APPLICATIONS REFERENCE LIBRARY

EAI®

HYBRID COMPUTATION

GENERAL SECTION HYBRID COMPUTING TECHNIQUES: 1.3.7h

THE SIMULATION OF TRANSPORT DELAY WITH THE HYDAC* COMPUTING SYSTEM

ABSTRACT: A general purpose hybrid computer program for the simulation of transport delay is described. The program utilizes a combination of analog and digital components in the EAI HYDAC Computing System to provide a high speed, multichannel variable delay capability. A standard expanded Digital Operations System of the HYDAC Computer can provide up to ten (10) channels of variable transport delay.

BACKGROUND

The need to simulate a transport delay arises frequently in analog computation, occurring as the solution of the flow equation and the wave equation, both partial differential equations. More explicitly, the realistic simulation of systems involving the flow of liquids and gases in pipes or the propagation of electric or acoustic waves requires the representation of the transport delay phenomena. A standard analog computer approach to simulation of time delay is to employ a linear computer circuit, for which the ideal delay transfer function has been approximated by a ratio of polynomials (often referred to as Padé polynomials). A typical approximation, by fourth degree polynomials, is illustrated by the circuit of Figure 1. In this case, a single channel of delay employs six amplifiers and twelve potentiometers to produce a fixed delay, τ_D, where the maximum allowable value of τ_D for a one degree phase error in a one cycle per second signal is 1.2 seconds[1]. For a ten cycle per second signal, the maximum allowable delay for a one degree phase error is 0.12 seconds.

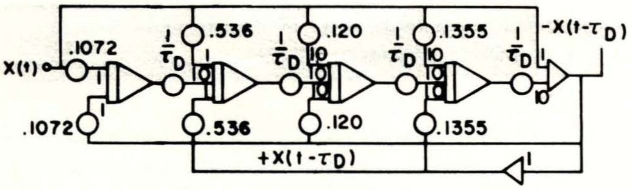


FIGURE 1: Analog Computer Circuit for Representing Transport Delay Using 4th Order Padé Approximation

Although widely used, this approach to transport delay simulation suffers from two basic limitations: 1) since large phase errors occur at high frequencies, rapidly changing and discontinuous functions can not be represented realistically; and 2) variable time delays, as with the simulation of variable fluid velocity, requires a large number of multipliers and hence becomes uneconomical.

An alternative approach for transport delay simulation employs sampling of the analog signal, storage of the discrete sampled values in a fixed number of memory cells, and reconstruction of a stair-step approximation to the function by reading out each stored value after a delay of τ_D seconds. If the sampled values are stored as "words" in a digital memory unit, the relation between sampling rate and delay time is:

$$\text{Sampling rate} = P = \frac{\text{Total number of words in Memory Unit}}{\text{Time delay in seconds}} = \frac{W_T}{\tau_D}$$

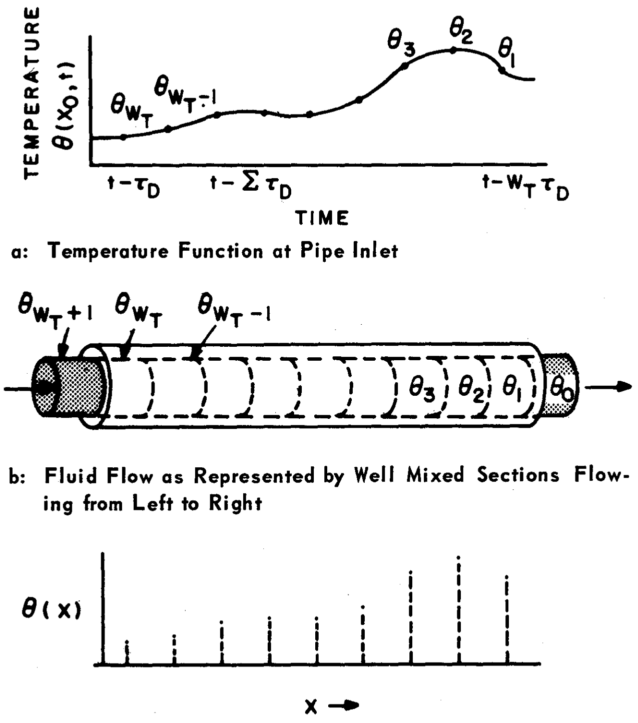
In this sampling method of transport delay simulation, the total number of words is usually fixed and a particular delay, τ_D, is obtained by selecting a sampling rate according to the above expression. The stored sampled values are read back into the analog computer, as new values are converted and written into memory. If the delay time changes as a function of an analog signal, such as the velocity of fluid in a flow process, then the sampling rate changes too—inversely, with the change in delay time.

*Trade Mark

Printed in U.S.A. 113

Bulletin No. ALHC 63011

This discrete simulation of transport delay, and the principle of variable sampling rate can be clarified by the analogy illustrated in Figure 2.



a: Temperature Function at Pipe Inlet

b: Fluid Flow as Represented by Well Mixed Sections Flowing from Left to Right

c: Discretized Temperature Profile as a Function of Pipe Length

FIGURE 2: Pictorial Representation of Discrete Sampling Method of Simulating Fluid Flow Transport Delay

The time history of the temperature of a fluid entering a pipe is to be reproduced at the outlet τ_D seconds later, where τ_D is the transit time of an element of fluid at constant velocity. Assume no heat loss in the pipe. At any instant, the temperature profile of the fluid in the pipe appears as a stored function which is moving from left to right (Figure 2c). For simulation by a device with W_T words of memory, the temperature function is sampled periodically (W_T/τ_D samples per second) and the samples stored until the W_T words are filled, whereupon the first samples appear at the outlet as the W_{T+1}, $W_{T+2}, \cdots$ samples are stored. Thus, the discrete analog can be visualized as a pipe filled with sections of fluid at constant temperatures that move in steps from left to right at a rate proportional to the sampling rate, P. Each section is well mixed and has a temperature equal to the value it had when it entered the pipe. As the sections appear at the outlet of the pipe, the temperature function is reconstructed from the values of the section temperatures to form the desired delayed function, $\theta(t - \tau_D)$.

If the fluid velocity is a changing function, then the sections of the fluid simply move through the pipe more or less rapidly. Hence, the variable time delay is simulated by changing the sampling rate proportionally to the velocity function.

The HYDAC program to be described employs this method for simulating transport delay. It utilizes analog-to-digital conversion, digital function storage, and appropriate logic components to provide a pure delay of sampled functions from the analog computer.

