

INTERFACE DESIGN FOR DIGITALLY CONTROLLED TRAINERS

MR. WYNN GAYLOR

Chief of Simulation, Hydrosystems, Inc.

A. INTRODUCTION

Design of interfaces for Digitally Controlled Trainers will be discussed in the light of experience gained during design and manufacture of several Submarine Control Trainers. The Interface is defined as that equipment which translates the real world (analog and discrete data at the training station) into digital formats acceptable by a general purpose computer and vice versa.

The alternatives of synchronous versus asynchronous operation of the interface will be explored and related to appropriate determining criteria. System input/output requirements such as number of inputs and outputs, updating rates and solution rates are tabulated for determination of a timing comparison between synchronous and asynchronous interfaces. Programming implications of synchronous versus asynchronous interfaces will be examined, and this will be followed by a description of the hardware and computer requirements for each type of interface.

The question of built-in test equipment (BITE) versus program diagnostics will be explored by considering the requirements imposed by the need to debug, test and adequately maintain the interface. Emphasis will be placed on design features of BITE which enable thorough debugging and testing of the interface independently of the computer.

The last aspect of interface design to be discussed is suppression of Electro-magnetic Interference (EMI). Problems posed by signal transients due to indicators and switches, power transients, and radiation and susceptibility of long lines will be discussed. An overall approach and philosophy of EMI suppression will be presented.

B. SYNCHRONOUS VERSUS ASYNCHRONOUS CONSIDERATIONS

The synchronous interface is under direct computer program control for each individual input or output. The asynchronous interface has its own Input/Output processor which operates on blocks of data, independently of the central processor. However, the central processor still selects the block, specifies the length of the block, and determines when data transfer is initiated.

Which approach should be selected?

In order to explore this question several assumptions are necessary regarding the system's requirements such as number, types, and rates of inputs and outputs. Table 1 lists typical aircraft trainer and submarine ships' control center trainer I/O requirements. The I/O requirements for each type of trainer, aircraft and submarine, are translated into 24-bit words. This discussion assumes a computer, I/O bus and/or processor which handles 24 parallel bits. Discrete I/O is packed one discrete per bit or 24 discretely per word. BCD I/O assumes an average of 3 decimal digits requiring 12 bits for each word. Each analog input or output is represented by 12 or 13 bits, plus a sign bit, per word. Each synchro output is equivalent to 2 DC analog words, representing sine θ and cos θ . The total shows that 174 words of I/O data are required for aircraft trainers and about 77 words for the submarine trainers. These are extremely small requirements for modern computer equipment.

Solution rates relate to computer program stability, which is determined by the mathematical model frequency of response characteristics and by the numerical integration methods used within the digital computer program.

Table 1. Typical Aircraft and Submarine Training Requirements

| DESCRIPTION | AIRCRAFT TRAINER | SUBMARINE TRAINER |
|------------------------------|---------------------------------------|---------------------------------|
| DISCRETE | | |
| INPUTS = SWITCHES | 500 = 21 WORDS | 170 = 8 WORDS |
| OUTPUTS = LAMPS | 150 = 7 WORDS | 120 = 5 WORDS |
| OUTPUTS = BCD DISPLAYS | 500 = 41 WORDS | 160 = 14 WORDS |
| ANALOG | | |
| OUTPUTS = METERS OR SYNCHROS | 70 = 70 WORDS | 40 = 40 WORDS |
| INPUTS = CONTROLS | 35 = 35 WORDS | 10 = 10 WORDS |
| TOTAL 24 BIT WORDS | 174 | 77 |
| SOLUTION RATES | 20/10/5/1 = AVERAGE \cong 5 PER SEC | 8/4 = AVERAGE \cong 4 PER SEC |
| UPDATING RATES | 20/10/5/1 = AVERAGE \cong 5 PER SEC | 8/4 = AVERAGE \cong 4 PER SEC |
| | a 5 X 174 \cong 1000/SEC | 4 X 77 \cong 300/SEC |
| | b 20 X 174 \cong 4000/SEC | 8 X 77 \cong 600/SEC |

Updating rates relate to the fidelity of display in the real world, and are determined by response characteristics of the human observer and the instrumentation observed. In addition, the closed loop response requirements between a trainee command activation and the system response must be realistic. For example, when a switch is activated, the closed loop response may be the activation of a lamp. Any system delay must be duplicated. This is especially important where the delays are short yet still observable. The interface updating rate should be at least as high as the computer solution rate associated with the variable being observed. Instructor station inputs and BCD displays may be exempted from this rule; these rates may be comparatively low, e. g., 1 per second.

