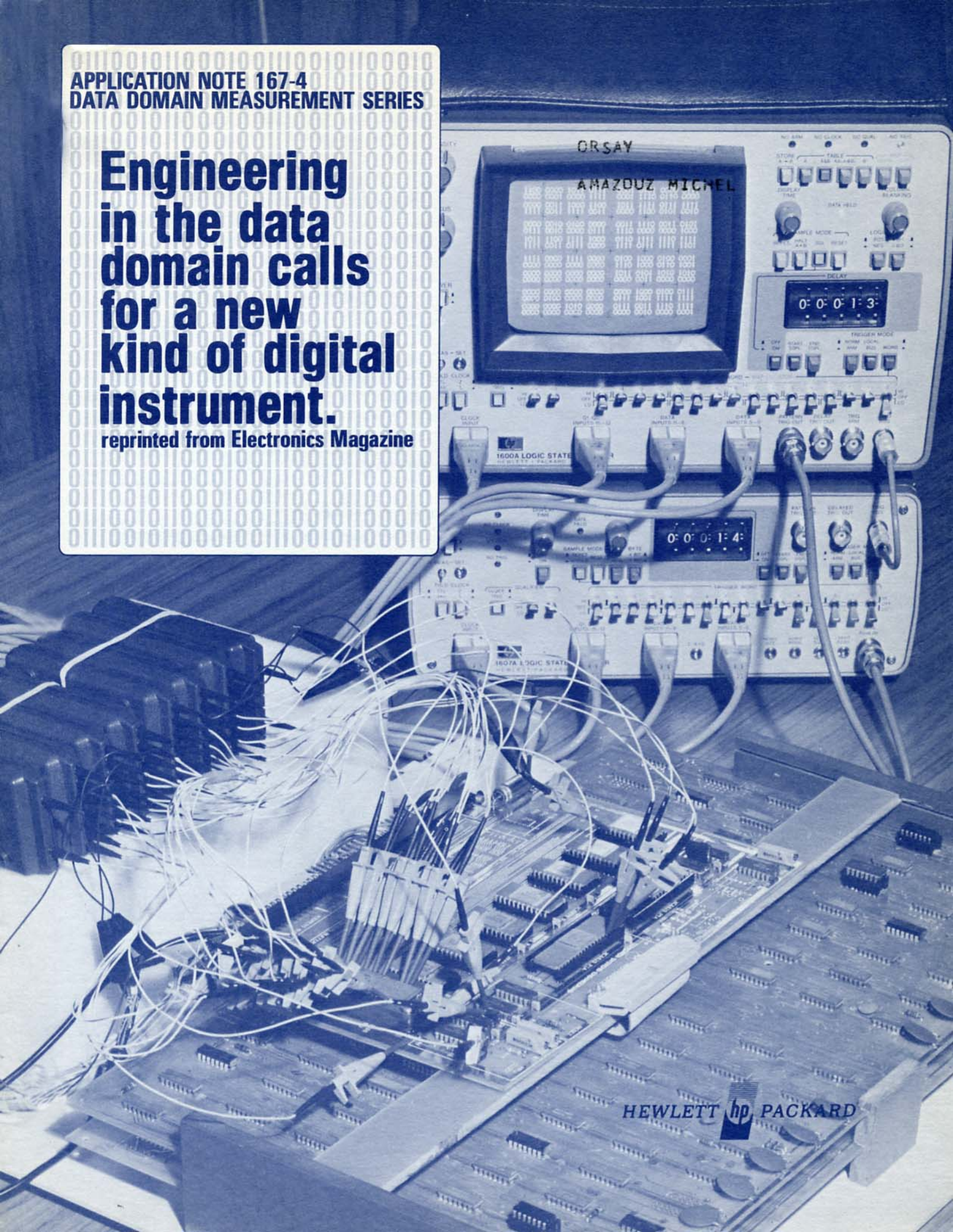
APPLICATION NOTE 167-4
DATA DOMAIN MEASUREMENT SERIES

Engineering in the data domain calls for a new kind of digital instrument.