PROGRAM CAPABILITIES

The functional characteristics of the HYDAC Transport Delay Program are defined by the input-output relationships shown in the diagram of Figure 3. Since the number of channels and the words per channel are dependent upon the individual program, it is difficult to define the speed and accuracy of the program exactly. However, an analysis for a particular configuration will indicate the capability provided. Assume the following to be the requirements desired for a particular problem:

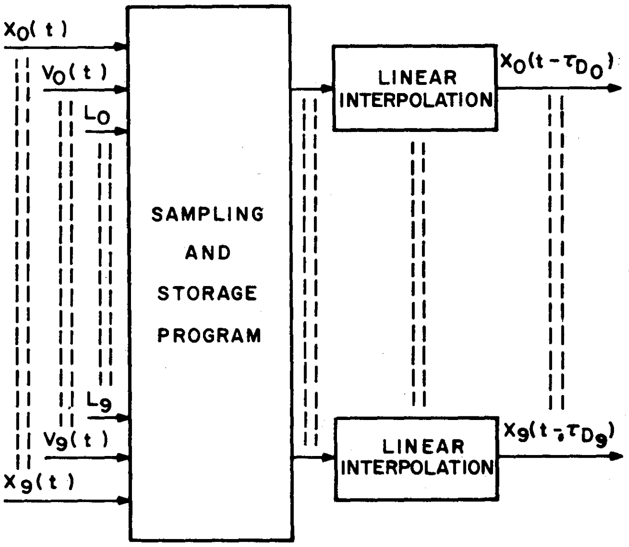


FIGURE 3: General Characteristics of HYDAC 2000 Transport Delay Program

Channels	5
Delay Time	0.6 sec. ± 0.1 sec.
Signal Frequency	10 cycles per second
Accuracy Required	0.5%

First, the number of storage words required per channel must be established. With linear interpolation, 0.5% accuracy for a sine wave can be obtained with 32 points per cycle. The maximum delay of 0.7 seconds at 10 cycles per second requires storage for approximately 7 cycles, so a minimum of 224 points is indicated for each channel. SM-8 Serial Memory Units, with 256 words, fulfill this requirement.

Second, one must check to see that the maximum conversion frequency will suffice. Since the converter-multiplexer runs at 4000 conversions per second, each of the five channels will be converted at a rate of 800 points per second. With an input frequency of 10 cps and 32 points per cycle, only 320 points per second must be converted, which indicates that the converter will be more than satisfactory.

Third, the access time of the serial memory must be checked to see that the data can be stored at the desired rate. With a maximum conversion rate of 320 points per second per channel, the period between samples is approximately 3 ms. Since the access time of the SM-8 is 2 ms. (See Appendix I), there will be no difficulty loading the serial memory.

In Appendix II, the $\omega\tau_D$ criterion for transport delay is described. A chart is provided also to determine the sampling rates for a desired delay for each of the serial memory units.

For comparison, one can determine the analog hardware needed to provide the equivalent delay. Thus, the use of a fourth-order Padé approximation for variable delay consumes 10 amplifiers, 8 potentiometers, and 4 multipliers to produce approximately 0.12 seconds of delay for the 10 cps signal at 0.5% accuracy. This HYDAC program, on the other hand, provides a total of 3.5 seconds of delay which would require approximately 30 fourth-order Padé's or 180 amplifiers, 240 potentiometers, and 120 multipliers. If the delay is fixed, the 120 multipliers can be replaced by 120 potentiometers.

GENERAL DESCRIPTION OF PROGRAM OPERATION

The basic operations performed by the program are shown in the block diagram of Figure 4. The pulse generator accepts an analog voltage proportional to the velocity, and generates the sampling rate, P.

The input variable, X(t), is sampled by a track-store amplifier, digitized by the time-shared analog-to-digital converter, and stored in the buffer and serial memory under command of the control block. The control also determines when the stored word is read out of the memory and interpolated to form the delayed output, $X(t - \tau_D)$. The operations required for a single channel of time delay must be duplicated for each additional channel. The time-shared portion of the program operates at a frequency of 4000 cycles per second, independently of the sampling rates of the individual delay channels.

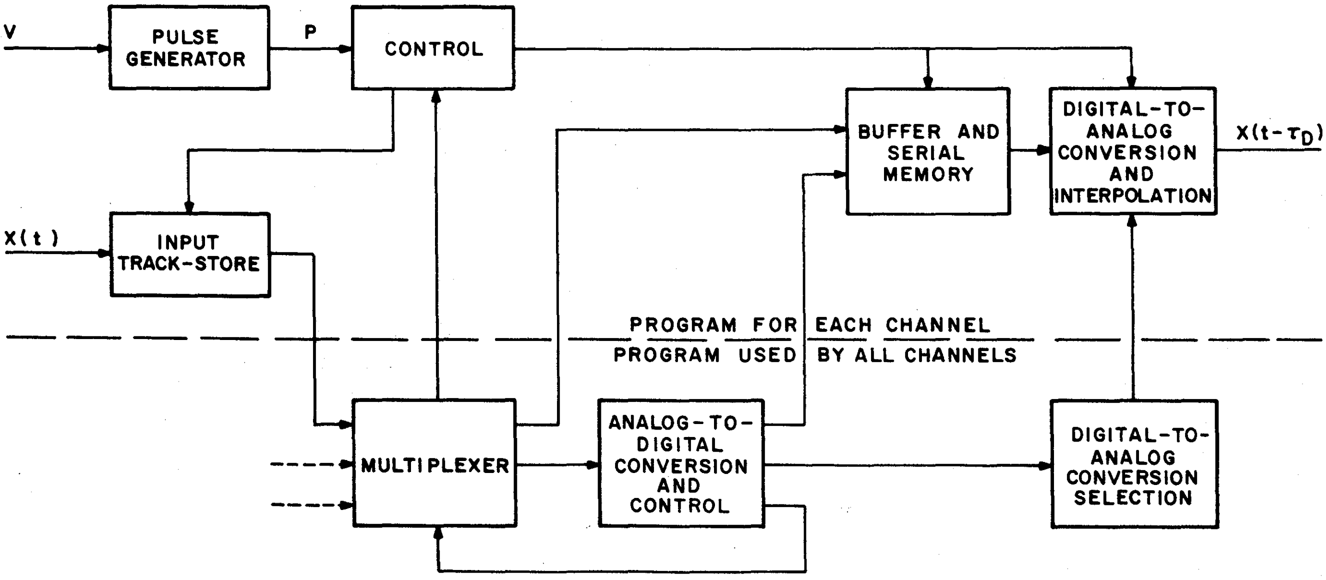

V
PULSE GENERATOR
P
CONTROL
BUFFER AND SERIAL MEMORY
DIGITAL-TO-ANALOG CONVERSION AND INTERPOLATION
X(t-τD)
X(t)
INPUT TRACK-STORE
PROGRAM FOR EACH CHANNEL
PROGRAM USED BY ALL CHANNELS
MULTIPLEXER
ANALOG-TO-DIGITAL CONVERSION AND CONTROL
DIGITAL-TO-ANALOG CONVERSION SELECTION

