

The spectrum of a frequency-limited periodic wave may be obtained rapidly by measuring its truncated Walsh spectrum, and then converting from Walsh spectrum to Fourier spectrum after the measurement. The conversion process consists of a matrix multiplication in which a measured Walsh spectrum vector, of dimension  $2^k$ , is multiplied by a  $2^k \times 2^k$  conversion matrix that is compensated for Walsh-spectrum truncation, to yield the corresponding Fourier spectrum vector. The microprocessor is well suited to this end; it is also useful in monitoring instrument panel switches and driving a display and print-out. This paper compares available microprocessors from the viewpoint of BCD processing—and concludes that the Fairchild PPS25 processor is the one best suited to meet the requirements.

# USING A MICROPROCESSOR IN A WALSH-FOURIER SPECTRAL ANALYZER

Reuven Kitai, Istvan Renyi,\* Ferenc Vajda\*  
McMaster University

## Introduction

An alternative to the direct measurement of the Fourier spectrum of a periodic wave, using a portable instrument, is to measure the Walsh spectrum and then to convert the measured spectrum to the Fourier spectrum using a special-purpose converter. The cal and sal sequency components  $A_s$  and  $B_s$  of a wave  $f(\theta)$  are given by

$$A_s = \int_0^1 f(\theta) \text{cal}(s, \theta) d\theta$$

$$B_s = \int_0^1 f(\theta) \text{sal}(s, \theta) d\theta$$

where  $s$  is sequency and  $\theta = t/T$  is the normalized time. It is well known<sup>1</sup> that if the highest harmonic

in a periodic signal is  $F$ , then  $2F$  Walsh values plus the dc value suffice to define it. Measurement of the spectral components  $A_s$  and  $B_s$  is attractive because cal and sal waves switch between +1 and -1, thereby obviating the need to multiply  $f(\theta)$  by the cosines and sines needed in Fourier processing. In Walsh processing, signal values are merely accumulated, with or without sign change, depending on the signs of signal, sal, and cal values at any instant. Also, synchronous Walsh wave generators exhibiting low orthogonality errors are extraordinarily simple and economical when compared with their Fourier counterparts.<sup>4</sup> Of course the Walsh spectrum is meaningful in its own right: it contains all the required information. However, the extension to convert to the corresponding Fourier spectrum after the measurement can be achieved simply and economically, using a microprocessor that is built into the instrument.

In principle, a single cycle of  $f(\theta)$  defines it fully; in practical measurements, however, one must first tune the Walsh waves to the signal, and this usually requires two, or perhaps three, additional cycles. For signals of low fundamental frequency the

\*On leave from the Central Research Institute for Physics of the Hungarian Academy of Sciences, Budapest, Hungary.

complete Walsh spectrum vectors are obtained by using a Walsh array generator and parallel accumulation. At fundamental frequencies above, say, 10Hz one may economize on hardware by using a serial processor in conjunction with a digitally addressable Walsh generator.<sup>4</sup>

It is desirable to acquire  $2^n \geq 2F$  Walsh spectrum values (dc included), not only because a Walsh set naturally comprises  $2^n$  functions ( $n$  being an integer), but also because the conversion process from Walsh to Fourier expansions is then of the matrix form

$$\begin{aligned} [a] &= [K_a][\Lambda_a][A] \\ [b] &= [K_b][\Lambda_b][B] \end{aligned} \quad (1)$$

where  $[A]$ ,  $[B]$  are the measured cal and sal spectrum vectors, each of dimension  $2^{n-1}$ , and  $[a]$ ,  $[b]$  are the desired cosine and sine spectrum vectors.  $[\Lambda_a]$ ,  $[\Lambda_b]$  are conversion matrices, each of order  $2^{n-1} \times 2^{n-1}$ , the elements of which are Fourier coefficients of Walsh functions;<sup>2,3</sup> and  $[K_a]$ ,  $[K_b]$  are diagonal truncation compensation matrices, each of order  $2^{n-1} \times 2^{n-1}$ , the coefficient values being  $[(\sin x)/x]^2$  terms, where  $x = \pi f/2^n$  for  $f < 2^{n-1}$ . The last coefficient value in the  $K$  matrix ( $f = 2^{n-1}$ ) is  $\pi^2/8 \approx 1.23$ .