reprinted from Electronics Magazine

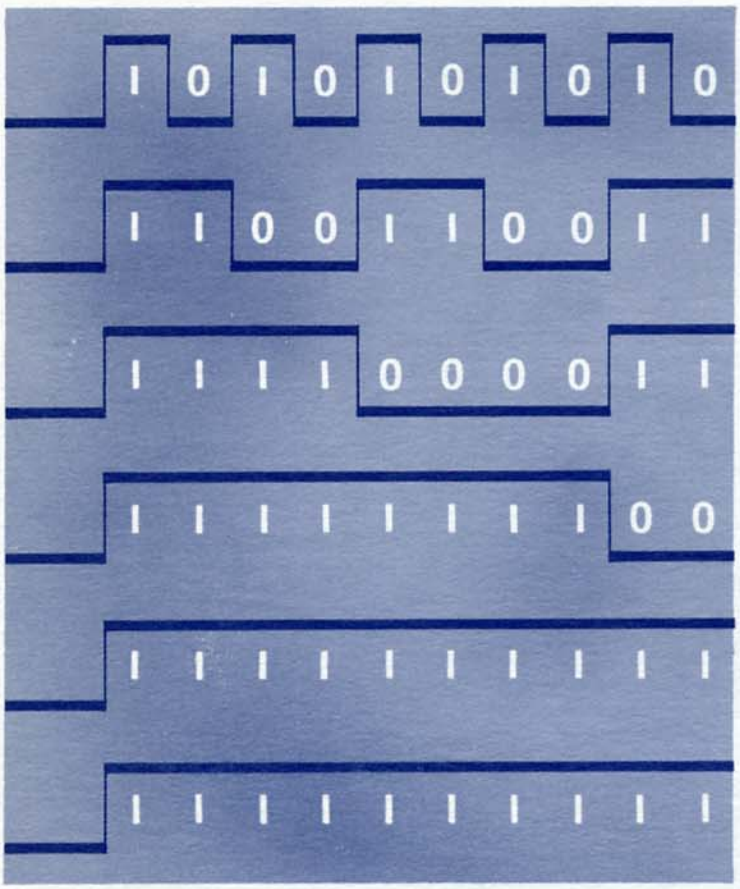


HEWLETT hp PACKARD

Engineering in the data domain calls for a new kind of digital instrument

Equipment based on concepts of frequency and time domain is no longer adequate for analyzing today's complex digital systems

by Charles H. House,
Hewlett-Packard Co., Colorado Springs, Colo.



☐ Debugging and troubleshooting a digital system can be an onerous task. Although many presently available test instruments are suitable for making gross checks or in-depth analyses, they aren't designed to handle most faults found in digital systems.

These shortcomings are not the fault of the instruments, which were designed to serve equipment operating either in the time domain, as defined by the mathematics of Heaviside and Laplace, or the frequency domain, as exemplified by the calculations of Maxwell and Fourier. Digital equipment operates in the entirely different data domain, according to rules laid down by Boole and von Neumann.

The importance of the data domain lies in the differences between digital and analog circuits. By understanding the concept, an engineer can more easily take the step from design requirements to hardware. The data domain is characterized by state-space concepts, data formats, data flows, and equipment architecture. Electronic-circuit design to implement these ideas plays a large part in the shift from analog to digital emphasis.

The data-domain problem

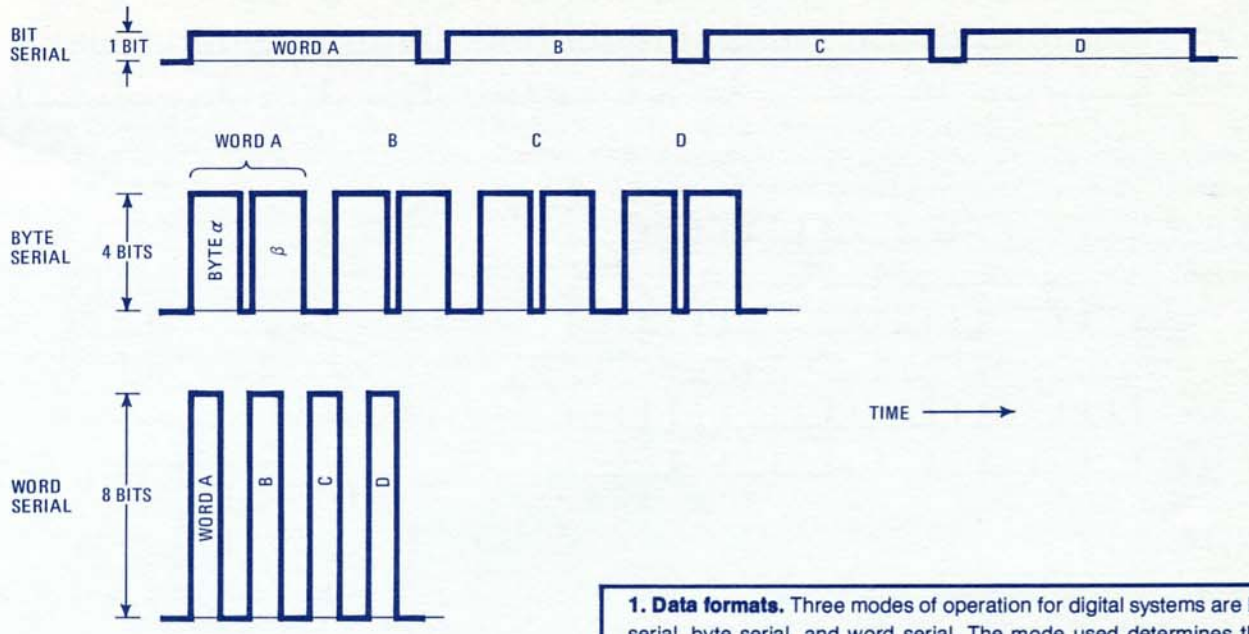
A typical data stream to be analyzed is composed of many bits of information, and, not surprisingly, every bit looks just like every other bit. Digital pulses are differentiated primarily by the choice of data format—how the bit pattern is organized into meaningful data words. For example, if one word is 8 bits long, it is possible to organize the word into 8 serial bits, 8 parallel bits, or 4 parallel bits followed by 4 more parallel bits. These formats, called bit-serial, word-serial, and byte-serial, respectively, are common in digital systems (Fig. 1).