Aircraft trainer computer solution rates vary from a maximum of 20 per second downward with an average rate of about 5 per second. The synchronous I/F permits updating each time there is a solution. Updating at the maximum solution rate is desirable for all interface inputs or outputs in the asynchronous interface, to be described later.

In the submarine trainer, all the solution rates and updating rates were kept at 4 per second, except for one system which required both at 8 per second. Solution rates for many of these systems could have been lower. However, this was not necessary since the computer speed was more than adequate to handle these requirements.

Approximately 1000 inputs and outputs per second are required for a single cockpit aircraft trainer; a submarine trainer's requirements are considerably less. Considering a 2 microsecond cycle time machine, about 50 microseconds per input or output operation is a realistic average number both for interface conversion and for I/O program execution. Approximately five percent of the computer time is used for the aircraft trainer I/O operation and about one and a half percent of the computer time is used for the submarine trainer I/O operations in the synchronous approach. These results indicate that the synchronous approach is not too time consuming, particularly in the submarine trainer, since 5 percent time use of a 2 microsecond computer may be acceptable.

The asynchronous approach should use the 20 per second updating rate for the aircraft, equivalent to the maximum solution rate, under a "stealing" mode. About 4000 inputs and outputs would take approximately 10 milliseconds including block set-up times. A comparison of the two methods indicates that the asynchronous approach is five times faster in transferring data to and from the interface. The question arises, is the extra speed necessary? This depends on the other computational requirements and the computer selected. It has been our experience that the standard available machines can easily handle all requirements including the five percent or the one percent as the case may be, for input/output via the synchronous approach.

The programming approach for the asynchronous interface is more complicated. A start-up program is necessary to input data to the main programs before any computations start in order to prevent erratic behavior on the first pass through the computations. A set-up routine for each type of input or output is required to determine the block of data, set-up I/O interrupt linkages, arm and enable I/O interrupts, perform readiness tests, and activate the cycle stealing. Interrupt service routines and interrupt acknowledge routines must be entered after each block of data transfer is completed. Since this complicates the executive control program, it becomes convenient to process all I/O at the highest solution rate, rather than in groups of I/O at their individual lower solution rates. Using the highest solution rate as the updating rate is important because computation in the present cycle will use input data from the last updating cycle since the new inputs have not yet been distributed. This could cause unrealistic responses at lower updating rates. Even though all I/O is at 20 updates per second, the distribution and gathering of the data for the programs may be done at the individual solution rates. This is important because these programs consume much more time than the basic input/output of the data from the interface. If recovery from a failure such as the absence of a ready signal is required, special interrupt logging routines become necessary. Absence of a ready signal is a problem because the computer is working with blocks of data rather than individual inputs and outputs. These comments indicate some of the more obvious effects of synchronous versus asynchronous interfaces upon programming. From a programming point of view, the simpler synchronous approach appears more attractive as long as I/O requirements permit.

Hardware requirements are much the same for both the synchronous and asynchronous approaches, except for the need of an Input/Output processor with the latter. A more important observation is that a less sophisticated computer handles the synchronous interface design, and therefore the interface approach exerts a major influence on the choice of the computer. In particular, hardware required for communication with the I/O processor is unnecessary and the interrupt system becomes a single real-time clock input.

From a maintenance point of view, the more direct activation of the synchronous interface by use of function words and the individual character of the data transfers simplifies the design of built-in test equipment, i. e., individual words may be processed through the interface independently of the central processor.

C. BUILT-IN TEST EQUIPMENT VERSUS PROGRAMMED DIAGNOSTICS

This brings us to the second portion of this paper, Equipment versus Programmed Diagnostics. Since both of these factors are important, the discussion becomes one of priority. The most important questions to be answered are:

1. Which approach will provide the most modular checkout and testing capability for the prototype and also allow implementation of standard testing procedures for production models?

