, Dyke spent most of his early years in California, and graduated from California State Polytechnic University at San Luis Obispo in 1974 with a BSEE degree. He joined HP in 1976 after designing power supplies and energy conversion systems for two years. Dyke's major interest is horsepower in two forms: high-performance auto engines and endurance riding. He also enjoys basketball, bicycling, and ice skating. He's single and lives in Loveland, Colorado.

Designing to Meet Electromagnetic Interference Requirements

by John C. Becker

Electromagnetic pollution may be defined as the effects of electromagnetic interference (EMI) produced by man-made apparatus. The seriousness of this interference ranges from annoying interference that affects a radio or television channel to interference that causes failure of an important communication channel or a cardiac pacemaker.

Electronic computers generate electromagnetic radiation that may range up to 1 GHz and beyond. This radiation is caused by alternating currents and voltages present in the computer hardware. The spectral content of these signals consists of both harmonic and broadband products.

The harmonic components are related to the system clock and other periodic system waveforms. Although the amplitudes of the harmonics in a given waveform generally decrease monotonically, factors relating to radiation efficiency, hardware, and packaging may allow significant radiation of frequencies as high as one hundred times the fundamental frequency component.

Broadband noise is related to transition time, or more specifically, to single-event occurrences. Radiation occurs when these currents and voltages exist in antenna-like loops and nodes. A simplified model that illustrates radiation of both electric and magnetic fields is shown in Fig. 1.

Regulations

Various organizations and countries have recommendations and laws establishing limits on the levels at which computers may radiate. MIL-STD-461[1] is a military document of the United States Government that is used as a control standard, primarily for procurement of military equipment. The Federal Communications Commission of the United States currently has a proposal specifying that the electromagnetic field at a distance of $\lambda/2\pi$ shall not exceed 15 μV/m, where λ is the wavelength of the signal. At present, the FCC regulation regarding computers is not very specific, merely prohibiting "harmful" interference. Comité International Spécial des Perturbation Radioélectriques (CISPR), which operates under the auspices of the International Electrotechnical Commission (IEC), is an international organization that seeks to establish international agreement on EMI limits. CISPR publication 11[2] has limits on industrial equipment covering frequencies between 0.15 MHz and 18 GHz. The limits are voluntary and have no legal status.

A West German law assigns the German Postal Service to control interference levels. VDE (Verband Deutsches Elektrotechniker, Association of German Electrical Engineers) is an organization that writes and publishes regulations and tests for compliance. These regulations are typically similar to CISPR regulations and are used by many European countries. The interference field strength limits for VDE 0871/3.68[3] are shown in Fig. 2.

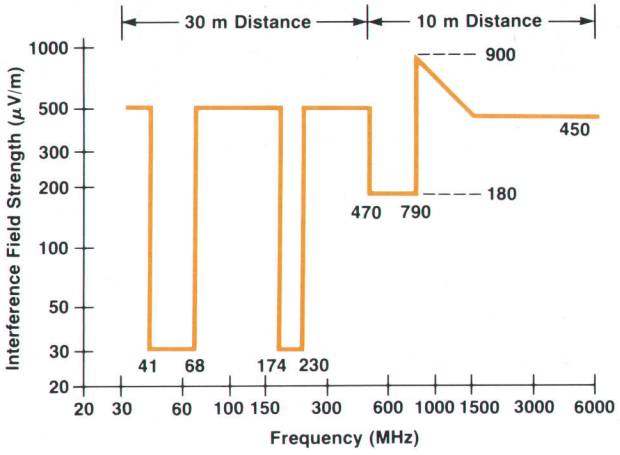


Fig. 2. *VDE interference field strength limits and distance of measurement.*

The frequency range between 30 and 470 MHz is measured at a distance of 30 meters in the far fields. Far fields[4] occur when the distance from the source exceeds $\lambda/2\pi$. The far-field wave impedance is 377Ω, while the impedance of near fields will vary significantly depending upon the actual distance from the source and whether the predominant field is electric or magnetic. The test range used by VDE in Offenbach, West Germany is illustrated in Fig. 3.