When implementing the conversion process, the premultiplied matrix product elements  $[K][\Lambda]$  are stored in a ROM so that (1) reduces to the multiplication of a column matrix by a square matrix for the cosine coefficients, and likewise for the sine coefficients. Relatively cheap LSI microprocessors are well suited to this end. They may also be usefully employed in the monitoring of switch settings and for the display of results. Our procedure was to compare available microprocessors to meet these objectives, having BCD arithmetic in mind, and to implement the matrix multiplication, using the selected microprocessor system.<sup>5,6</sup>

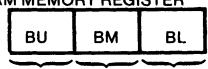
## Microprocessor Selection Procedures and I/O Considerations

Our approach to microprocessor selection was to use "bench-mark" programs that are typical of the application, writing them, and, if possible, running them (if program simulators are available). This method was used, viewing the matrix multiplication as "bench-mark." We compared four microprocessor types capable of BCD operations: the Fairchild PPS-25, Rockwell PPS-4, Intel MCS-4, and Intel 8080. To simplify the problem, we used only BCD addition as a basic element of multiplication. Thereafter the efficiencies of the individual programs were evaluated on the basis of execution times and on the memory capacity required for storing them.

On analyzing the instruction sets of the different microcomputers we found that an instruction set including the BCD addition operation is not the most important feature, because decimal correction

may be carried out quite simply for all of the types considered. The possibility of multi-digit register definition and usage of such registers in multi-digit instructions seemed to be more important. The main advantage of the PPS-25 is that it enables the selection of digit "slices" of data registers according to different previously-defined digit patterns, and to execute operations on an assigned multi-digit part of a single instruction. The PPS-4 has the same capabilities, and one could define instructions for only a single digit; but it is possible to arrange multi-digit registers by memory organization, with a single instruction to load a digit and point to the next one (see Table 1). In the case of the MCS-4 and

**Table 1. Processor instructions and operations**

TYPE	MNEMONIC	INSTRUCTION	DESCRIPTION OF OPERATION
PPS-25	TE	TIME ENABLE PATTERNS	SLICES OF THE 25 DIGIT DATA REGISTERS CAN BE SELECTED BY 6 DIFFERENT TIME INTERVALS (BY MASK PROGRAMMING). THE DIGITS OPERATED ON IN AN ARITHMETIC INSTRUCTION ARE DEFINED BY THE SELECTED TE PATTERN.
PPS-4	LD	LOAD ACCUMULATOR FROM MEMORY	M → A B(7:5) ← I(3:5) → B(7:5) THE 4-BIT CONTENT OF RAM CURRENTLY ADDRESSED ARE PLACED IN THE ACCUMULATOR. B: RAM ADDRESS REGISTER I(3:1): IMMEDIATE FIELD OF THE INSTRUCTION ← EXCLUSIVE OR
PPS-4	EXD	EXCHANGE ACCUMULATOR AND MEMORY AND DECREMENT BL	RAM MEMORY REGISTER  <p>RAM # REGISTER WORD LOCATION LOCATION A ↔ M B(7:5) ← I(3:5) → B(7:5) BL-1 → BL Skip on BL = 1111</p>
MCS-4	RDM	READ RAM CHARACTER	M → A
MCS-4	WRM	WRITE A INTO RAM CHARACTER	A → M
MCS-4	ISZ	INCREMENT INDEX REGISTER SKIP IF ZERO	(R) + 1 → R IF RESULT 0 P <sub>n</sub> → P <sub>n</sub> P <sub>n</sub> → P <sub>n</sub> P <sub>L</sub> + 2 → P <sub>L</sub>
8080	STAX D	STORE A IN MEMORY ADDRESSED BY REG. D AND E	(A) → [(D)(E)]
8080	LDAX D	LOAD A WITH MEMORY	[(D)(E)] → A
8080	INX D	INCREMENT D & E	(D)(E) + 1 → (D)(E)
8080	JNZ <B <sub>1</sub> > <B <sub>2</sub> >	JUMP IF NONZERO	IF (ZERO) = 0 <B <sub>2</sub> > <B <sub>1</sub> > → (PC)