FIGURE 4: Block Diagram of Transport Delay Program

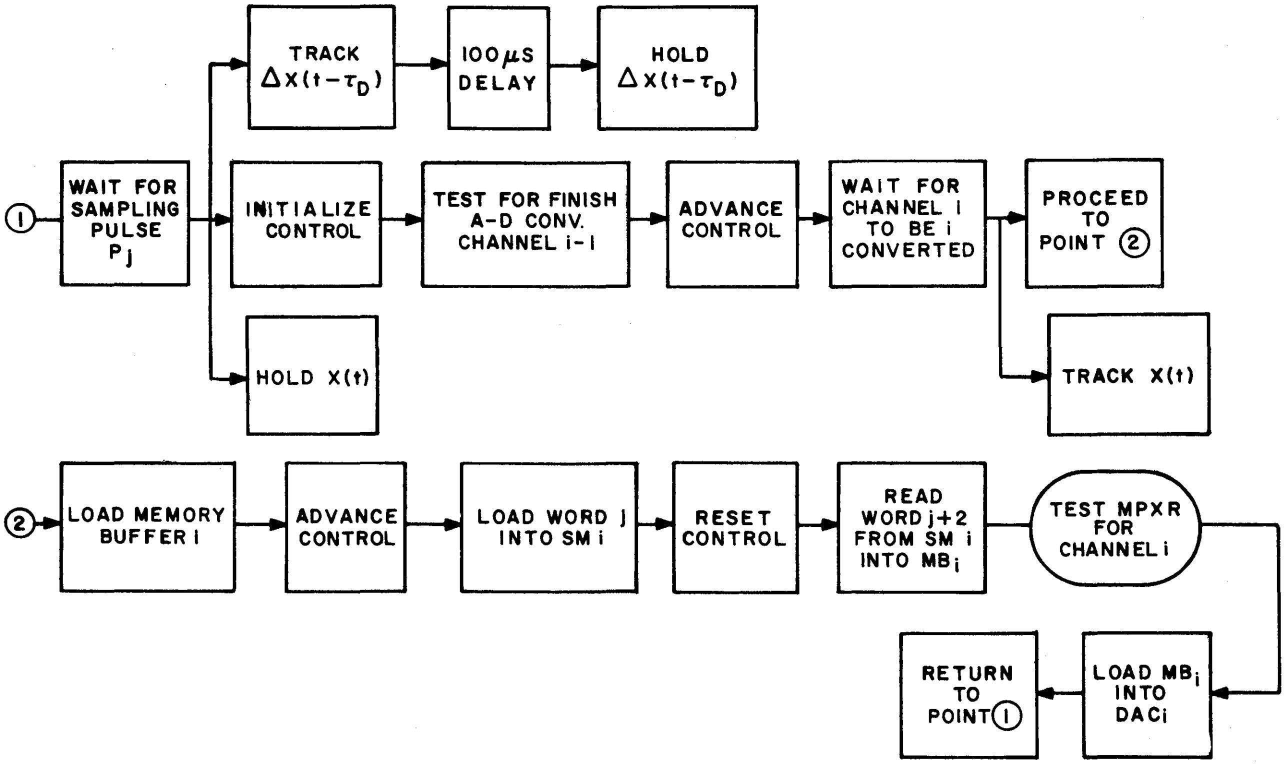
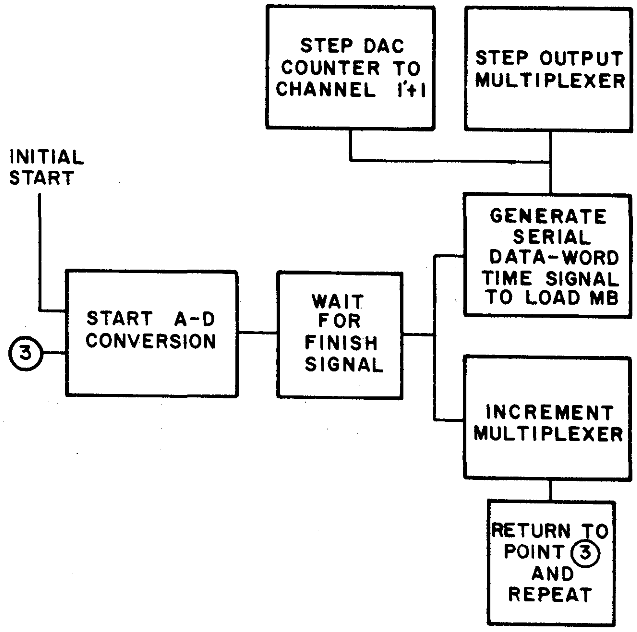


FIGURE 5: Sequence Diagram of Single Channel Control for Transport Delay Program.

The sequence of program operations for a single channel of delay is shown in Figure 5. When the pulse generator produces an output signal, P_j, three operations take place simultaneously: 1) the output interpolator tracks the incremental change in the delayed function, $X(t - \tau_D)$, for the interval j to $j+1$; 2) the analog value of the input function $X_j(t)$ is stored by the input track-store amplifier; and 3) the control block is initialized. A test circuit then determines when the analog-to-digital converter has completed conversion of the sampled value, and the control is then advanced to store this information. The input track-store amplifier is then allowed to track X(t), and the converted value $X_j(t)$ is loaded into the Memory Buffer (MB). The control is then advanced to transfer that word into the Serial Memory Unit at the proper word position. At this point, the control is reset, and the word two points ahead on the stored function, $X_{j+2}(t - \tau_D)$, is read out of the serial memory into the register of the digital-to-analog converter to provide the output interpolator with the correct point for the next sample pulse, P_{j+1}. The program then returns to the initial state in preparation for the next sample pulse.

The multiplexing and conversion operate in an independent mode, as shown in Figure 6. When a conversion is finished, the input multiplexer is incremented and a flip-flop set. A data word time signal is then generated to load the appropriate memory buffer. When the word has been loaded, the output multiplexer is advanced one step in preparation for the next conversion, and the DAC selection counter is stepped to prepare to read out the next stored point. The input multiplexer automatically starts a new conversion when the switching transient has disappeared.