Measurement Accuracy

Accuracy is difficult to determine for radiated electromagnetic measurements. Many measurement uncertainties exist. It is not uncommon to have measurement differences greater than 6 dB between two different test ranges. In particular, vertically polarized waves may produce large measurement differences.[5] Ideally, radiated EMI measurements would be measured in free space where fewer uncertainties would exist. Unfortunately, this is not practical, so more practical test sites such as the one depicted in Fig. 3 are used. To further complicate the issue, some regulations require measurements of near fields while others specify far-field data. It is very difficult to transform between far-field data and near-field data. The origin of the wave (magnetic or electric) is one of the variables

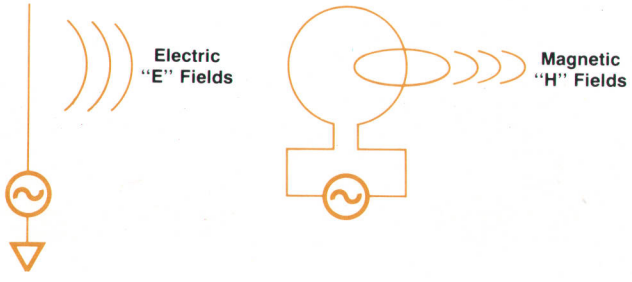


Fig. 1. *Radiation model of electric and magnetic fields.*

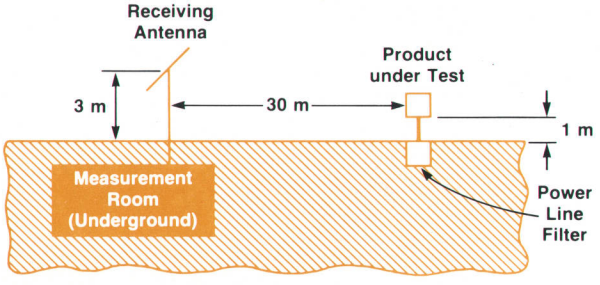


Fig. 3. *EMI test range similar to that used by VDE in Offenbach, West Germany. The surface is paved with asphalt and a wire mesh under the asphalt provides a stable reflected wave.*

required to perform this transformation. The uncertainties of EMI measurements place an additional burden upon the designer in isolating the key variables as well as in setting the design standards.

EMI Design Objectives

Our philosophy in establishing the EMI design objectives for the 9835A/B was to satisfy our worldwide customers. This led to setting objectives that covered regulatory requirements throughout the world as well as internal Hewlett-Packard EMI standards. Among these objectives were:

- Not to exceed CISPR[2] and VDE[3] radiated interference
- Not to exceed CISPR[2] and VDE[3] conducted interference limits
- To be immune to static discharges up to 15 kilovolts
- To be immune to external fields up to 1 volt/meter.

EMI Design Features of the 9835A/B

A significant amount of EMI shielding has been designed into the 9835A/B to reduce EMI. Shielding the 9835A/B presented a special problem, since the case parts are molded from structural foam polyurethane, which is nonconductive. A conductive material that surrounds the E field source is required to provide effective shielding. A material with good magnetic properties is required to shield against lower-frequency components; however, only the higher-frequency components required attenuation in the 9835A/B.

A number of methods were investigated to provide a conductive surface to the molded plastic parts, including various conductive paints, vacuum metallizing, and sprayed metals. Resisting corrosion and maintaining a low electrical resistance under prolonged severe environmental conditions were the major criteria. Zinc was selected as the primary conductive material. A method called flame spraying or arc spraying is used to apply a thin layer of zinc approximately 0.005 inch thick. A silver-filled paint is also used for some smaller parts.