Assume the message "data domain" is sent in ASCII code. Table 1 shows the ASCII code set, and Fig. 2 shows the data waveforms that would be sent over synchronous systems for this message in each of the three format structures. If a fault occurs—say the letter "i" is received as "y"—where can an oscilloscope be triggered, and how can the resulting display be analyzed? The answer is uncertain, and sometimes a designer or service technician must keep searching for days with instruments designed for analog analysis.

The digital world

Digital designs are based on words or data as a function of time or sequence more often than on voltage as a function of time or frequency. Only when the word flow

This article defines what the data domain is and why it is important in digital design and troubleshooting. It is the first of a two-part series. In the second part, William Farnbach of Hewlett-Packard will describe how to use data-domain instruments to solve problems in digital circuits.



1. Data formats. Three modes of operation for digital systems are bit serial, byte serial, and word serial. The mode used determines the meaning of a particular bit by putting it in context. Otherwise, any bit would be indistinguishable from any other bit.

TABLE 1:
PROPOSED AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE (ASCII)

BIT+	1234567		1234567		1234567		1234567
A	1000001	a	1000011	0	0000110	$	0010010
B	0100001	b	0100011	1	1000110	%	1010010
C	1100001	c	1100011	2	0100110	{	1101111
D	0010001	d	0010011	3	1100110	}	1011111
E	1010001	e	1010011	4	0010110	[	1101101
F	0110001	f	0110011	5	1010110	]	1011101
G	1110001	g	1110011	6	0110110	BELL	1110000
H	0001001	h	0001011	7	1110110	CR	1011000
I	1001001	i	1001011	8	0001110	LF	0101000
J	0101001	j	0101011	9	1001110	BS	0001000
K	1101001	k	1101011			HT	1001000
L	0011001	l	0011011	.	0111010	VT	1101000
M	1011001	m	1011011	,	0011010	SOH	1000000
N	0111001	n	0111011	:	0101110	STX	0100000
O	1111001	o	1111011	;	1101110	ETX	1100000
P	0000101	p	0000111	?	1111110	EOT	0010000
Q	1000101	q	1000111	'	1110010	ACK	0110000
R	0100101	r	0100111	(	0001010	DC_1	1000100
S	1100101	s	1100111	)	1001010	DC_2	0100100
T	0010101	t	0010111	-	1011010	DC_3	1100100
U	1010101	u	1010111	+	1101010	DC_4	0010100
V	0110101	v	0110111	=	1011110		
W	1110101	w	1110111	/	1111010		
X	0001101	x	0001111	*	0101010		
Y	1001101	y	1001111	#	1100010		
Z	0101101	z	0101111	"	0100010		

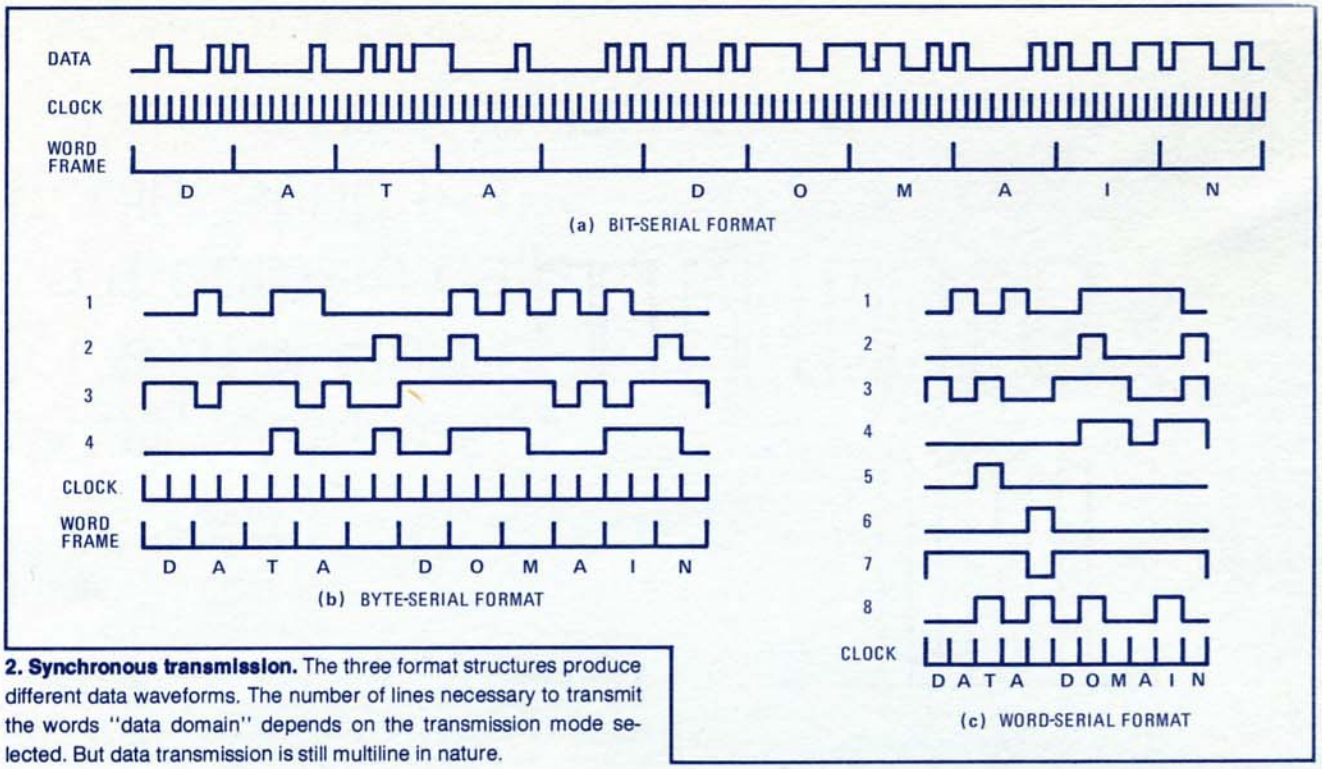
Abbreviations:

