IOS HARDWARE USAGE AND TRAINING EFFECTIVENESS

THE BEST APPROACH

Catherine Meyn
Computer Scientist
Tactical and Training Systems Division
Logicon, Inc.
San Diego, California

ABSTRACT

Appropriate selection of hardware can simplify simulator operation as well as increase training effectiveness. Generally, flight simulation groups are aware of this and plan accordingly. Too often, however, hardware is selected without a complete front end evaluation of the functional requirements of the application. The effective use of hardware in the development of an instructor/operator station (IOS) is just as important as its selection. This fact is often overlooked, partly due to the lack of money applied to this area and partly to an assumption that requirements from one simulator to another remain the same.

Most flight simulators are built as if the simulator and the IOS have the same requirements. That is, the entire system is built around the concept of tight cycle times, generally ranging from 20-50 milliseconds (ms). Furthermore, the IOS will tend to resemble a cockpit in the sense that numerous repeater gauges are used. For simulator realism, these concepts are very important. However, an IOS has very different requirements from the flight simulator. The IOS does not require tight cycle times, nor does it require real-time operation, except for certain monitoring capabilities. The IOS should be an instructor support work station. It should not be used merely to control the simulation, but should enhance the training task, relieve the instructor of tasks unrelated to the training function, and collect data that is necessary in evaluating student progress. An IOS requires instructor-oriented control, flexibility, and meaningful output.

Typical IOSs consist of function keyboards, CRTs, repeater gauges, and occasionally hardcopy. More modern systems include high resolution color graphics and digitized speech hardware. While these trends are encouraging, the effective use of this hardware still lags behind current training requirements, causing a reduction in the training benefits which state-of-the-art hardware can provide.

This paper considers several types of IOS hardware devices in terms of their potential to support the training task. It also discusses optimal ways to use this hardware to provide the appropriate instructor support while maximizing training effectiveness.

BACKGROUND

In the past, simulation/IOS design was constrained by the inadequacies and restrictions of the hardware/system software available. For example, only one program could run in a computer at a time. This meant that the simulation and the IOS had to be highly integrated to provide control of the simulation while still maintaining cockpit operation fidelity. Because memory was limited, highly compressed code had to be written to provide complete functionality. In addition, instructor inputs were very primitive, relying on teletypes and hardwired push buttons to control the system. Due to these limitations, the primary focus of simulator and IOS design became the simulation. Training support was virtually ignored, because the hardware simply could not support the instructor effectively. The IOS became primarily a control device.

As new hardware devices developed, the focus of design gradually expanded to encompass both the simulator and the IOS. Currently, the IOS is beginning to receive more attention and new hardware configurations are being presented to improve IOS operation and increase training effectiveness. Unfortunately, the general approach in hardware acquisition has been to procure equipment without understanding how to use each device effectively in the training environment. Thus, some of the new simulators have three input devices (among them keyboard, push buttons, touchpanel, light pen or joystick), three output devices (CRT, graphics display and printer), and a degree of complexity previously unheard of. In this situation, the instructor has to be an operator/engineer/mathematician just to accomplish simulator operation. Unfortunately, instruction remains a parallel task. This is not to say that selection of state-of-the-art hardware is not appropriate. *The problem is a lack of understanding* of just what is needed and how it can be used most effectively. Some of the issues that should be considered in selecting and using IOS hardware are discussed in this paper.

IOS REQUIREMENTS

Because of hardware restrictions in the past, the IOS and simulation have been designed as if they had the same requirements. In fact, the IOS and the simulation have quite different requirements, and hardware *and software* design should reflect this. A simulator must respond to pilot inputs in a realistic fashion. The visual scene must be smooth and continuous. The cockpit instruments must reflect the aircraft and vehicle system states accurately and with strict adherence to aircraft fidelity. These requirements have traditionally required short interval timing loops, high speed computers and minimal I/O. On the other hand, the IOS should monitor selected events in the simulation. It should respond to instructor inputs in a timely fashion. It should also maintain and present student records, course syllabi, and statistics on demand. These requirements require large amounts of computer memory, flexible asynchronous functions, disk I/O, and instructor-oriented control.

To assess these very different requirements, the IOS and simulation should be different programs. Cost and spare requirements tend to require these programs to reside in one computer. Ideally, the two programs should reside in different computers, linked with a universal interface to provide maximum flexibility. A universal interface would allow different computers to be used on either side. In this way, hardware selection could focus on the actual application rather than on the idiosyncracies of individual computers.

INPUT DEVICE SELECTION

There are many types of input devices available for IOS operation. A brief overview of these devices and their potential for supporting the training task are described below.

Keyboard

The traditional typewriter-style keyboard has been in use for some time. This device was useful because it permitted any number of different commands. Unfortunately, to be completely effective, it requires the operator to be able to type while performing other tasks. In addition, it diverts the instructor's attention away from the student and requires extensive error checking capabilities. It is useful for systems support and debugging but has drawbacks that hinder its effectiveness in the training environment.