A low-impedance path along the seams where the case parts join is necessary to provide an effective shield, since otherwise the seam may act as a slot antenna. In general, a slot width equal to 1/4 wavelength of a given excitation frequency may produce an effective slot antenna. Various materials for gaskets were evaluated, such as knitted mesh, mesh over elastomer, metallized fabrics, convoluted wire in silicone, and metal contact strips. A wire mesh gasket was selected because of its ability to provide a low-impedance path and extended service under adverse environmental conditions. An illustration of a metallized and gasketed case part is shown in Fig. 4. Special design consideration was also give to providing a low-impedance path between the CRT module and the mainframe. A strip of gold-plated spring fingers contacts a gold-plated bar to provide this low-impedance path. The spring fingers are attached directly to the zinc of the CRT module and the bar is connected to the zinc coating of the mainframe. A line filter that provides approximately 50 dB of attenuation between 1 MHz and 100 MHz is used to reduce the conducted radiation.

Results

All of the EMI objectives set for the 9835A/B Desktop Computer have been met. The product has passed the legal requirements of VDE for both conducted and radiated electromagnetic interference. VDE tested the 9835A both separately and as a system. The system consisted of a 9835A, a 9885M Flexible Disk Drive, a 9866B Thermal Printer, and a 9872A Plotter. The 9835A/B has passed other internal specifications such as static discharge to 15 kV, line transient tests, and susceptibility to EMI fields.

Acknowledgments

David Anderton provided significant contributions to solving the EMI problems of the 9835A/B. Wade Clowes implemented new production shielding processes. Thanks to the people of the customer assurance department, who provided valuable testing assistance.

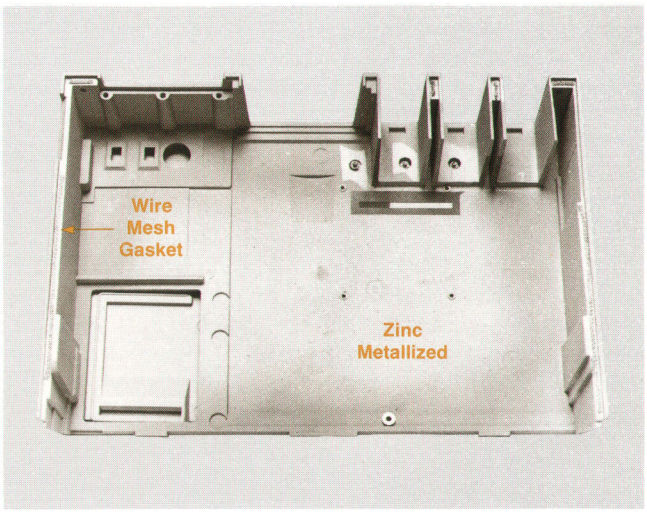
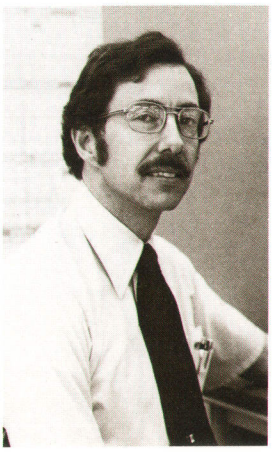


Fig. 4. *A metallized and gasketed case part used in the 9835A/B.*

References:

1. U.S. Department of Defense, MIL-STD-461A (1968), "Electromagnetic Interference Characteristics Requirements for Equipment," Dept. of Defense, Washington, D.C.
2. CISPR Publication 11, (1975), "Limits and Methods of Measurements of Radio Interference Characteristics of Industrial, Scientific and Medical (ISM) Radio-Frequency Equipment," International Electrotechnical Commission, Geneva, Switzerland.
3. VDE 0871/3.68 (1976), "Regulation for the Radio Frequency Interference Suppression of High Frequency Apparatus and Installations," VDE Publishing House (VDE Verlag), Offenbach, West Germany. (English Translation: EMACO, EMC Consultants, San Diego, California, USA.)
4. Ott, Henry (1976), "Noise Reduction Techniques in Electronic Systems," John Wiley and Sons, New York.
5. W.S. Bennett and H.E. Taggart, "Characterization of a CISPR/VDE far field EMI test site with ground screen," Proceeding of the 3rd European Symposium and Technical Exhibition on Electromagnetic Compatibility, Rotterdam, May 1-3, 1979 (To be published).