STEP DAC COUNTER TO CHANNEL I+I
STEP OUTPUT MULTIPLEXER
INITIAL START
GENERATE SERIAL DATA-WORD TIME SIGNAL TO LOAD MB
3
START A-D CONVERSION
WAIT FOR FINISH SIGNAL
INCREMENT MULTIPLEXER
RETURN TO POINT 3 AND REPEAT

FIGURE 6: Sequence Diagram of Conversion Routine for Transport Delay Program

DESCRIPTION OF TRANSPORT DELAY PROGRAM OPERATION

The HYDAC program for transport delay is divided into three basic sections: 1) The conversion program which is time-shared by up to 10 channels, as shown in Figure 7a; 2) The Control and Memory program which is duplicated for each channel, as shown in Figure 7b; and 3) The Sampling and Interpolating Program which also is duplicated for each channel, as shown in Figure 7c. Certain portions of this program... such as the

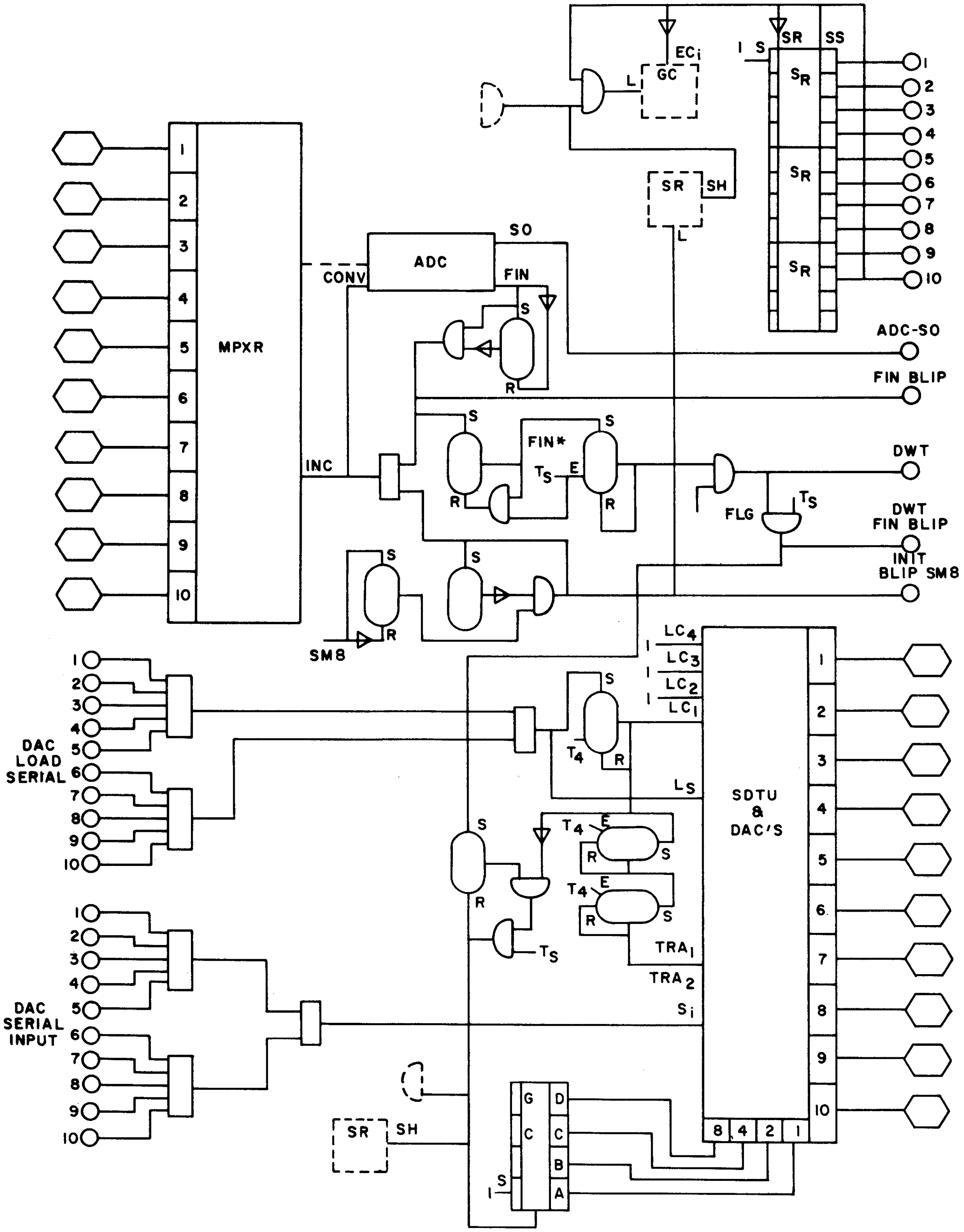


FIGURE 7a: Conversion Program

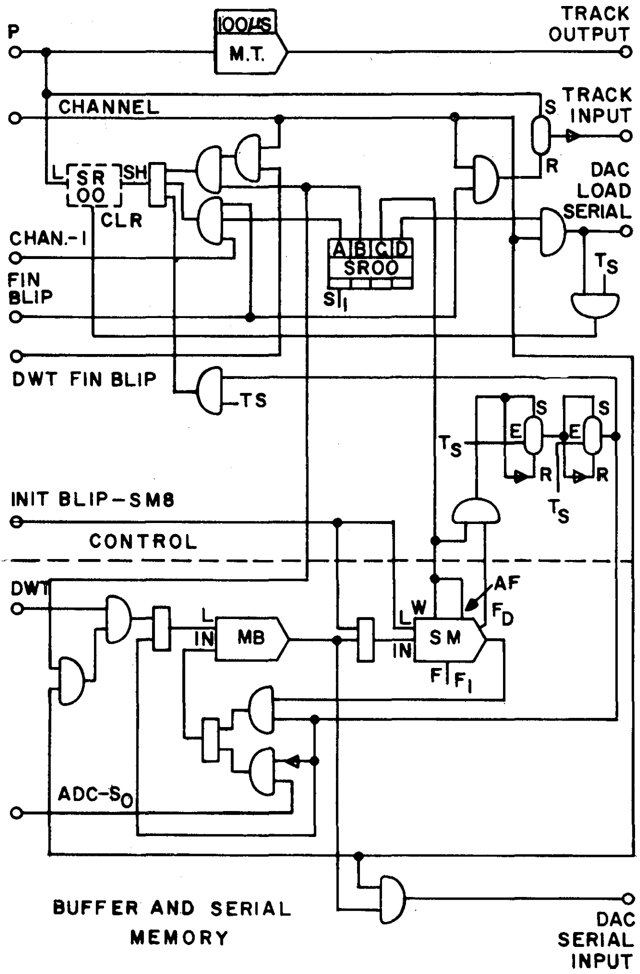


FIGURE 7b: Control and Memory Program per channel

Input Sampling and Output Interpolator... may be eliminated, depending upon the requirements of the problem. If only a fixed delay is desired, the Sampling Pulse Generator program can be simplified, as well.

Operation of the basic program functions are described briefly in the following sections.

Pulse Generator: Individual channel sampling rate signals are generated from voltage-to-pulse-rate converters that produce sampling signals, P, at rates proportional to their input analog voltages, V. This circuit, shown in Figure 7c, employs a standard analog summing amplifier, an integrator, and a potentiometer set to W_T/L. In addition, there is a HYDAC comparator, 2 D-A switches, and a logic differentiator circuit. The comparator output is low (binary 0) until the integrator output exceeds the bias voltage applied to the summer input, at which time it goes high (binary 1). The time required is dependent upon the product of the bias voltage, the pot setting (W_T/L), and the magnitude of the analog voltage representing the Velocity, V. A high signal at the comparator output causes the differentiator to produce an output signal, P, which is high for one clock time and, in addition, switches an opposite polarity bias to the summing amplifier by means of the D-A switch. The polarity of the rate voltage applied to the integrator is also switched, eventually causing its output voltage to be reduced, thus allowing the comparator output to go low, producing a second signal, P, which starts the cycle over.