Function Keyboards

This type of keyboard has specially made keycaps for each function designated in the IOS. It generally provides one button operation which is certainly an improvement over the traditional keyboard. Unfortunately, there is no way to determine what function is valid at any time. Error checking is required and a message must be generated to provide the instructor with feedback in the event of an invalid input. Worse, in many cases invalid inputs are ignored in the interest of saving code, leaving the instructor futilely pounding the keys in an effort to initiate an unavailable function. The function keyboard also inhibits expansion. Spare keys must be provided and new keycaps generated whenever a new function is implemented.

Pushbuttons

These buttons are generally built into the IOS console. They are very useful for emergency operations that require immediate action such as motion abort or emergency power off. For more extended uses, however, they are subject to the same limitations as function keyboards.

Joystick, Trackball

These devices can be useful for defining waypoints or tactical elements of a mission. Occasionally they are used by the instructor for flying the simulator in a demo mode. Using a joystick for this function requires practice to ensure a smooth flight. Joysticks and trackballs are not useful for precise data input and therefore should not be used as primary input devices.

Mouse

This is a small box connected to the display. As the box is moved over a flat surface (such as a table top), a cursor on the graphics display follows its movement. The mouse has 3 programmable function buttons that can be selected independently or in "chords" for increased functionality. The mouse can be used for menu selection or to input precise data.

Light Pen

The light pen was designed to take advantage of a menu-driven system. A menu-driven input system has real advantages in IOS operation. The approach allows instructor-familiar language to be used on flexible graphics formats. As training requirements change and new functions are added, the menu database can easily be changed as well, requiring only minor software or data modification to remain current. Basically the instructor uses a light pen to select elements from a menu on a display screen. The concept is a good one, but there are tremendous limitations in this implementation. For example, ambient light can affect the acceptance of a light pen input. Each screen requires its own light pen because of the individual idiosyncracies of the pen/screen pairs. Thus, if multiple input screens are required, it may be difficult logistically to make valid inputs. Light pens can also be lost or broken. Generally they must be tethered to the output display. Ignoring these problems for a moment, a menu driven system is a good idea, and hardware exists that can provide such a facility with fewer problems.

Touchpanel

Touchpanels are becoming more widespread in simulation/IOS control. A touchpanel can either be overlayed on a graphics display or be a self-contained package. Generally, touching a point on the screen closes a circuit or breaks a light beam and the position selected is stored for validation by the software. The touchpanel provides the flexibility of a menu-driven system without the problems typical of light pen devices.

Programmable Pushbuttons

This device is relatively new, but provides a good compromise when the more flexible touchpanel displays are not feasible. Basically, this system consists of pushbuttons whose legends are software controlled alphanumeric or graphic data. Only the buttons which are valid during the selected state need be imprinted and the legends can be blinked for feedback or to attract attention.

Speech Recognition

This area is still very new. It permits control of the IOS by simple vocal instructions. When the technology reaches maturity, it will probably be a very effective approach to IOS control and training. At this time, however, there are several reasons why it may not be entirely appropriate for an IOS. For example, each user must provide individual voice patterns for the speech recognition device to use in evaluating voice commands since no two people have the same voice patterns. If the tolerance is set low enough to recognize two different people using the same voice patterns, invalid commands might be recognized with unpredictable results. A second problem is the lack of consistency of language in a training environment. To use speech recognition for control in an environment where the language is not rigidly specified is not generally feasible at this time.

OUTPUT DEVICE SELECTION

Although there is less variety in the major areas of IOS output devices, some consideration needs to be given to what is most useful for any IOS application. Often the most elaborate output system is not the most effective. On the other hand, too limited a capability can hamper the instructor in the training task. A brief description and evaluation of currently available output devices is presented below.

Aircraft Gauges

Often the traditional IOS relies on aircraft gauges to provide the instructor with cockpit information. These indicators are usually identical to those used in the actual aircraft. Although these indicators are certainly realistic and familiar to the instructor, they often provide little or no instructional support. These devices require constant scanning to determine potential problems or inappropriate actions. Their maintenance schedule is low priority behind actual aircraft. Often they break or are borrowed to repair working aircraft or the simulator. They are more expensive and more difficult to use than their graphics counterparts. Although some gauges are necessary to provide complete information to the instructor, a good graphics display can provide the data of many aircraft indicators without the potential problems.

Printer

Printers provide hardcopy output. They can be very useful for providing student records, flight data and syllabus information for postmission analysis. They are not particularly useful for real-time mission data analysis. There are many types of printers available. Color and graphics printers are available in addition to high speed laser printers. The need for these advanced technologies is very dependent on the application. Therefore, a careful evaluation of the requirements of a particular system is vital.

Alphanumeric Display (CRT)

The alphanumeric CRT has become the primary output device for the traditional IOS. It provides alphanumeric text and has been used extensively for data display. Modern CRTs have some graphics symbology and are more analogous to graphics display systems. In general CRTs do not provide the flexibility available to a good graphics system.

Graphics

Graphics provide a useful source of information for the instructor. Graphics can be used to supply aircraft positional information, diagnostic feedback and display online/offline technical information related to the mission performed. Graphics combined with touch sensitive panels can also provide software controlled touch inputs, very useful in a complex IOS. There are several issues associated with graphics technology which impact the IOS.