CR = Carriage return, LF = Line feed, BS = Back space
HT = Horizontal tabulation, VT = Vertical tabulation
SOH = Start of header, STX = Start of text, ETX = End of text,
EOT = End of transmission, ACK = Acknowledge, DC = Device control.

is incorrect need a technician be concerned with the voltage conditions that created the words. Even when word-flow errors require analysis of electrical parameters, the number of signal nodes in the vicinity of the error complicates the use of traditional oscilloscopes in the analysis. Thus, it is helpful to define scope functions—probing, triggering, and display—in terms either of words versus event or sequence, or words versus time, rather than in volts versus time.

The job of defining instrument capabilities desirable for testing digital systems is aided by clues from the major signal characteristics of data-domain systems:

- Digital signals are almost invariably multiline. As shown in Fig. 2, even the bit-serial format of data transmission implies use of a clock and a word-frame-counter line, yielding a minimum of three simultaneous signal lines even before control signals are considered.
- Many signals occur only once—single-shot—as the program is executed, or else the concern is only about a single occurrence. In a page of transmitted text, for example, the letter "a" may occur many times, but be in the wrong location only once.
- Many more signals occur repetitively, but aperiodically. There is no periodicity to the occurrence of the letter "a," for example. Even in architectures usually considered synchronous and periodic, such as central-processing units, variable-rate cycle times for different functions are becoming common. The present generation of microprocessors routinely operates in this way.
- Because the stimulus is seldom controllable, it is impossible to answer the classical time-domain question, "What happens after the switch is closed (or the pulse edge occurs) at time t_0?" Also, since an error typically occurs in a vast flow of correct data, it becomes practical to recognize the error only after it has occurred. This situation obviously requires capturing and storing the pertinent causes of error that occur prior to the error—negative-time signals—because they occur before the trigger at time t_0.
- Registration within a digital data stream is accomplished by unique Boolean expressions or data words. Thus instrumentation could be designed to trigger on and index the display from the trigger event as a function of words.
- The speed of digital signals varies dramatically. If



2. Synchronous transmission. The three format structures produce different data waveforms. The number of lines necessary to transmit the words "data domain" depends on the transmission mode selected. But data transmission is still multiline in nature.

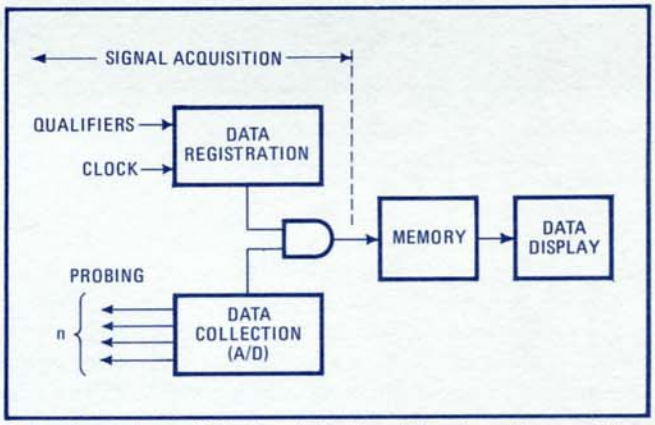
one is concerned about potential overlap of two pulses in a high-speed central processor, time resolution of 50 picoseconds is desirable. In contrast, the registration of the strobe pulse for a keystroke on an electronic typewriter may be measured in milliseconds. If an instrument is to deal with data words, the speed can be considerably slower than if it is concerned with electrical portions or components of those words. For example, in monitoring the execution of an algorithm, all that is necessary is to watch the flow of data words. When an error is detected in the algorithm, one needs to analyze the cause of that error, which may require an instrument that operates at a much higher speed.

As an extension of this example, the types of problems that exist in a digital system should be considered in terms of equipment needed to show the parameters required for analysis. A designer would have to spend considerable time with a scope to trace the execution of an algorithm, even to discover that it has an error.

The problem may be functional in that the correct function in terms of the data domain did not occur. If it did not occur as expected, the cause needs to be analyzed after the problem is located by a display of word parameters. Basically, four probable causes exist: an incorrect functional instruction may be present, a functional problem exists in hardware, an electrical problem exists in hardware, or an electrical problem from elsewhere is intermittently causing a malfunction. Obviously, one display format will be insufficient for appropriate analysis of all of these causes.