**Table 2. Program features**

PROCESSOR TYPE NUMBER	CLOCK FREQUENCY [MHz]	EXECUTION "TIME UNIT" [μsec]	NUMBER OF "TIME UNITS" IN THE PROGRAM	PROGRAM RUN TIME [μsec]	PROGRAM MEMORY CAPACITY [bit]
PPS-25	0.4	2.5	25	62.5	12
PPS-4	0.2	5.0	100	500.0	80
MCS-4	0.74	1.35	1336	1704.0	136
8080	2.09	0.48	479	230.0	152

8080, where the content of a digit counting register must also be incremented, we have to take the single digit instruction into a program loop; the end of the cycle is detected by either a skip or a jump instruction.

The main features of the programs are shown in Table 2, which lists program run time and memory capacity required as main features, in addition to the clock frequency, execution "time unit," and the number of "time-units" in the program. The PPS-25 emerges as the best choice. Considering the run time of the matrix multiplication routine, the results given in Table 2 do not supply an entirely objective picture; however, execution time was not a key factor in our application.

Input/Output interfacing represents an important area in the design of microprocessor-controlled measuring instruments. In general, and also in our case, three basic tasks have to be considered: (1) input data to be processed, (2) monitor front panel switches and controls, and (3) output calculated results (numeric displays, printer, etc.).

As a consequence of the unique I/O structure of the PPS-25,<sup>5</sup> the external data source needs to be realized in the form of a shift register; this register can be selected by any processor arithmetic instruction in which the "Dummy Register" is used as

destination register. A similar method is used for outputting temporary data. The fact that the PPS-25 is not able to access an external data base directly causes only a slight inconvenience. Due to the simple addressing algorithm of the matrices to be multiplied and stored, only a small amount of auxiliary addressing hardware is necessary.

The simplest way to consider the settings of the different front panel switches is to modify the program execution. Two attributes of the PPS-25 are applied in the realized system: immediate or delayed (stored) address modification (jump), and the representation of an input code as status information, used for program branching. The implementation of these methods is facilitated by making use of the keyboard encoder chip included in the set. A display chip, also included in the set, drives a numerical panel display directly. It also cares for the necessary digit selection.

### Process Realization

The functional block diagram of the Walsh-Fourier Spectrum Analyzer is shown in Figure 1, and the flow diagram of the entire instrument is given in the Appendix. The instrument can be divided into two