Multiplexing and Conversion: When the sample rate signal, P, goes high, a flip-flop is set causing the complement of its output to go low. This switches the input track-store amplifier to hold X(t) at the value it had when P went high. When the input multiplexer is stepped to the channel, the ADC converts the sampled voltage to an equivalent 13-bit binary number (12 bits plus sign). When the conversion is completed, a high signal from the ADC causes a differentiator to produce a high signal for one clock time. This high signal

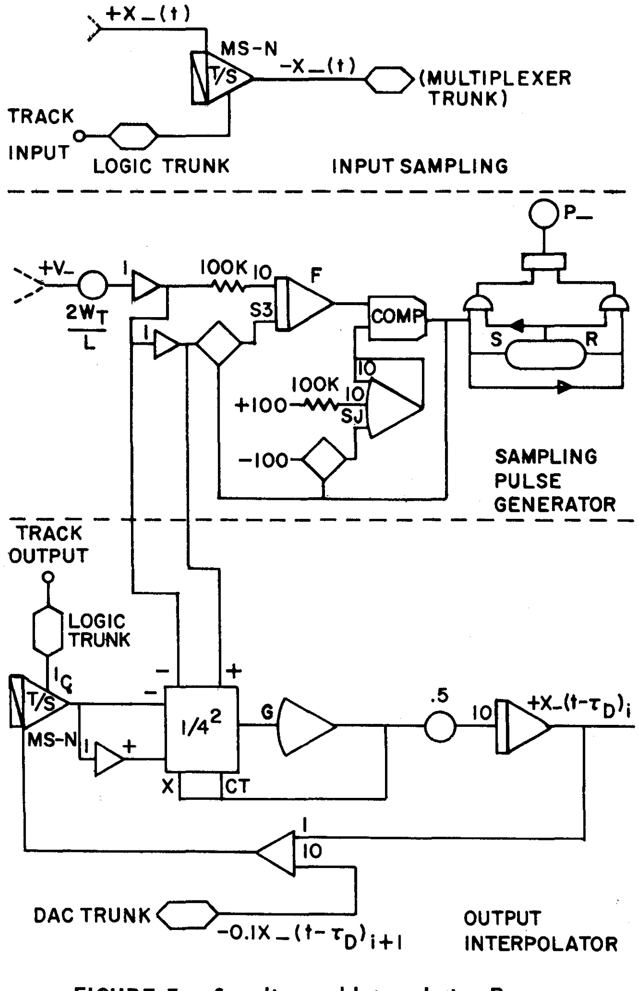


FIGURE 7c: Sampling and Interpolating Program for Channel

causes: 1) a flip-flop (FIN*) to be set to produce a high signal on its output; 2) the input multiplexer to be stepped to the next channel; 3) an indication to be given to the channel "control" circuit that the conversion has been completed; and 4) the input track-store amplifier to be returned to the track state by resetting the flip-flop originally set by the sampling rate signal. The high signal on the output of the FIN* flip-flop produces a signal on the output of a second flip-flop which is high for one serial word time. This signal, ANDed with $\overline{\text{FLG}}$, loads the converted value of the sampled input for the 13-bit data word time (DWT) from the ADC to the Memory Buffer selected by the output multiplexer. After a time interval equal to a serial word time, a finish signal (DWT FIN BLIP) steps the output multiplexer (shift register) to the next channel preparatory to the next conversion, and the DAC counter (GC01) to the next channel. A blocking operation is included if data is being loaded into the DAC. Only one of the shift register stages is active at a time. The active stage selects a memory channel corresponding to the selected input signal, determines which input track-store amplifier to return to the track state upon completion of the ADC conversion, and outputs the word two points ahead on the stored function from the serial memory to its memory buffer for transfer into the DAC channel. Each input shift pulse advances the active stage of the register one step. When the last stage is active, the shift signal transfers the active signal around the end, back to the first stage. Note that the multiplexing and conversion program operates independently of the timing of the rest of the program.

Control: The basic control of the program for each channel is performed by a shift register (SR00). The sampling routine is initialized when a high P signal causes the A flip-flop in the register to be set to the ONE state. After a conversion has been completed by the ADC and the output multiplexer has stepped to the channel, the output of an AND gate goes high, advancing the control by shifting the register from the A to the B stage. The output of the flip-flop in this stage goes to the ONE state, allowing the converted sampled value of the input to be loaded from the ADC to the channel memory buffer. After a time interval equal to one word time, the register is shifted from the B stage to the C stage by a high signal on the output of the second AND gate. The C stage flip-flop goes to the ONE state, permitting the word in the memory buffer to be stored in the serial memory, and advances the serial memory "flag" for readout of the next point. When the j + 2 word has been read from the serial memory into its memory buffer, the control is advanced to the D stage. When the output multiplexer returns to that channel, a word time signal is generated to load the memory buffer into the appropriate DAC channel. At the completion of this loading, the control register is cleared. The cycle is then ready to be repeated upon generation of the next sampling rate pulse.

Buffer and Serial Memory: Loading of the memory buffer with the converted sampled value of the input takes place when a high signal appears at the output of the AND gate connected to the LOAD input of the MB. After the word is loaded into the buffer, the control shift register is advanced to the C stage, which causes a high signal to be applied to the W and AF terminations of the serial memory. When the word position containing the F_1 "flag" is detected, the high W line causes the contents of the memory buffer to be loaded into this location in the serial memory (the function point originally stored in this position has been read out during the P_{j-2} pulse). At the same time, the high AF line advances the F_1 flag to the next word position to enable the storage of the P_{j+1} sample. When the word containing the F_1 flag is detected, the FLAGGED DATA line goes high, which initiates a logic and timing circuit which effectively counts two word times. After a time interval equal to two word times, the output of a flip-flop goes to the ONE state, allowing the memory buffer to be loaded with the word two points ahead on the stored function. This word is then held in the memory buffer until it can be loaded into the DAC channel. This value is then held in the DAC until the P_{j+1} sampling cycle, when it is used to interpolate the function between points j+1 and j+2. Note that no particular serial memory has been assigned. The choice will depend upon the total number of storage points required and the minimum delay. A description of the HYDAC Serial Memory Units, including the minimum time to load a complete function, is presented in Appendix I.