Data handling

Several instrument companies have introduced products designed to solve the problem of testing in the data domain. In such applications, this new class of instruments, which may be called logic analyzers, offer several advantages over time-domain test instruments because of innovative approaches to signal acquisition, processing, and display (Fig. 3).



3. Logic analyzer. Data-domain testing differs from time- and frequency-domain testing in the means used for probing, data collecting, data registration, memory, and data display. Shown here are the data and control-signal paths in a logic analyzer.

Signal acquisition may be divided into three stages: probing, data registration, and data collection. Probing, in both electrical and mechanical terms, requires attention to the multi-node, variable-level, physically restricted areas of access to densely packaged digital hardware (Fig. 4). Data registration is essentially the triggering function, but it may include sampling strobes to indicate when data should be collected. Data is now collected for analysis by using either single- or dual-level comparators that process at a clock edge. If the clock edge is generated by the system under test, the analyzer is said to operate synchronously. If the clock is

internal to the analyzer, it operates asynchronously to data.

Because the acquired signal is available in the instrument in digital form, it can easily be stored in memory. The data can then undergo further processing, if required. For example, the analyzer may generate displays other than the conventional curve of level versus time. And since events that occurred before the trigger can be stored in memory, the events leading up to a malfunction may be displayed and analyzed for probable causes of error.

Data registration—finding a unique point within a long data stream to establish the reference for a meaningful measurement—is a complex problem. Several requirements exist: a unique starting point must be recognized; a scanning area, or search window, as well as a display window, must be defined; if the scanning area is not adjacent to the starting point, a means must be provided for indexing the display with respect to the starting point, and a stopping point must be established.

In time-domain equipment, oscilloscope triggering and sweep circuits provide these functions. The starting point is determined by the slope, coupling, and threshold controls. The scanning area or display window is set by the sweep-time control. Indexing is commonly provided by a delaying sweep, which sets a time-interval holdoff between the starting pulse and the scanning area. The stopping point is provided by the sweep-holdoff circuit, which allows all circuits to reset to their initial conditions after the sweep-time circuit has finished its traverse.

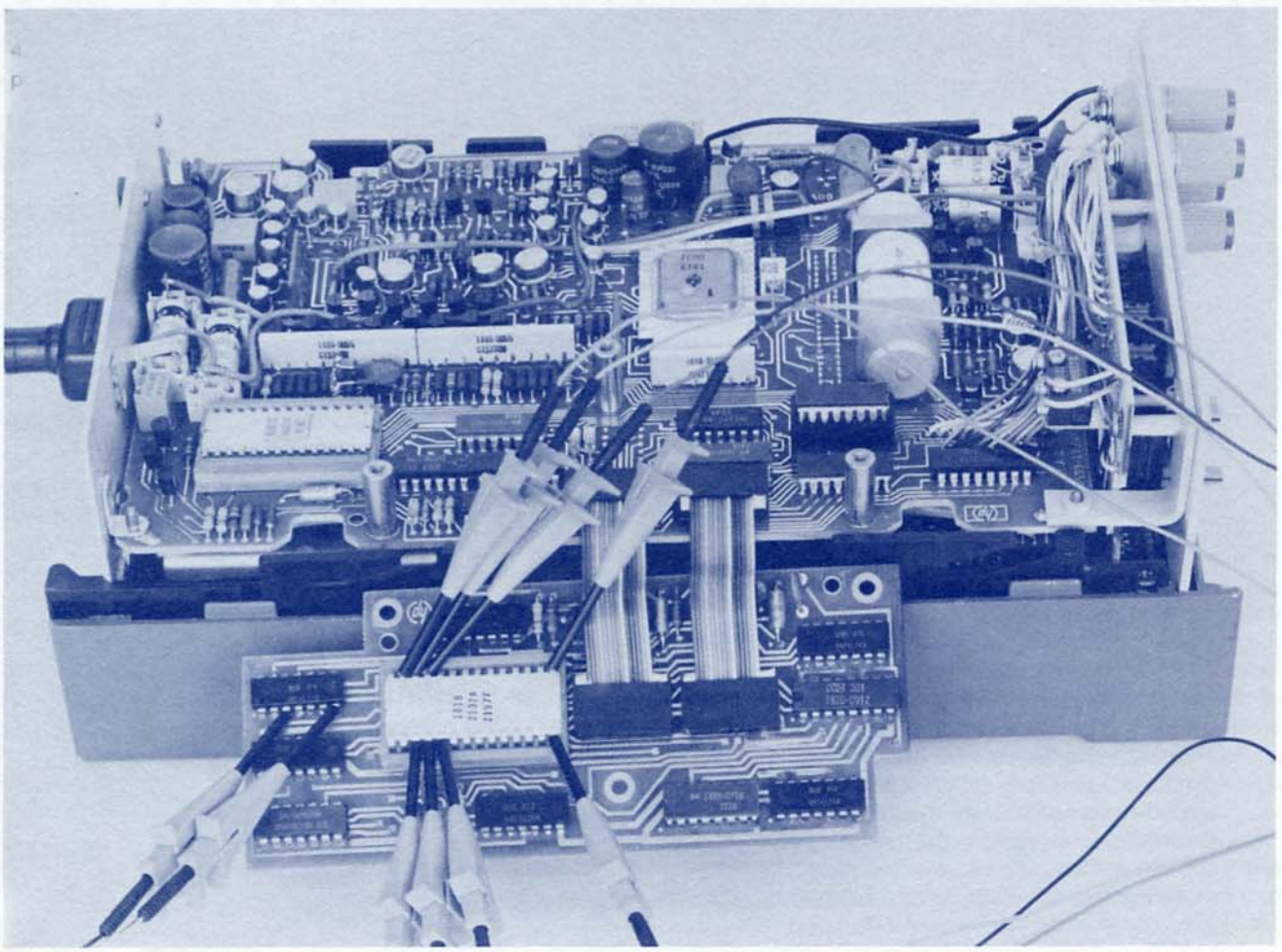
The data domain offers an analogous situation. The starting point, which may be termed a Boolean or word-pattern trigger, occurs at a certain time in the form of a unique word pattern. An indexing capability may be provided in whatever indexing parameter is convenient for the equipment under test—bits, words, time, frames, or blocks. A scanning area is defined by memory size or word-boundary conditions. And a stopping point may be determined, either by a filled memory or the recognition of another unique word pattern.