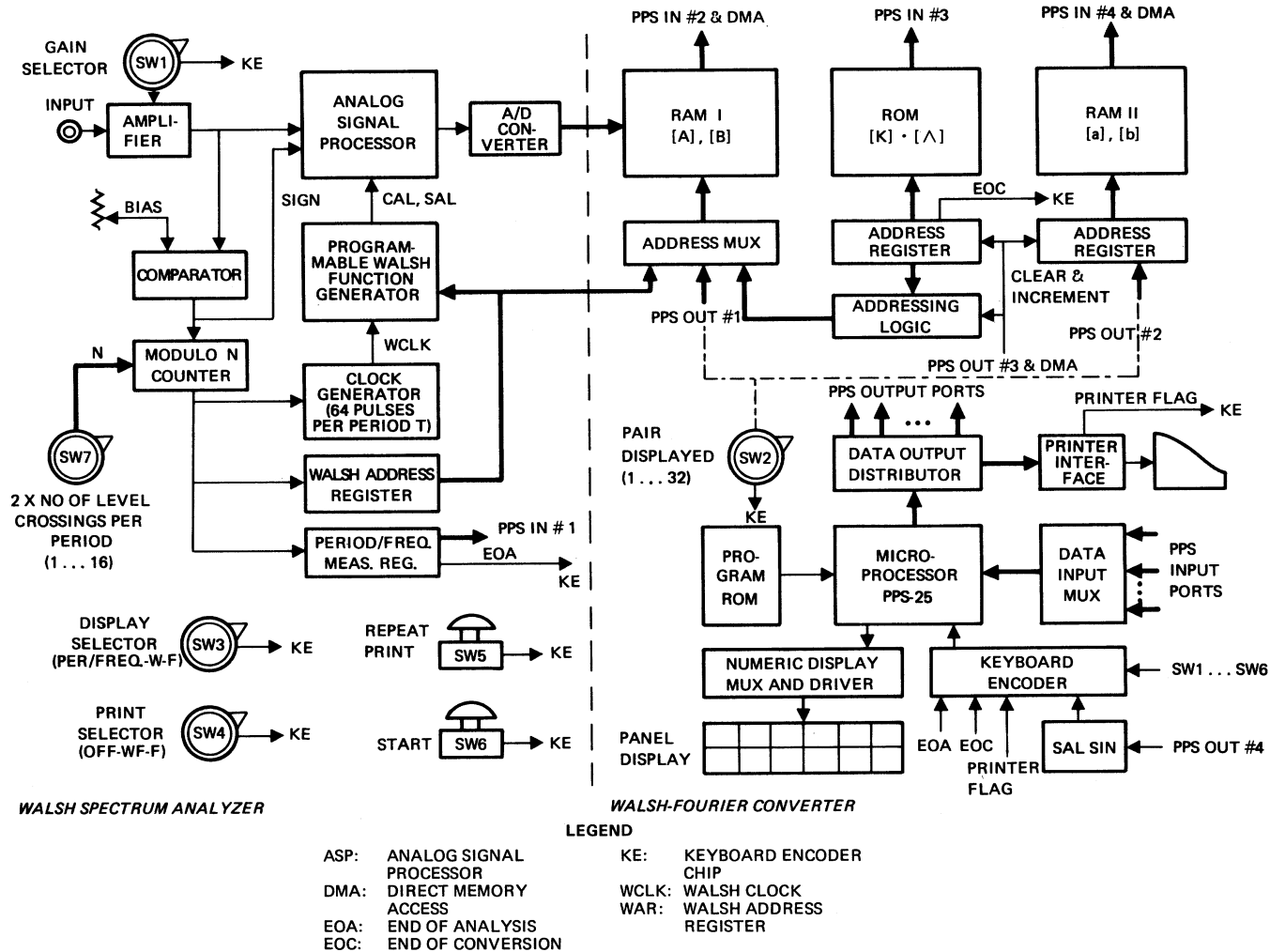


Figure 1. Walsh-Fourier Spectrum Analyzer functional block diagram

main sections: Walsh Spectrum Analyzer and Walsh-Fourier Converter. The Walsh Spectrum Analyzer—shown to the left of the broken line—produces the Walsh spectrum vector of a periodic analog signal and stores it in RAM I. Here the most important unit is the signal processor which, when fed from the input amplifier, simultaneously produces Walsh spectrum components  $A_{\text{WAR}}$  and  $B_{\text{WAR}}$  (orthogonal pairs) of sequency defined by the Walsh Address Register (WAR). For fundamental frequencies below 500 Hz, digital operation is preferred, while for higher frequencies analog methods are faster. The digital signal processor employs A/D conversion for each of 64 equally-spaced time intervals of the signal period, followed by positive or negative accumulation depending on the sign of the signal and of the Walsh function in the interval considered. The analog signal processor integrates the original or the inverted signal in each interval, and at the end of the period an A/D converter produces  $A_{\text{WAR}}$  and  $B_{\text{WAR}}$  in digital form. With the joint use of the digital and analog signal processor in the same instrument, signals encompassing a wide range of fundamental frequency can be analyzed.

A second route for the signal leaving the input amplifier is via a level crossing comparator, followed by a modulo N counter. This logic passes a pulse for the following stages at every Nth positive signal level crossing (N is set by SW7). The time between two successive pulses is taken as period T.

The unit "PERIOD/FREQ. MEAS. REG." measures period T (or frequency  $1/T$  above 100 Hz) by means of its built-in 10 MHz crystal clock. The function of the "CLOCK GENERATOR" is to generate a square wave (WCLK) of frequency  $64/T$ , coherent with the analog signal level crossings. The Walsh Function Generator<sup>4</sup> produces functions cal and sal of sequency determined by the Walsh Address Register (WAR) for the signal processor.