Interpolation: Reconstruction of the stored function is performed by a linear interpolation circuit. This circuit accepts the sampled data points from the DAC and converts them into a continuous analog function with straight line segments between points. The period between samples is a variable as defined by the following relationships from flow through a pipeline.

$$\text{Time Delay} = \tau_D = \frac{\text{Length of Pipeline}}{\text{Velocity of Flow}} = \frac{L}{V}$$

$$\text{Period between Samples} = \Delta t = \frac{\tau_D}{W_T} = \frac{L}{V\,W_T}$$

Then

$$\frac{d\,X(t - \tau_D)}{dt} = \frac{\Delta X(t - \tau_D)}{\Delta t} = \Delta X(t - \tau_D)\,\frac{V\,W_T}{L}$$

As shown in Figure 8, $\Delta X(t - \tau_D)$ is the difference between the j+1 point on the curve held in the DAC register (loaded into the DAC at the end of the j-1 sample), and the value of the function on the output of the integrator when the register contents are sampled by the track-store amplifier. Since W_T and L are known constants for a given problem, and V is the analog voltage controlling the duration of the delay, the rate of change of the delayed function can be formed by a multiplier and integrated to produce the function. A new value of $\Delta X(t - \tau_D)$ is obtained for each sampling pulse. A high signal to the input of a monostable timer produces a high signal which switches the track-store amplifier to the tracking state. This amplifier samples the difference between the voltage output of the DAC and the integrator to form $\Delta X(t - \tau_D)$ for the j to j+1 interval. At the end of a 100 microsecond delay (sufficient to enable the track-store amplifier to accurately sample the voltages), a low signal switches the track-store amplifier to the hold state, fixing $\Delta X(t - \tau_D)$ for the interval. The integrator then integrates the rate as determined by V to produce the function.

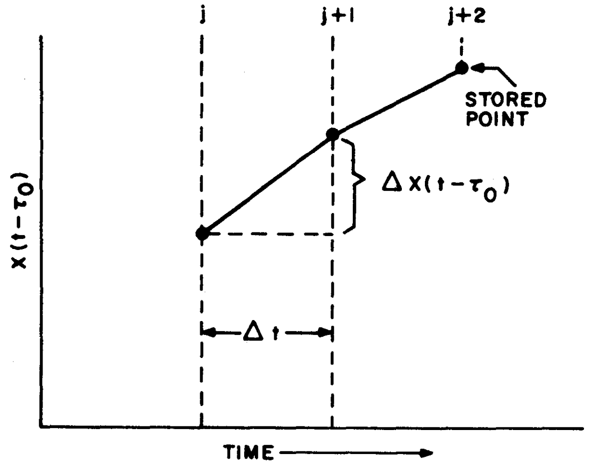


FIGURE 8: **Linear Interpolation Between Stored Points of Delayed Function**

Sample Results: A plot of typical results obtained on the HYDAC system using the Transport Delay Program is shown in Figure 9.

The input signal, $x_1(t)$, is a sine wave ($\sin \omega t$, with $\omega = 1$) modulated by a sawtooth wave of lower frequency. The signal $x_2(t)$ has the same character as signal $x_1(t)$, except that the frequency of both the sine wave and the sawtooth wave for the former are ten times higher. For the first channel, a sampling rate, P, of 10 points per second was used, while the second channel employed a sampling rate of 100 points per second. SM-8's were used for this sample program; τ_{D1} was 25.6 seconds and τ_{D2} was 2.56 seconds.

To illustrate the effect of variable delay, the velocity of both channels was reduced by a factor of two at the point indicated in Figure 9. Although a pure step change in velocity may not be realizable in physical systems, the simulated condition can be considered a worst case test. Since the fluid is incompressible and no mixing is assumed, the step change in input velocity caused an immediate reduction in the frequency at the output of the pipe. This new frequency is sustained until the input appears at the output with the new velocity. In Figure 9 it can be seen that channel one had not yet, at the end of the recording shown, reached the new steady state.

CONCLUSIONS

A general purpose HYDAC program for multichannel variable transport delay has been described. The major features of this program, in terms of capacity and versatility, include:

- The provision of up to ten independent channels of delay
- The availability of storage capacities of 16, 64 or 256 words per channel
- The capability to obtain a wide range of delay times since each independent channel can accept a different velocity variable
- The extension of this wide-range-delay-time capability since each channel can use a Serial Memory Unit of different capacity
- For example, the capability to provide delay from 25 to 800 milliseconds at 0.5% accuracy for signal frequency of 10 cps
- Finally, through the use of digital storage, the capability to simulate very long delays such as the pure delay of a high frequency pulse through a loseless medium, or that encountered when the velocity of flow approaches zero.

This hybrid transport delay simulation program is one of a series of standard general purpose programs which are available as "software" for the HYDAC System. Other programs, including one for multi-tapped transport delay of one function for use in correlation analysis, are available or are scheduled for inclusion in the overall HYDAC "software" capability.