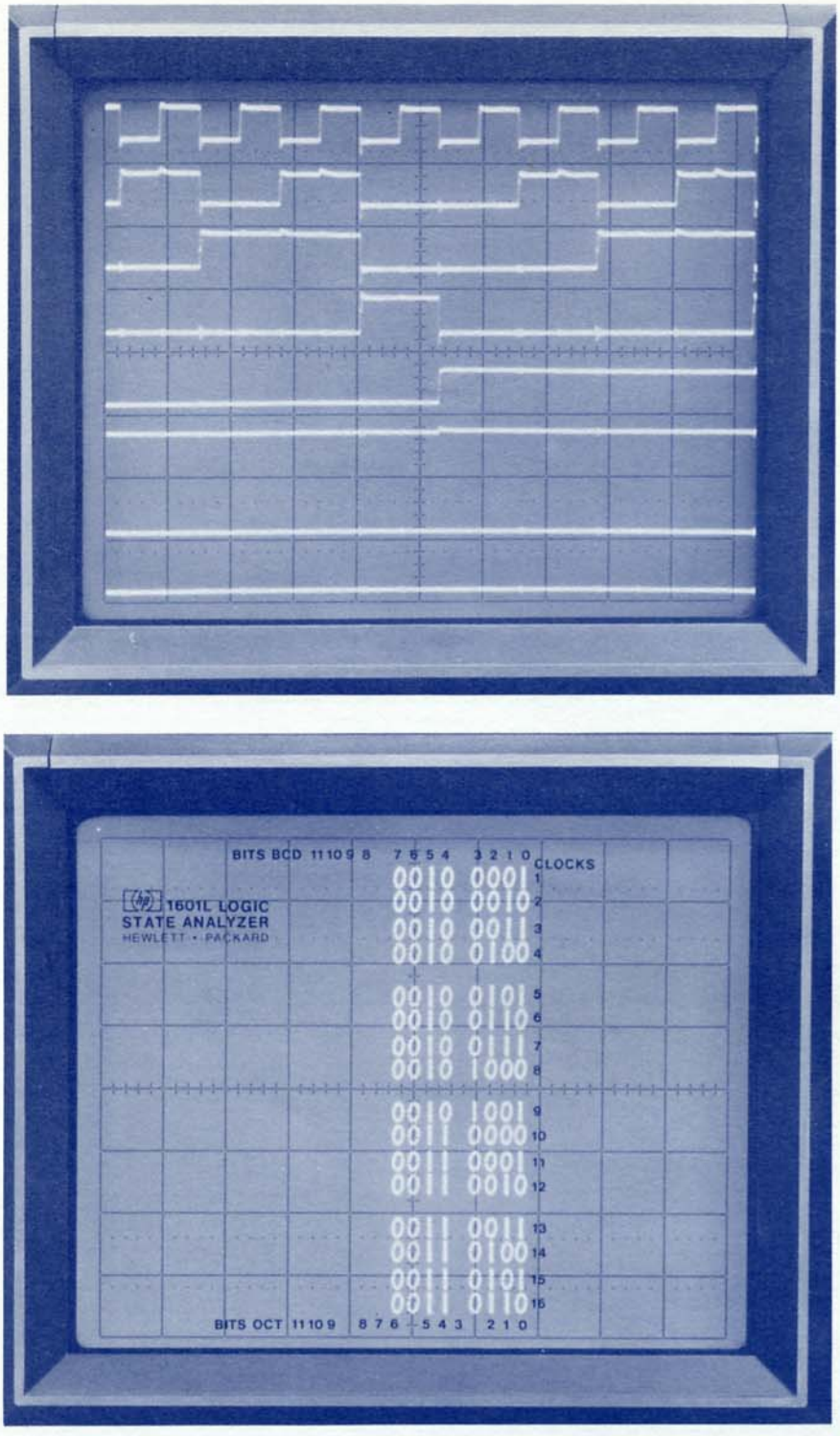
Instrument solutions

Data-domain test instruments that make use of these concepts are divided into three classes: logic-state analyzers, logic-timing analyzers, and logic triggers for oscilloscopes (Table 2).