2. Which approach provides the best methods for maintenance of the interface?

The hardware test panel gets first priority when one considers these questions, especially for the prototype. The capability to test and debug the interface independently of the central digital computer is highly desirable. It allows simultaneous testing of central computer programs, the interface, and the real world equipment. The order of combinations of these major subsystems is by ascending order of complexity, real world to interface first and then the computer. Prototype errors associated with the computer programming may be extremely difficult to detect and correct since they may involve actual mistakes within the mathematical model as well as programming bugs. However, since the built-in test equipment has already verified the operational status of the interface and real world equipment, the problem of testing the computer as part of the integrated system is greatly simplified. Production problems, on the other hand, are minimal with the computer subsystem since the follow-on computers may be programmed identically to the prototype unit. Complete checkout of the real world to the interface subsystem is necessary for both prototype and production models and can be accomplished effectively by an interface test panel (Figure 2). The decision to provide built-in test equipment must be made early and, if affirmative, this equipment must be carefully designed into the interface subsystem. In addition, careful attention must be given to a design which lends itself to the implementation of program diagnostics later. Fortunately, the design requirements for a good test panel leads to easy implementation of program diagnostics, which in essence can be thought of as the automation of the manual procedures required in the use of the test panel. The most important additional design requirement for programmed diagnostics is provision for disconnection of the real world and the addition of a closed loop at the interface connection to the real world. In this way, the computer commands will result in a feedback which can be measured. An alternate open loop approach enables introduction of fixed known inputs to the interface for comparison with data stored in the computer. Computer generated interface outputs are checked by comparison of display readings with the predicted results.

Figure 2, the Interface Test Panel, implements the capability for complete checkout of the interface either independently or in conjunction with the real world subsystems. All function and data words may be set up exactly as they would appear as outputs or inputs to the digital computer. Basically, the panel provides the capability to execute one instruction at a time independently of the computer. Since these instructions may be executed in any arbitrary order or groups, this approach provides flexibility in testing not practical via the computer. The advantage of interface program diagnostics will be reviewed later in this paper.