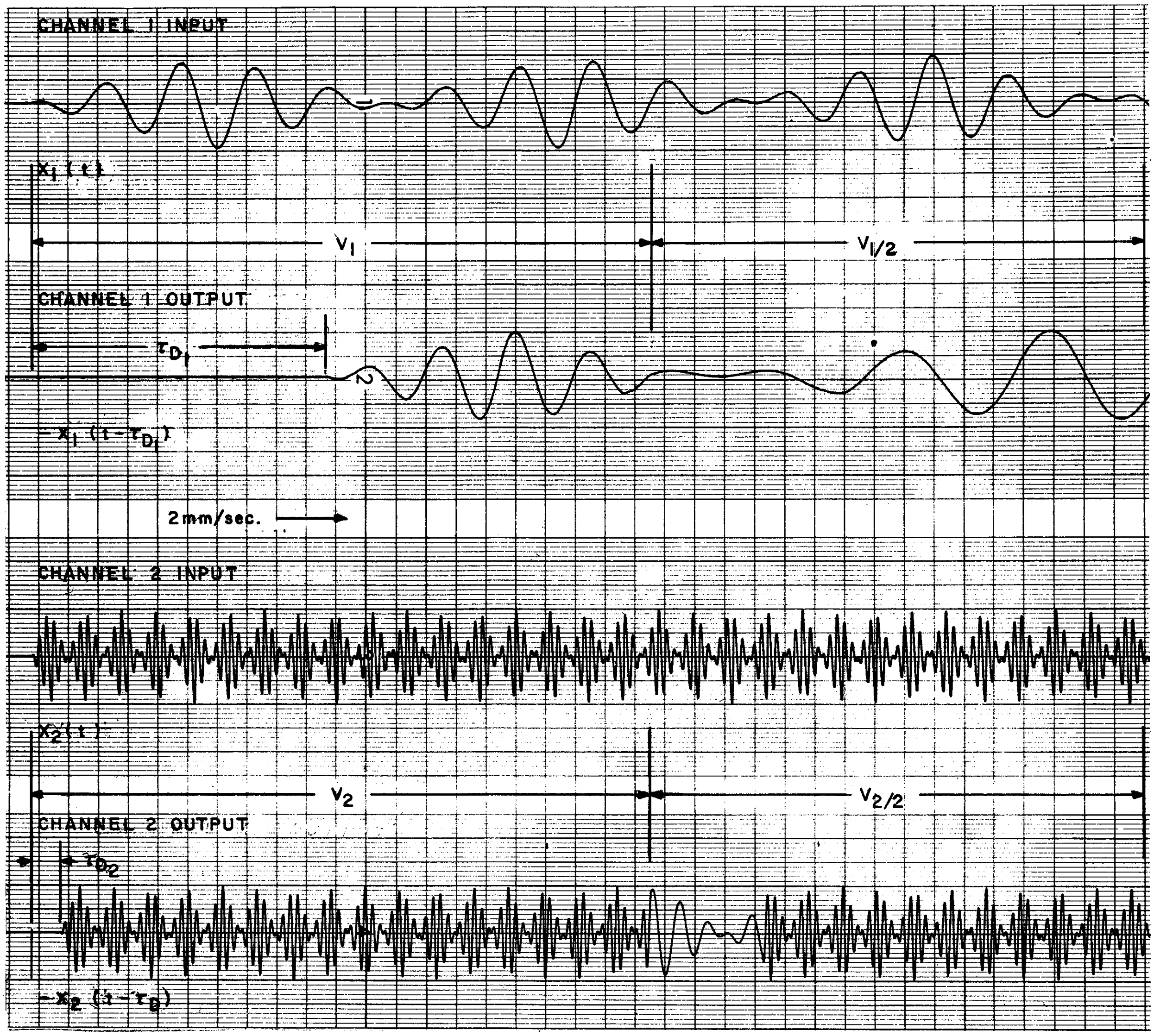


FIGURE 9: Sample Results for Two Channel Transport Delay Including Effect of Velocity Change

REFERENCE

(1) Rogers and Connolly: Analog Computation in Engineering Design, McGraw,-Hill Book Co., Inc., New York, 1960, pp 419-424.

- APPENDIX I -

GENERAL DESCRIPTION OF HYDAC SERIAL MEMORY UNITS

The Serial Memory Unit used to store X(t) in the transport delay program can be described as a device for storing a table of numbers, or digital "words", that represent a sequence of sampled values of an analog signal. All numbers are stored as serial binary words of 16 bits each. Each unit has its own control circuitry for writing, addressing, and reading the tabular values in the proper sequence. The three sizes of Serial Memory Units are the SM-8 with a capacity of 256 words, the SM-6 with 64 words, and the SM-4 with 16 words. Each size of memory unit has a different Read/Write Cycle rate. This rate is the maximum number of data words per second that, in the normal mode of operation, can be written into and read from memory in sequence. Any lower rate can be used, as may be required by the analog sampling rate. The entire stored table of numbers can be read rapidly from memory in a single Read/Write cycle but, in normal use, a single word is selected from the table and a new word written into the table in the same position during a cycle. The selected position is then moved to the next in the sequence.

The tables of data words are addressed by a moving index "flag". The flag is moved forward one step each Read/Write Cycle during which the AF (Advance Flag) control line is energized. The flag may be moved in the reverse direction, one step per cycle, by the RF (Reverse Flag) control line. If the AF line is energized all the time, the entire table will be addressed at the full Read/Write cycle rate (512 cps, 2048 cps, or 8192 cps), and thus the flag will sweep the entire function table in the minimum sweep time (0.5 seconds, 31 milliseconds, 2 milliseconds). If the flag is moved beyond the last word position in the sequence, it steps to the first position.

The tabular data words selected by the moving flag are "read out" by forming the logical product (with an AND gate) of the data output and the FD control line. The FD (Flagged Data) line provides an enabling signal long enough to pass the serial bits of the indexed or flagged word, one each Read/Write cycle. Thus, reading from the memory is possible without employing any specific "read" control signal.

New data is written into the memory at the flagged position whenever the W control line is energized and the new serial data is available at the input.

The F (Flag) control line permits the simultaneous use of as many as three independent flags in the Serial Memory Unit. Thus, in special programs, it is possible to move these flags back and forth independently to select the desired position for reading or writing. The L (Load) control line is similar in function to the W line but is used to pre-store or initialize the memory unit with special data, such as initial flag locations.

The range of memory sizes and maximum Read/Write cycle rates satisfies a wide range of problem requirements. For transport delay programs, the required minimum sampling rate and the delay time determine the choice of memory unit. For function storage and generation, the SM-8 unit will handle a 256-point function in as short a period as 0.5 seconds. Thus, for real time programs the SM-8 is very satisfactory; however, for high speed repetitive programs the SM-6 is better, since it can be used with a time base as short as 31 milliseconds. The SM-4 is useful mostly for very short delays and with other units. It must be noted that function tables of any size can be programmed simply by cascading Serial Memory Units. If different size units are employed, the lowest maximum Read/Write Cycle rate applies to the combination.

To apply a figure of merit to different methods of simulating transport delay, the product of the frequency and the time delay for a particular accuracy is generally used. This product — $\omega\tau_D$ — represents the storage capacity of the time delay, or the phase shift in radians which one can obtain with a particular transport delay system. The fourth-order polynomial approximation provides approximately 7.5 radians of delay with a one-degree phase error which corresponds to approximately 0.5% error in amplitude. To obtain the same accuracy with the serial memories in this HYDAC program, 32 points per cycle must be stored using linear interpolation of the output function. Therefore, one can establish a maximum $\omega\tau_D$ product for each serial memory unit.

$$\text{SM-8:}\ \omega\tau_{D_{max}} = 2\pi\left(\frac{W_T}{32}\right) = 2\pi\left(\frac{256}{32}\right) = 50.2 \text{ radians}$$

$$\text{SM-6:}\ \omega\tau_{D_{max}} = 2\pi\left(\frac{\dot{W}_T}{32}\right) = 2\pi\left(\frac{64}{32}\right) = 12.5 \text{ radians}$$

$$\text{SM-4:}\ \omega\tau_{D_{max}} = 2\pi\left(\frac{W_T}{32}\right) = 2\pi\left(\frac{16}{32}\right) = 3.1 \text{ radians}$$