At the end of a measurement period the hold outputs of the integrators in the analog signal processor are converted to three-digit signed BCD numbers, and the two Walsh components are stored in the appropriate locations of memory RAM I. When the digital signal processor is used, the outputs can be stored directly. A/D conversion and store operations take one or more periods. The next complete T is again a measurement period, etc.; the measurement, conversion, and store periods follow each other until all 32 orthogonal pairs have been processed. This is the end of the Walsh spectrum analysis phase which is followed immediately by the Walsh-Fourier conversion.

According to the conversion algorithm, the pre-multiplied conversion and compensation matrix  $[K]*[A]$  is multiplied by the measured spectrum vectors cal ([A]) and sal ([B]) stored in RAM I. The result, the cosine and sine spectrum vectors [a] and [b] respectively, are stored in RAM II. The conversion matrix used is the same for both the cal→cos and the sal→sin conversions, except for the signs of the elements, so that two signs have to be stored

together with each  $[K]*[A]$  element. The multiplication and accumulation necessary for the matrix multiplication are performed by the PPS-25. As mentioned above, all RAM and ROM memories have addressing logic of their own, and the microprocessor has only to supply them with clear and increment pulses.

In principle the inputting or outputting of a 3-digit, floating point number requires two machine word-cycles—i.e., an I/O addressing and a data transfer cycle ( $2 \times 62.5\mu\text{s}$ ). In many instances, however, the sequence of I/O operations is determined (e.g., address incrementation is always followed by reading from ROM and then reading from RAM I); therefore it is not necessary to include an I/O addressing cycle before each data transfer. In this way the total time necessary for matrix multiplication may be shortened substantially at the expense of some additional hardware. There are 342 nonzero elements in the  $32 \times 32$  square  $[K]*[A]$  matrix, and due to the addressing algorithm, all zero elements are skipped in the multiplication process. A 3 by 3 digit floating point decimal multiplication takes about 4 milliseconds on the average, and requires 19 programming memory (ROM) locations. The total cal→cos or sal→sin conversion takes about 2.5 seconds.

When the conversion is complete, the microprocessor arranges for the printout of the Walsh or Fourier spectra depending on the position of PRINT SELECTOR switch SW4 (OFF—W & F—F). The printer is interfaced to the PPS-25 and receives data from RAM I and RAM II through the accumulator. The microprocessor also takes care of the format of the printout and writes the appropriate text above each of the spectra. Additional printouts may be obtained by pressing button "REPEAT PRINT" (SW5).

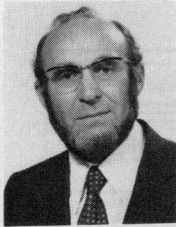
Following the printing phase, the two panel displays continuously display a pair of numbers according to the position of switch SW3. This may be any one of the following:

- Period and frequency of the analog signal,
- Walsh coefficients of sequency selected by SW2, or
- Fourier coefficients of harmonic selected by SW2.

In order to display the true magnitude of the coefficients, the setting of gain selector switch SW1 is sensed. An alternative way of displaying the Walsh and Fourier spectra is to use a CRT display. For driving a non-storage tube display, however, the PPS-25 is far too slow, and therefore direct memory access (DMA) should be used. For simplicity the panel displays are blanked, and printing should not take place during DMA. The entire program occupies 320 12-bit words of memory, of which about 120 locations are required for the conversion.

In the design phase the PPS-25 was interfaced to a PDP-8/L computer which simulated the program ROM and allowed monitored running of programs. This is worthwhile, partly to become familiar with the hardware (the PPS-25 system is not very well documented), but mainly to ease program writing and debugging. ■





**Reuven Kitai** is a professor of electrical engineering at McMaster University in Hamilton, Ontario. He received his degrees in electrical engineering from the University of Witwatersrand, Johannesburg. His current research interests lie in the applications of microprocessors and other hardware to the broad areas of signal analysis and synthesis.



**Istvan Renyi** has been on the Research Staff of the Central Research Institute for Physics of the Hungarian Academy of Sciences since 1968, when he obtained his diploma in electrical engineering from the Technical University, Budapest. His current interests are microprogramming and multimicroprocessor system design. Earlier he participated in the design and development of several digital systems, including a minicomputer and a

graphic display terminal.