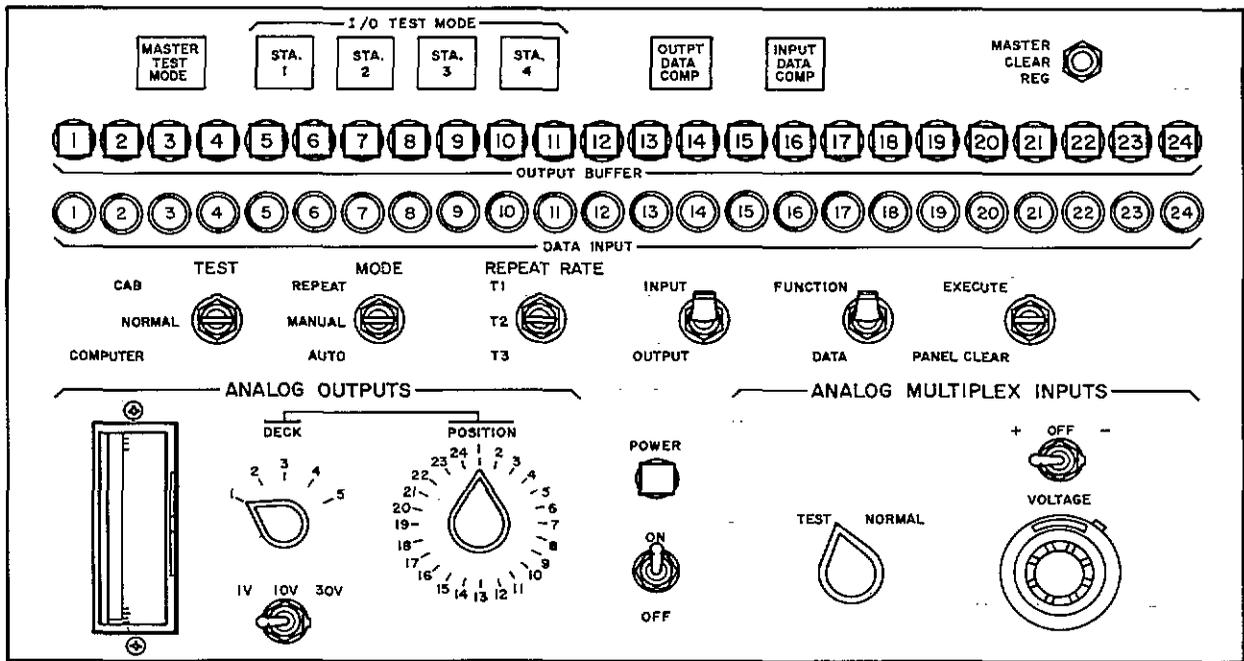


Figure 2. Interface Test Panel

The following example demonstrates how an analog input may be tested. On the lower right of Figure 2, is a section called Analog Multiplex Inputs. The VOLTAGE switch is positioned to + and the TEST-NORMAL switch is set to TEST. (The NORMAL position is used for accepting inputs directly from the real world.) The TEST potentiometer is set at any desired voltage. The TEST switch is set to CAB and the MASTER TEST MODE red light comes on indicating that the interface has been disconnected from the computer and connected in a test mode. The MODE switch is placed in REPEAT and the REPEAT RATE is set to T1, T2 or T3, depending on the updating rate desired. The INPUT-OUTPUT switch is set to INPUT and the FUNCTION-DATA switch is set to FUNCTION. The function word is set up on the DATA INPUT pushbuttons. This is the same bit configuration which would be sent from the computer, and any address may be selected to determine the exact path through the interface. The next step is to momentarily activate the EXECUTE-PANEL CLEAR switch to EXECUTE. The function word will set up the proper gating structure and appear transferred to the OUTPUT BUFFER indicators for check purposes. Next the FUNCTION-DATA switch is placed in the DATA position and the EXECUTE switch is again activated. This will cause the analog voltage to be inputted

on the proper addressed line to the analog-to-digital converter, converted and displayed on the OUTPUT BUFFER indicators. Other input or output transfers may be activated in a similar manner. The most important maintenance features of the panel are:

1. The ability to set up a repetitive operation which allows maintenance personnel to track the signals through the system.
2. The ability to exercise any path by addressing it.

Programmed diagnostics become more useful after a system is completely checked. The first module of the programmed diagnostic is the test portion which should be used as part of an overall morning readiness test. If the test provides any indication that a problem exists in the interface, the diagnostic portion of the test may be called and executed after the interface has been disconnected from the real world and put into an automatic test mode. These diagnostics would narrow the interface problem down to a particular path by printing out the results of all tests and flagging any failed channel or channels. The maintenance personnel may then quickly set up the failed path on the interface test panel and track down the problem by repeated activation of the problem path. Programmed diagnostics, therefore, become a significant time saver in determining if a problem exists and the approximate cause of trouble.

As final comment, most maintenance personnel have a hardware background and it has been our experience that this type of test panel is in their words "just what the doctor ordered". They use it extensively on the ships' control center trainers. A test program is fine, but ultimately the exact point at which a wire is broken, etc. must be isolated and built-in test equipment is the ideal solution.

D. ELECTROMAGNETIC INTERFERENCE (EMI) SUPPRESSION CONSIDERATIONS

The phenomenon of Electromagnetic Interference is a major problem to be considered in interface design. This problem is generally alleviated by techniques of filtering, shielding, ground isolation, and structural mechanizations, which help minimize pickup (susceptibility) and generation (radiation) of interference. Each of these techniques must be considered and each could easily be the subject of a comprehensive study. Filtering is of special importance and a design approach is presented.