By this criterion, it is obvious that the SM-8 is the most powerful unit. However, the larger storage unit is most useful only for the longer delay times, i.e. those exceeding 0.5 second (there is no inherent upper limit to the delay). If the minimum delay time is 0.5 second, the maximum analog frequency for this unit, at 0.5% precision, is:

$$\omega\tau_D = 50.2 \text{ radians}$$

or

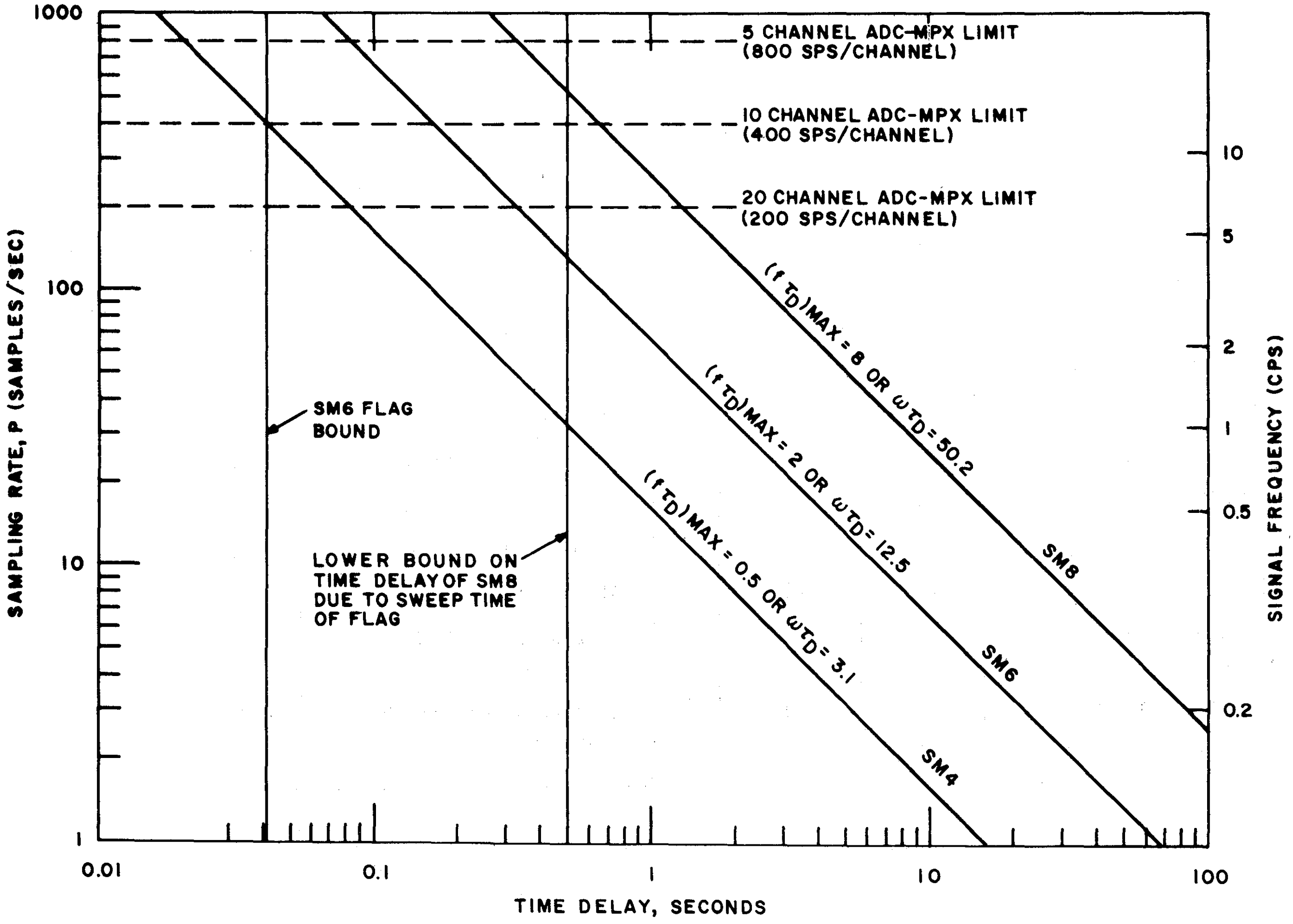


FIGURE 10: Sampling Rate vs Time Delay for HYDAC Serial Memories (32 samples per cycle of input signal minimum is assumed)

$$f_{max} = \frac{50.2}{2\pi \tau_D} = \frac{104}{6.28} = 16.6 \text{ cycles/sec}$$

For shorter delays and higher frequencies, the SM-6 unit provides a minimum delay of 0.031 seconds. In this case, the maximum analog frequency is:

$$f_{max} = \frac{12.5}{2\pi \tau_D} = 64 \text{ cycles/sec}$$

The corresponding frequency for the SM-4 is approximately 256 cps.

The following equation relates sampling rate, P, to time delay, τ_D:

$$P = \frac{W_T}{\tau_D} \quad (W_T = \text{number of words in SM unit})$$

Taking the log of both sides

$$\log P = \log W_T - \log \tau_D$$

Figure 10 shows this relationship as a straight line on log-log paper. Also included in this graph are the conversion rate limits and the sampling rates required to provide 32 points per cycle for various analog input frequencies. This graph permits the programmer to determine rapidly what Serial Memory Unit to use for a particular application and the allowable analog frequency.

To illustrate the use of the chart, the sample requirement described under Program Capabilities will be used. With a signal frequency, f, of 10 and a $\tau_{D_{max}}$ of 0.7 second, the $f\tau_{D_{max}}$ is 7, which is below the SM-8 and above the SM-6 indicating that the SM-8 unit must be used. Entering the chart on the 0.7-second delay line, a sampling rate, P, of approximately 370 points per second is found for the SM-8, which is well below the five-channel ADC-MPX limit. Since this point is over the f=10 cps line, an accuracy somewhat greater than 0.5% will be obtained. Thus, the chart permits the programmer to compare conveniently the requirements of the problem with the capabilities of the program, to determine the size of memory required and whether time-scaling of the analog is indicated to fulfill the requirement.

- APPENDIX III -

The complement of HYDAC 2000 components required to implement the Transport Delay Program can be summarized as follows:

Components	Number for One Channel	For each Additional Channel
Track-Store Amplifier	2	2
Analog Integrator	2	2
Analog Multiplier	1	1
Analog Summer	6	6
Potentiometer	2	2
AND gates	40	20
General Purpose Flip-Flop	13	4
Digital Comparator	1	1
Digital-Analog Switch	2	2

Components	Number for One Channel	For each Additional Channel
Serial Memory Unit	1	1
Memory Buffer	1	1
Quad Shift Register	4	1
General Purpose Counter	1	0
Monostable Timer	1	1
Multiplexer	1	0
Analog-to-Digital Converter	1	0
Digital-to-Analog Converter	1	1

12

EAI®

ELECTRONIC ASSOCIATES, INC. *Long Branch, New Jersey*

ADVANCED SYSTEMS ANALYSIS AND COMPUTATION SERVICES/ANALOG COMPUTERS/HYBRID ANALOG-DIGITAL COMPUTATION EQUIPMENT/SIMULATION SYSTEMS/SCIENTIFIC AND LABORATORY INSTRUMENTS/INDUSTRIAL PROCESS CONTROL SYSTEMS/PHOTOGRAMMETRIC EQUIPMENT/RANGE INSTRUMENTATION SYSTEMS/TEST AND CHECK-OUT SYSTEMS/MILITARY AND INDUSTRIAL RESEARCH AND DEVELOPMENT SERVICES/FIELD ENGINEERING AND EQUIPMENT MAINTENANCE SERVICES.