John C. Becker

Now production engineering manager for desktop controllers, John Becker was lead engineer for the 9835A/B Desktop Computer CRT and EMI designs. With HP since 1964, he's also contributed to the design of the 9825A Desktop Computer. John received his BSEE and MSEE degrees in 1972 and 1974 from Colorado State University. A registered professional engineer in Colorado, he's author of a paper on ROM-based multiplication and is named inventor on a reed relay patent. Born in Lincoln, Nebraska, he's married, has a family, and lives in Loveland, Colorado. His leisure pursuits include tennis, skiing, and travel.

Assembly Programming Capability in a Desktop Computer

by Robert M. Hallissy

THE ASSEMBLY PROGRAMMING OPTION for the HP 9835A/B Desktop Computer represents a significant step in the evolution of desktop computers. Integrated into a single system is the friendliness of an interpreted high-level language and the power, speed, and flexibility of direct assembly language programming.

Traditional descriptions applied to interpretive systems contain such phrases as friendly, easy to learn, high-level language, easy to program, slow to execute. HP desktop computers also exhibit such properties as transparent compiling, built-in editors, and high-level debug tools. Traditional assembly language systems tend towards the converse of all of the above qualities. It was our objective to implement an extension to the 9835A/B that would provide access to the raw power of the processor and yet eliminate the negative aspects of assembly language programming.

The phrase "friendly assembly language system" seemed to be so paradoxical that the initial investigation was expanded to include languages other than assembly. We found that compiled high-level languages such as BASIC and FORTRAN did not allow the programmer direct access to the processor hardware. The system programming language used to develop the system firmware was a machine-dependent language only one step removed from assembly. Our internal experience with this SPL proved that its intricacies were almost as unfriendly as straight assembly language. Therefore, assembly language was chosen as the second language for the 9835A.

The project team then set about designing a friendly assembly language system. Ideas were taken from existing desktop computers and from colleagues doing assembly language programming. The following sections describe some of the features of the resultant package.

Source Entry

As can be seen in Fig. 1, the assembly source code for the 9835A/B looks similar to that of other assemblers, that is, label followed by mnemonic, followed by expression, followed by comment. Source lines on the 9835A/B are typed in using the same EDIT capability provided by 9835A/B BASIC. The keyword ISOURCE tells the syntaxer to treat everything that follows as an assembly source statement. This allows the syntaxing to be performed when the line is entered, providing immediate feedback to the user concerning typing errors. Also, the symbol table structure needed for the assembly process is set up when the line is entered, eliminating symbol searching and sorting at assembly time. Another aid to source entry is the space-independent characteristic of the syntaxer. Spaces can be inserted anywhere in the ISOURCE line and all characters can be in either upper or lower case. The syntaxer automatically converts labels to standard form (an upper-case letter followed by lower-case letters) and mnemonics to upper case.

Finally, when a source line is successfully entered into the computer, it is converted to an internal form optimized for the assembly process and stored in the read/write memory of the computer.

Assembler/Linker

The assembler, which generates machine instructions from the source lines stored in memory, is invoked by the IASSEMBLE statement. While this step is traditionally a time-consuming task, the 9835A/B's assembler is extremely fast, assembling source code at a rate greater than 800 lines per second. This speed is a result of the presyntaxed source code and the fact that both source code and object code (the output of the assembler) are resident in read/write memory instead of on a disc or tape device. Assembly is so rapid, in fact, that there is no speed penalty in reassembling the source program each time a program is run. Thus, the traditional debug mode of modifying memory cells (patching) is virtually eliminated. The user simply modifies the source program and presses the RUN key.

High speed is only one of the features of the built-in assembler. Others include completely relocatable object code, conditional assembly, literals, intermodule linkage, and auto indirect addressing. The latter feature allows users to forget about the inherent addressing mode of the processor. Standard memory reference instructions, for example, have a 10-bit address field, allowing direct addressing of the 1024 locations surrounding the address of the instruction. With our system, if a user references a location more than 512 words away, the assembler automatically implements indirect addressing through a link placed in a literal pool.