Logic-state analyzers display binary data in a word-versus-event format. Because they concentrate on word sequences, they are useful in examining the functional

4. High density. Because digital systems are so densely packaged, data-domain test instruments must use innovative approaches to probing points under test. Small probes make it easy to connect these instruments to closely spaced test points.





5. Counter display. The output of a two-decade counter can be displayed by either a logic-timing analyzer (top) or a logic-state analyzer (bottom). The logic-timing analyzer display yields more information about the timing relationships between signals, but the logic-state analyzer display is more easily read, and so is more useful when information about logic levels is all that is required.

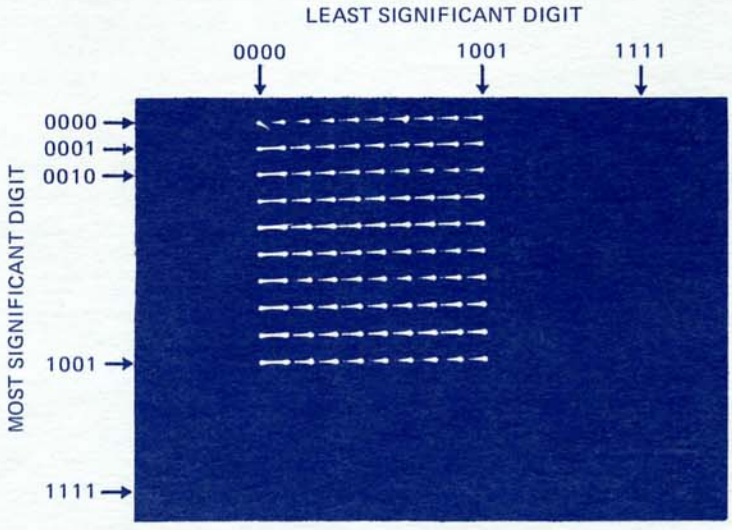
behavior of digital systems. They may provide an output suitable for triggering an oscilloscope for voltage-versus-time displays when electrical analysis is required. The logic-state analyzer may also provide facilities for some correlation analyses, such as displaying a mathematical combination of two different data fields.

A logic-timing analyzer, on the other hand, displays binary data in a word-versus-time format, or it may reconstruct the original voltage waveform and display it in a pseudo-voltage-versus-time mode. Because these instruments display bit sequences, they are most useful in examining the functional behavior of subsystems and components. Some capability to analyze electrical parameters is usually included in a logic-timing analyzer so that glitches, rise times, or pulse-ringing can be detected.

Logic-trigger generators, designed for use with oscilloscopes, are simpler instruments. They may be as simple as four-input AND gates connected to scope trigger inputs to synchronize the scope with the occurrence

TABLE 2: MEASUREMENT ABILITIES OF DIFFERENT INSTRUMENTS

INSTRUMENT	DATA ACQUISITION			MEMORY	DISPLAY FORMAT	
	Probing nodes	Collection repetition rate	Registration trigger and index	Negative-time information	Parameters	Speed
Logic state analyzer	Many	Single shot, aperiodic, periodic	f(Word), f(Event)	Yes N words	Word vs Event A ⊕ B	Word rate
Logic timing analyzer	Many	Single shot, aperiodic, periodic	f(Word), f(Event)	Yes mN words	Word vs Time Pseudo V vs Time	mWord rate ~Voltage rate
Oscilloscope trigger generator	Many	Single shot, aperiodic, periodic	f(Word), f(Event)	–	–	–
Oscilloscope	Few	Single shot – 1, 2 channels, periodic	f(Voltage), f(Time)	No	Voltage vs Time	Voltage rate



(a) STATE MAP

0	20	40	60	80
1	21	41	61	81
2	22	42	62	82
3	23	43	63	83
4	24	44	64	84
5	25	45	65	85
6	26	46	66	86
7	27	47	67	87
8	28	48	68	88
9	29	49	69	89
10	30	50	70	90
11	31	51	71	91
12	32	52	72	92
13	33	53	73	93
14	34	54	74	94
15	35	55	75	95
16	36	56	76	96
17	37	57	77	97
18	38	58	78	98
19	39	59	79	99

(b) DECIMAL CODE DISPLAY

6. Variety. Logic states may be presented in many ways. Two of them are shown here. The state map (left) is a graphic presentation that clearly shows the stages through which a counter steps. The decimal code display (right) is another way to show the same thing.

of a particular parallel word. They may also include digital counters to delay triggering by a set number of events, or shift-register comparators to allow bit-serial word recognition for triggering.