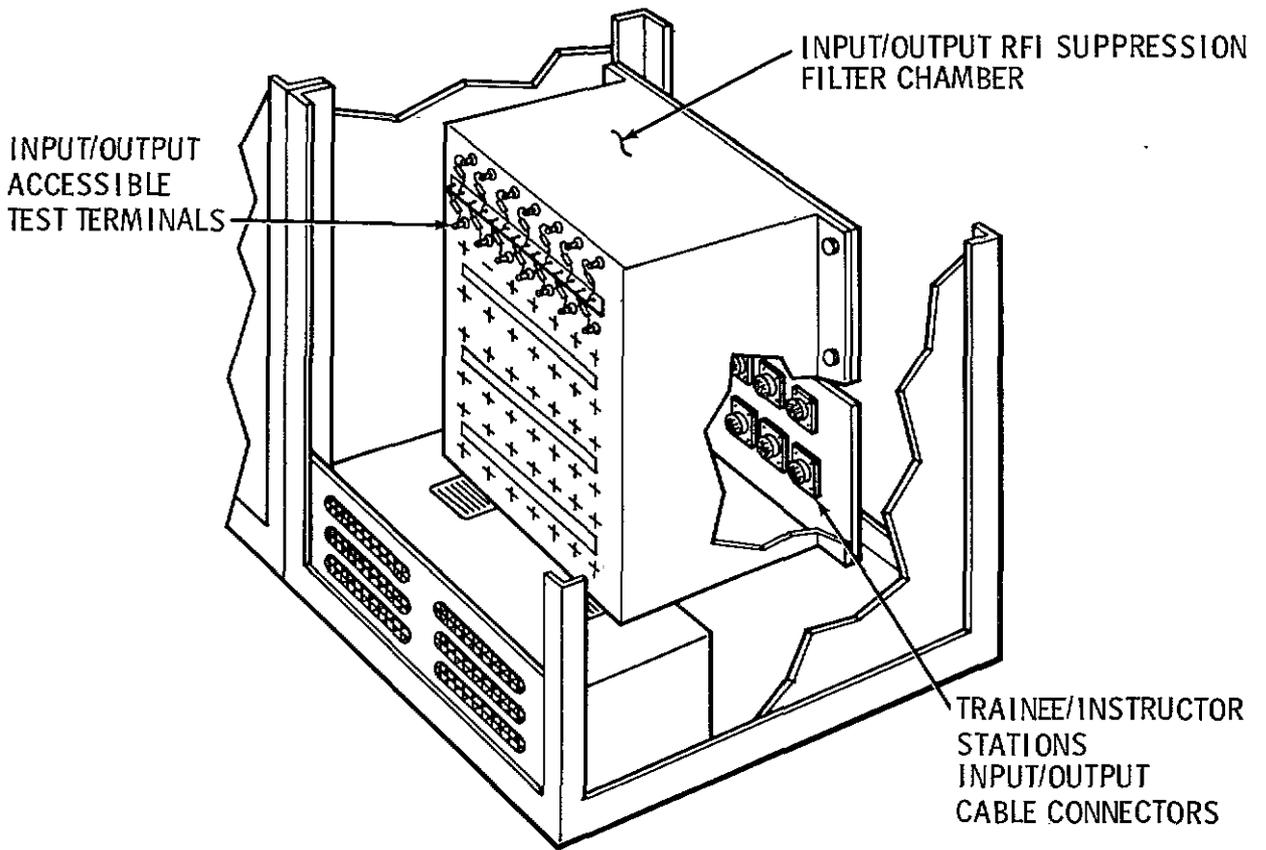
Predominant sources of trouble are the real world inputs to the interface such as switch signal transients, signal noise pick up over the long lines, and power transients due to motors, generators, air conditioners and other intermittent high load power demands. Signal noise inputs to the interface picked up on the long lines originate when the interface output lamp drive turns on a cold lamp. Abnormally strong interfering fields couple through the lamp driver output lines into the input lines. Also, conducted interference will be picked up by the long transmission line in the interface output due to interfering fields coupled through the lamp driver power, which is supplied to other electronic equipment, thereby affecting their normal operation.

Approximately 90 percent of the command and response signals to and from the real world are discretely, namely switches and indicator lamps and these have been shown to be critical sources of EMI.