Mr. Renyi was with the McMaster University, Hamilton, Ontario in 1973 as a research associate.



**Ferenc T. Vajda** has been a senior research fellow at the Central Research Institute for Physics of the Hungarian Academy of Sciences since 1957, and also an associate professor in the Instrumentation and Measuring Technics Department of the Technical University, Budapest. His current interests are man-machine communication, microprogramming, and LSI microprocessor applications. He was with the McMaster University as a post-doctorate fellow in 1972-73. Dr. Vajda received the diploma in electrical engineering in 1957 and the doctor of technics degree from the Technical University of Budapest in 1968.

## References

1. H. F. Harmuth and R. deBuda, "Conversion of Sequence-Limited Signals into Frequency-Limited Signals and Vice Versa." *IEEE Trans. on Information Theory*, IT-17, 1971, pp. 343-4.
2. R. Kitai, "Walsh to Fourier Spectral Conversion for Periodic Signals." *IEEE Trans. on Electromagnetic Compatibility*, EMC-17, 1975, pp. 266-9.
3. N. M. Blachman, "Sinusoids versus Walsh Functions." *Proc. IEEE*, Vol. 62, 1974, pp. 346-354.

## DIAGNOSIS AND RELIABLE DESIGN OF DIGITAL SYSTEMS

*M. A. Breuer and A. D. Friedman*

This book considers the problems of test generation, simulation, and reliability enhancing design techniques for digital circuits and systems. Features include: practical test generation algorithms including the *D-algorithm*, *Boolean Difference*, *ATG System*, *LASAR System*; parallel, fault list (deductive) and concurrent simulation techniques; design of self checking and fault-tolerant circuits. *July, 1976, 300pp. (approx.), \$18.95.*

## JOURNAL OF DESIGN AUTOMATION & FAULT TOLERANT COMPUTING

*M. A. Breuer, Editor-in-chief*

This new journal will be concerned with computer system reliability and the use of computers in the design of digital systems. Topics include: test generation; simulation; design verification of hardware and software; fault tolerant design; physical design (partitioning, placement, routing, packaging); high level design languages; data base. Papers describing actual systems and solutions to practical problems, as well as theoretical results of potential practical significance will be emphasized. *Published quarterly beginning October, 1976. Subscription \$50.00 for institutions, \$20.00 for individuals.*

Send orders and submit papers to:

**COMPUTER SCIENCE PRESS, INC.**

4566 POE AVENUE

WOODLAND HILLS, CA. 91364 USA

California residents add 6% sales tax. Prices subject to change without notice.

Reader Service Number 421

Keep abreast of a fast moving field by participating in . . .

## Computer Society Technical Committees

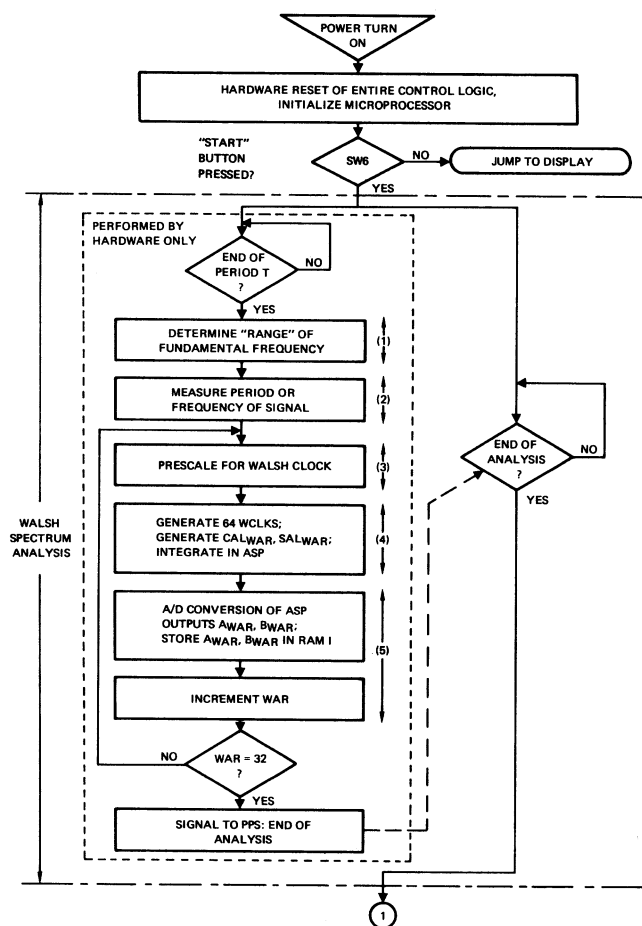
Use the Reader Service Card on page 75 to contact the TC of your choice. Simply circle the number(s) from the list below and drop the card in the mail; we'll forward your name to the TC chairmen.