Linking object modules together is the final step users of traditional systems must perform before trying to execute the assembled program. The 9835A/B performs linking automatically and transparently to the user. As soon as a module is successfully assembled, it is linked with other modules in memory. Again, this process is so rapid it is not noticeable to the user.

Debugging

Traditional assembler systems provide distinctly unfriendly tools for debugging programs. Once debug mode is invoked, the user can typically establish one or more break points at memory locations within the program. The addresses of these points must be computed from the listing and a load map. When and if the program reaches a break point, it stops, and a monitor program allows the user to inspect and change memory or registers from the terminal. The user must often have a listing and load map of the program as well as instruction bit patterns for decoding or

patching instructions.

The debug tools provided by the 9835A/B are designed to minimize the length of time needed to track down bugs. For example, since all 9835A debug tools allow symbolic addressing, load maps are unnecessary. Also, when a break point is reached (there can be eight break points at a time), the user is prompted by the appearance on the CRT of the source line that generated the instruction. Thus, listings are unnecessary for debugging.

However, the feature that sets the 9835A/B apart from other systems is what the user can do when a break point is reached. When a break is reached, the system saves the current processor state and returns control to the BASIC interpreter. This allows users to execute BASIC statements and even execute BASIC programs or other assembly programs. When desired, the original assembly program can be resumed where it was interrupted. As illustrated in Fig. 2, the user can program complex tests in BASIC as part of a debug routine to be executed automatically each time a breakpoint is reached.

```
10   INTEGER A(1:100),N
20   ICOM 200                  ! Allocate 200 words for object code
30   RANDOMIZE -.356789411
40   FOR I=1 TO 100
50   A(I)=100*RND
60   NEXT I
70   IASSEMBLE ;LIST,XREF      ! Assemble assembly source
80   N=100
90   BEEP
100  ICALL Sort(N,A(*))        ! Execute object code; pass two parameters
110  BEEP
120  PRINT A(*)
130  END
140  !
150  ! This assembly language subprogram sorts an integer vector
160  ! in descending order
170  !
180         ISOURCE             NAM Intsort
190         ISOURCE             EXT Get_value,Get_info,Get_element
200         ISOURCE             EXT Put_element
210         ISOURCE             SUB
220         ISOURCE Arg1:       INT
230         ISOURCE Arg2:       INT (*)
240         ISOURCE Sort:       LDA =Size
250         ISOURCE             LDB =Arg1
260         ISOURCE             JSM Get_value     ! Get # of elements in vector
270         ISOURCE             LDA =Info
280         ISOURCE             LDB =Arg2
290         ISOURCE             JSM Get_info      ! Get information about vector
300         ISOURCE             LDA Size
310         ISOURCE             STA Jcount        ! Initialize J loop counter
320         ISOURCE             STA Savjcount
330         ISOURCE             ADA M1
340         ISOURCE             STA Icount        ! Initialize I loop counter
350         ISOURCE             LDA =0
360         ISOURCE             STA I
370         ISOURCE Iloop:      JSM Isetup
380         ISOURCE             JSM Get_element   ! Get (I+1) element of vector
390         ISOURCE             LDA I
400         ISOURCE             SIA ++1
410         ISOURCE             STA J             ! Initialize J loop index
420         ISOURCE             DSZ Savjcount
430         ISOURCE             LDA Savjcount
440         ISOURCE             STA Jcount        ! Set J loop counter for Ith loop
450         ISOURCE Jloop:      JSM Jsetup
460         ISOURCE             JSM Get_element   ! Get (J+1) element of vector
470         ISOURCE             LDA Jelement
480         ISOURCE             TCA
490         ISOURCE             ADA Ielement      ! Ielement minus Jelement
500         ISOURCE             SAP Dont_switch   ! Skip if in descending order
510         ISOURCE             LDA Jelement
520         ISOURCE             STA Temp
530         ISOURCE             LDA Ielement
540         ISOURCE             STA Jelement
550         ISOURCE             JSM Jsetup
560         ISOURCE             JSM Put_element   ! Put Ielement in (J+1) spot
570         ISOURCE             LDA Temp
580         ISOURCE             STA Ielement
590         ISOURCE             JSM Isetup
600         ISOURCE             JSM Put_element   ! Put Jelement in (I+1) spot
610         ISOURCE Dont_switch: ISZ J
620         ISOURCE             DSZ Jcount        ! Done with J loop?
630         ISOURCE             JMP Jloop         ! No
640         ISOURCE             ISZ I
650         ISOURCE             DSZ Icount        ! Done with I loop?
660         ISOURCE             JMP Iloop         ! No
670         ISOURCE             RET 1             ! Yes
680         ISOURCE !
690         ISOURCE Isetup:     LDA I
700         ISOURCE             STA Info+16       ! Save offset of (I+1) element
710         ISOURCE             LDA =Ielement     ! A = address of Ielement
720         ISOURCE             LDB =Info         ! B = address of Info
730         ISOURCE             RET 1
740         ISOURCE Jsetup:     LDA J
750         ISOURCE             STA Info+16       ! Save offset of (J+1) element
760         ISOURCE             LDA =Jelement     ! A = address of Jelement
770         ISOURCE             LDB =Info         ! B = address of Info
780         ISOURCE             RET 1
790         ISOURCE !
800         ISOURCE Info:       BSS 22            ! Receiving area for vector info
810         ISOURCE I:          BSS 1             ! Offset of (I+1) element
820         ISOURCE J:          BSS 1             ! Offset of (J+1) element
830         ISOURCE Ielement:   BSS 1             ! (I+1) element
840         ISOURCE Jelement:   BSS 1             ! (J+1) element
850         ISOURCE Icount:     BSS 1             ! I loop counter
860         ISOURCE Jcount:     BSS 1             ! J loop counter
870         ISOURCE Savjcount:  BSS 1             ! J loop counter save area
880         ISOURCE Size:       BSS 1             ! Size of vector
890         ISOURCE Temp:       BSS 1             ! Temporary location
900         ISOURCE M1:         DAT -1            ! -1 constant
910         ISOURCE             END Intsort
```