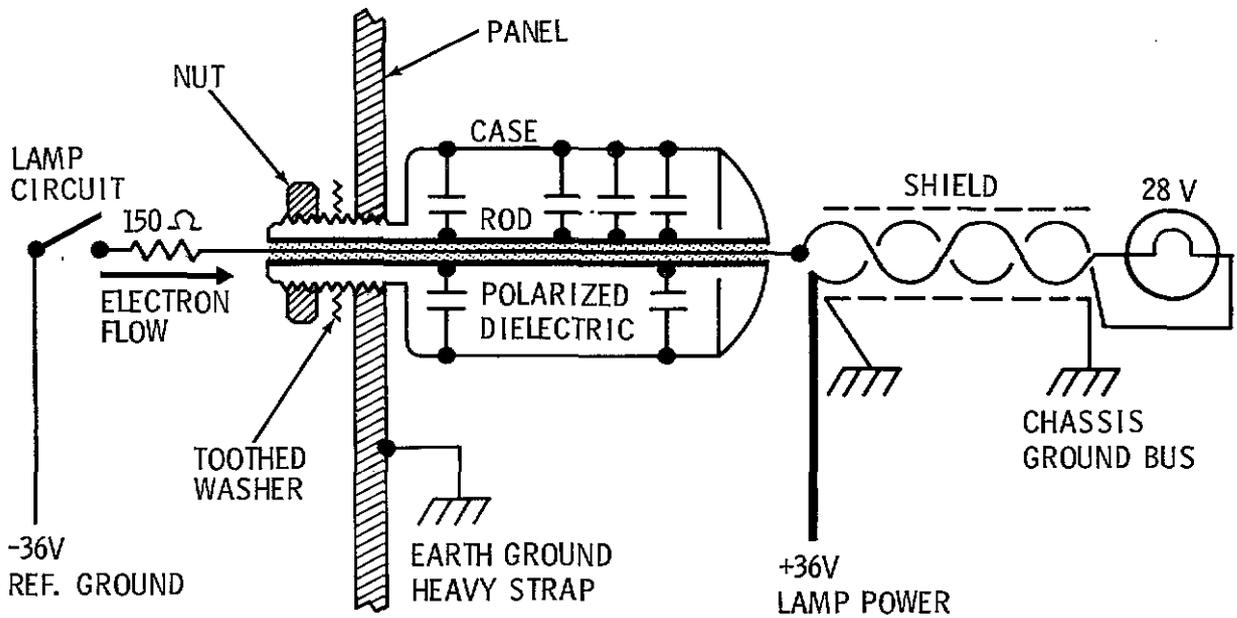
In the design of the Submarine Diving Trainers, it was decided that all input and output signals entering or leaving the interface on the real world side of the trainers would be fully protected against EMI emanating from our system and possible pickup of strong radiation from nearby electrical systems which we could not control. More than 700 individual filters were provided for the ships' control center trainers to provide protection on all lines except the BCD output display lines whose signal pulse shapes must be maintained. Figure 3(a) illustrates the EMI suppression system supplied for the ships' control center trainers. The real world connects through the rear of the cabinet. Maximum filtering effectiveness requires the minimizing of impedance common to the noise source and the output terminals. Therefore, feed-through capacitors were mounted in a metallic barrier (panel) forming one wall of the filter chamber.

High frequency current is shunted directly to earth ground in the circuit shown in Figure 3(b). Other circuits similar to the one shown in Figure 3(b) but tailored to each particular type of problem were provided.

The signal lines terminating in the filter panel form a matrix of 24 x 32 points, and the panel lends itself ideally to doubling as a junction or distribution panel. Any group of signals can be selected to be cabled through the connector and routed to a specific row of terminals. For example, the 24 bits of a computer word will be in one row in the identical positions as in the computer word, allowing extremely quick identification and coordination with the computer program. Since the filter panel is readily accessible by simply opening the cabinet front door, all terminating filters are available as test points which can be used in conjunction with the built-in test equipment.



(a) EMI Supression System



(b) Typical Lamp Circuit

Figure 3.

E. CONCLUSION

Three general interface design topics have been considered. Specific design approaches and their implementation have been recommended based on experience with digitally controlled submarine ships' control center trainers. In general, these recommendations would also apply to aircraft trainers.

1. Synchronous input/output techniques provide adequate speed, and permit the use of simple programming and less sophisticated hardware than asynchronous techniques.

2. Built-in test equipment which enables repetitive testing of each interface path expedites interface testing as a subsystem module for both the manufacturer and the user. This is of special benefit to user maintenance personnel.

3. A centralized electromagnetic suppression module located at the interface connection to the real world provides improved electromagnetic interface control in place of the prevalent, scattered individual circuit type of approach. Also, the module provides an extremely useful test point matrix for maintenance.

F. ACKNOWLEDGEMENT

The author thanks Messrs. F. Gray, J. Berrio, and K. Schantz of Hydrosystems for their helpful suggestions in preparation of this paper.

REFERENCES

1. F. Platt, "A Technique for Defining the Interface between Digital Equipments"; presented at Polytechnic Institute of Brooklyn, Graduate Center, Farmingdale, N.Y., June 4, 1968.