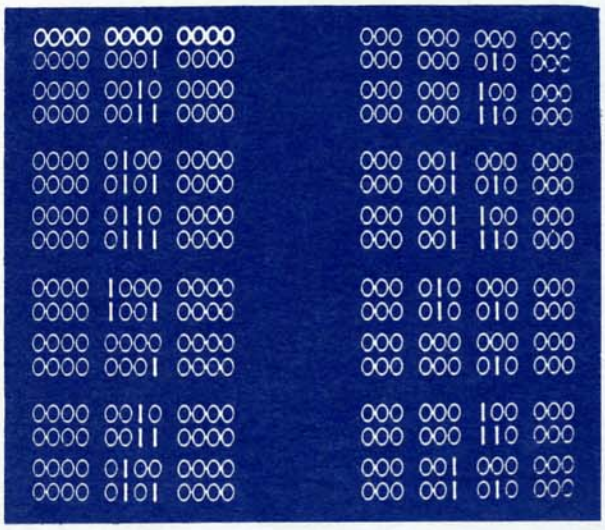
Each type of data-domain instrument, of course, has both merits and limitations for different applications (Table 3). For example, a logic-state analyzer is the best choice for the purely functional tests of verifying a counter output sequence or checking a microprocessor program. A logic-timing analyzer is more suitable for such essentially electrical measurements as checking logic-threshold noise immunity and propagation-delay times. A logic-trigger generator may be adequate when cost is a consideration and an oscilloscope is readily available.

A common application can illustrate how these considerations affect the choice of analysis tool. The count-sequence output of a two-decade decimal counter can be displayed by either a logic-state analyzer or a logic-timing analyzer (Fig. 5). The timing analyzer display looks much like what might be expected from an oscilloscope—if that time-domain instrument could handle eight channels simultaneously. Although this timing diagram yields more information than does a truth-table display, its greater complexity makes it less useful than a truth table when only functional representation of the circuit under test is required.

For multilevel data, the relative ease of reading a word-encoded sequence by means of a logic-state analyzer is obvious. In contrast, reading several binary-level channels by means of a logic-timing analyzer is extremely difficult. For example, checking the execution

TABLE 3: TYPES OF LOGIC ANALYZERS

TYPE	ADVANTAGES	LIMITATIONS	CURRENT EQUIPMENT
Logic state analyzers	Multichannel single-shot Multichannel negative-time Display of words	Lack of voltage-vs-time capability	Hewlett-Packard 1601L Hewlett-Packard 5000A
Logic timing analyzers	Multichannel single-shot Multichannel negative-time Familiar timing diagram	Lack of word format Higher cost for voltage-vs-time resolution	Biomation 810D, 8200 E-H Digiscope (AMC 1320) Iwatsu LS-6211
Logic trigger generators	Everyone owns a scope Lower cost for voltage-vs-time resolution	Lack of negative time Lack of multichannel single-shot Lack of word data	Hewlett-Packard 10250A, 10251A, 10252A Hewlett-Packard 1620A Dumont 2100A Tektronix 821A Tektronix DD501, 5B31, 7D11, 7D15



7. Legibility. Twelve-bit patterns can be broken up into groups of 3 or 4 bits to make reading easier.

of a branching algorithm is much easier with a logic-state display than with a logic-timing display, especially if long wait loops or idling loops are involved.

Clearer presentations

Of course, it is possible to develop displays that are even more easily readable for functional checks. Two are shown in Fig. 6. The first is a graphic map of the binary words of a counter output. The intersection of a row, which denotes the least significant digit, and a column, which represents the most significant digit, is a unique dot position signifying a unique word.

The dot at the upper left-hand corner represents the word 00000000, the dot in the upper right-hand corner the word 00001001, and the dot at the lower right-hand corner 10011001. The intensity of the display increases as the trace nears a new point, so the direction of flow between states can be determined. If an unauthorized state occurs, the access path to that state can readily be determined. If a necessary state does not occur, it is obvious which did not and what occurred instead.

Figure 6 also shows a decimal-code display of a counter output. This is easier to read than its binary-coded-decimal equivalent, as the BCD equivalent is easier to read than a voltage-versus-time plot. The code conversions possible for the data domain are numerous and varied; instruments currently provide only the most elementary of them—binary equivalents that can be organized into 3- or 4-bit bytes for easier reading of octal, BCD, or hexadecimal data (Fig. 7).

When analyzing why a data error occurs, as opposed to finding whether or where, it frequently becomes important to display the word events line by line, usually as a function of time. For this task, scopes and logic-timing analyzers are better than logic-state analyzers. When an error has been found, the cause—a glitch, a noise spike, or a faulty instruction—must be found. A timing analyzer or a scope allows the user to magnify the area surrounding the fault and watch for unexpected level transitions. This analysis requires attention to occurrences of much shorter duration than a data word, which increases the required data-collection rate of the test equipment. Consequently, logic-timing analyzers should be n times faster than logic-state analyzers for the same data fields, where n subcycles of resolution are required to analyze a data error.

No single product covers all requirements, and even if all the capabilities were incorporated in a single instrument, it would cost too much and be too complex for many jobs that one instrument—a logic-state analyzer, logic-timing analyzer, or logic-trigger generator—could easily perform. The message is clear: digital-equipment problems require data-domain instruments for solutions. Instrument manufacturers are beginning to recognize the need and build appropriate test equipment. The digital designer can only benefit from this new approach to problem-solving. □

5952-1974 PRINTED IN U.S.A.