Fig. 1. *BASIC-language program for the 9835A/B containing an assembly language subprogram.*

```
10   IBREAK Label GOSUB Dump_routine    ! SET UP BREAK POINT
20   ICALL Test                         ! CALL THE ASSEMBLY LANGUAGE ROUTINE
30   !
40   ! ... REMAINDER OF PROGRAM
50   !
60   ! THE FOLLOWING ROUTINE GETS CALLED EACH TIME THE BREAK IS REACHED:
70 Dump_routine:   IDUMP A TO B;ASC String TO String,2  ! DUMP CONTENTS OF A AND
80                                      ! B TO PRINTER IN OCTAL; DUMP 3 WORDS
90                                      ! STARTING AT String IN ASCII CHARACTER.
100  IF IMEM(A)=0 THEN RETURN           ! IF A REG IS ZERO, RETURN IMMEDIATELY
110                                     ! TO ASSEMBLY
120  BEEP                               ! ELSE TELL OPERATOR AND
130  PAUSE                              ! WAIT FOR HIM TO PRESS CONTINUE.
140  ICHANGE A TO 100                   ! MODIFY CONTENT OF A REGISTER.
150  RETURN                             ! RETURN TO ASSEMBLY.
160  !
170  ! ...
180  !
190       ISOURCE ! ...
200       ISOURCE !
210       ISOURCE             LDA Hook
220       ISOURCE Label:      STA Temp
230       ISOURCE             CPA =0
240       ISOURCE !
250       ISOURCE ! ...
260       ISOURCE !
```

Fig. 2. *9835A/B assembly language provides powerful debugging tools. Breakpoints can be set up (as many as eight at a time) and BASIC or assembly language program statements can be executed when a breakpoint is reached. Thus complex tests can be executed automatically at breakpoints.*

Other debugging capabilities include memory modification (ICHANGE statement), successive single instruction execution for single step (manually with the STEP key or under program control using the IBREAK ALL statement), break on instruction or data, interpretive mode execution (allowing memory access violations to be detected), and multiformat dumps (octal, hexadecimal, binary, decimal, or character).