Technical Committee	RS No.
Computer Architecture	400
Computer Communications	401
Computer Elements	402
Data Acquisition and Control	403
Data Base Engineering	404
Design Automation	405
Fault-Tolerant Computing	406
Machine Pattern Analysis	407
Mathematical Foundations of Computing	408
Mass Storage	409
Microprogramming	410
Mini/Micro Computers	411
Operating Systems	412
Packaging	413
Simulation	414
Software Engineering	415

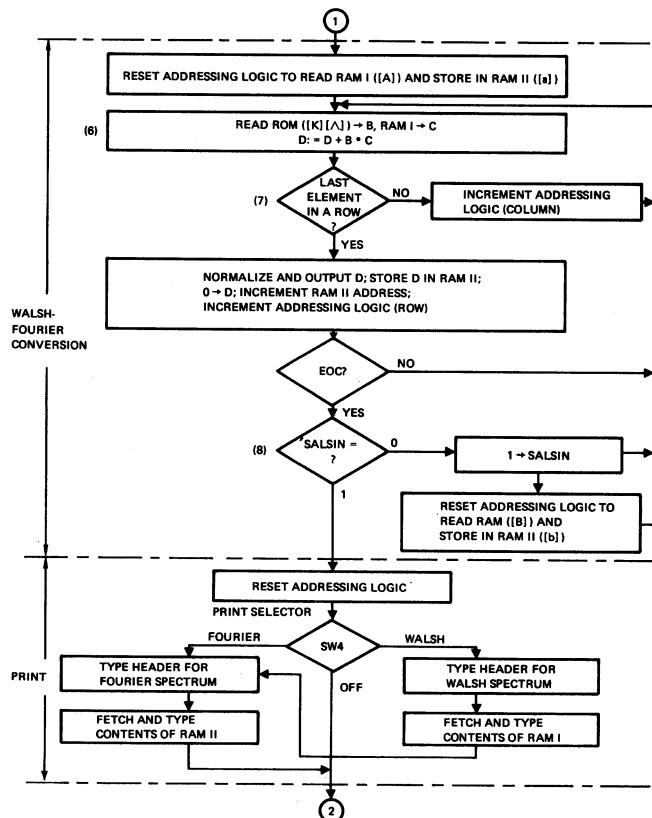
4. H. Schreiber and G. F. Sandy, Eds., "Applications of Walsh Functions and Sequency Theory," *IEEE*, 1974, 74CH0861-5EMC Chap. 15.
5. J. Ogden, "Microprocessor scorecard," *EUROMICRO Newsletter*, January 1975.
6. G. Fox, "Evaluation of Microprocessor Performance for Military Systems Application," *CAD Newsletter*, Vol. 7, 1975, pp. 6-8.

## APPENDIX A Flow of Events in the Instrument

In the flow diagram below the four phases of operation of the Walsh-Fourier Spectrum Analyzer are shown.



- (1) Takes one period T.
- (2) Takes one T or one second; can be performed during (3)
- (3) Takes one T
- (4) Takes one T
- (5) Takes one or more Ts; can be performed during (3)



- (6) A, B, C, and D are data registers of PPS-25
- (7) An additional Last Element Marker bit is stored together with each word in ROM ([K] [A])
- (8) SAL SIN = 1 during the sal → sine conversion and zero otherwise