The ROM Concept

One of the reasons program development using traditional assembly language systems is so time-consuming is that each of the development steps (EDIT, ASSEMBLE, LINK, and DEBUG) typically requires loading and running a special program.

With the 9835A/B, all of those special programs are built into the assembly language development ROM and are immediately accessible by the execution of simple BASIC statements.

A second part of the ROM concept is that the assembly language system is available in two forms. The development system contains all tools necessary to create, debug, and execute assembly language programs. For the OEM whose end user only needs to run assembly language programs developed by the OEM, an execution ROM is available that provides only those capabilities needed to load, run, and store complete programs.

Applications

Assembly language inherently requires more programming effort than a high-level language. If it is used, it is used to gain some benefit in either speed or capability. In the 9835A/B, the benefit is primarily speed, since there is almost nothing that can be done in assembly language that can't be done in BASIC. Thus we can measure the increased system performance by the speed ratios of identical tasks programmed in assembly language and BASIC.

Operation	Improvement Factor
1. Single Floating Point	0.8
2. Real Array Manipulation	3.0
3. Interrupt Service Response	20.0
4. Integer Multiply	50.0
5. Most Simple Operations, e.g., Branches, Loops, Integer Manipulation	100.0

Fig. 3. *Examples of speed increases (and one decrease) using assembly language programming for particular operations.*

Fig. 3 shows the range of speed ratios that can be expected for various primitive operations. Note that single-precision floating-point arithmetic operations actually take longer in assembly language than in BASIC because the 9835A/B's BASIC interpreter is optimized for these operations. Since any particular application would use combinations of the different primitive operations, performance gains from 0.8 to 100 or more are possible.

The kinds of applications that can benefit most from the use of assembly language are those for which 9835A/B BASIC does not provide high-level statements. For example, since there is already a matrix inverse statement in BASIC, there would be no gain in coding the algorithm in assembly language. On the other hand, there is no built-in inverse function for complex-valued matrixes. A user can perform this function in BASIC using FOR-NEXT loops to index through the array. The identical algorithm coded in assembly language may decrease the time by a factor of three or more.

Other applications that can gain performance include data-dependent or computation-dependent I/O, real-time control, data formatting, integer array manipulation, and data logging.

Acknowledgments

Preliminary market studies by Brad Miller and early investigation and development by John Bidwell provided the guidelines for the development of this very friendly assembly language system. Paul Morrison served double duty as developer of the execution ROM and as product manager for the marketing aspects of the product. Sam Sands was responsible for the part of the development ROM devoted to source entry and the actual assembly process. Many thanks go to these and all the others who dedicated part of their time to this project.

Robert M. Hallissy

Bob Hallissy designed option ROMs for the 9815A and 9835A/B Desktop Computers and helped design the address extension chip for the 9835A/B. A native of Hampton, Virginia, he received his BSEE and MSEE degrees in 1974 and 1975 from Virginia Polytechnic Institute and State University. Now a product manager in controller marketing, Bob lives in Loveland, Colorado and is married to a student at Colorado State University. His interests include skiing, music, and computers.

Hewlett-Packard Company, 1501 Page Mill Road, Palo Alto, California 94304

Bulk Rate
U.S. Postage
Paid
Hewlett-Packard
Company

HEWLETT-PACKARD JOURNAL

MAY 1979 Volume 30 • Number 5

Technical Information from the Laboratories of Hewlett-Packard Company

Hewlett-Packard Company, 1501 Page Mill Road
Palo Alto, California 94304 U.S.A.
Hewlett-Packard Central Mailing Department
Van Heuven Goedhartlaan 121
Amstelveen-1134 The Netherlands
Yokogawa-Hewlett-Packard Ltd., Suginami-Ku
Tokyo 168 Japan