Display Technology. There are two general ways to display graphics. Stroker or calligraphic technology involves moving a beam to discrete, predefined points to draw the vectors associated with a display. Raster displays perform a horizontal line by line update of the entire display. Although stroker technology is still used on some simulators, it is, for the most part, obsolete. Stroker displays are generally monochrome, although there are color stroker systems available. Stroker technology has potentially more resolution than raster, but raster is cheaper, more flexible, and reflects the current technical trend. Until recently, only two-dimensional presentation was available. There is now one 3-D system whose display is impressive. At this time, however, there are severe constraints in viewing positions, amount of displayable data, cost factors, and maintenance that limit 3-D application. Appropriate use of 2-D shading and colors can provide an effective simulation of 3-D graphics.

Resolution. Generally the higher the resolution or precision of a display, the better the picture and the more it costs. Most state-of-the-art applications use high resolution graphics. Low resolution graphics are difficult to read and often tiring. Resolution is defined by the number of lines discernable in one plane by the number of lines discernable in the other. A resolution of 512 x 512 is the minimum acceptable resolution for an IOS. 1024 x 1024 is the best resolution readily available.

Color. Color has become a key hardware issue in recent years. The number of colors used at one time is a major question. For most graphics applications, eight to sixteen colors is popular and sufficient. Color can be of great benefit in enhancing instructor support. Appropriate use of color to identify key elements in student training has only recently been provided in modern systems and has proved to be remarkably useful.

Voice Output

Voice output can be very useful in supporting the instructor. For example, a voice output device can simulate air traffic controllers, boom operators, and other outside agencies that usually require the instructor or other students. Voice output can also be useful in instructorless training, to inform the students of potential problems during a practice mission. There are three types of voice output generally available. The first type, tape output, is generally not very useful except for replay or demonstration purposes. Because tape output cannot be accessed randomly, it provides little realism during a real-time mission. The other two methods of voice output both provide the capability for selection of any pre-recorded phrase at any time. These methods are phoneme voice output and speech digitization. In the phoneme method of voice output, the phrases are defined by the use of phoneme codes, which the output device generates as sounds. In the speech digitization method, actual voices are recorded for storage on disk. Of these two methods, the phoneme method generally requires less disk space, but it requires an expert to program it. The result is intelligible but lacks realism. The speech digitizer provides more natural, intelligible phrases and generally is worth the extra space it requires.

FINAL SELECTION

Selection depends strongly on the application. For this reason functional requirements should be established well before a specification is generated. Functional is the keyword here. Too often requirements are presented with rigid definitions as to the capabilities of hardware devices. Because there is generally a substantial amount of time between when a specification is written and when it is bid, new technology may arrive in the interim which cannot be used due to the rigidity of the definition. In many cases, the result is a system that is obsolete by the time it is completed. For example, a display might be specified at 1024 x 1024 pixels of resolution with a 60 Hz refresh cycle, when what is actually desired is the highest resolution available, continuous lines with minimal stairstepping and no flicker. Should a better system be developed, it is effectively excluded from the competition. Obviously risk must be taken into consideration before selecting any form of new technology. However, a functional approach can protect the end user while still providing the capability for the best system possible.

HARDWARE USAGE

For purposes of illustration, a candidate system is described below. An operational flight trainer (OFT) IOS is specified. The hardware purchased for this device includes four color graphics displays with touchpanel overlays, two minicomputers to drive them and provide instructor support, two disk drives, one tape drive, one printer with graphics capabilities, and a speech digitizer.

The first step is to determine which hardware is useful in which IOS function. Some of these decisions are self-evident. For example, a tape drive will be useful in development and for saving student records and flight data, but would be ineffective for data retrieval in real-time online operations. In this example, one minicomputer will contain graphics and touchpanel related software and the other will contain instructor support software. This will permit the graphics system to run independently, and reduce the disk access required on the instructor support computer. The printer will be used for generating student reports and hard copies of pertinent graphics displays. The printouts will be used to supplement the instructors' logs and provide readable charts for later review. Usually these papers will not be examined until debriefing. The speech digitizer will be used to simulate air traffic controllers and the copilot and to coach the student in the instructorless practice mode. To provide realism and prevent confusion, a distinct voice will be used for each individual simulated. Two of the graphics displays will be used for real-time mission control, and two will be used for scenario generation and debriefing. This approach will prevent valuable simulation time from being wasted with offline functions.

Once the IOS configuration has been established, more specific data should be considered. For example, display formats should be carefully defined. The instructor must be able to find required data quickly. Data should be organized in meaningful groups. In this example, communication and TACAN-related data will be displayed in one area of the display and aircraft configuration data in another. These groupings will remain consistent regardless of the number of displays to prevent confusion.

To improve readability and ensure quick recognition, windowing and icons can be used. Windowing permits relevant data to be highlighted without removing background data (see figure 1). Icons are immediately recognizable pictures that represent actions and objects to the computer user. Icons take advantage of the symbology already familiar to the system's intended users and conveys more information than can be meaningfully displayed in standard text. (Again, see figure 1.)

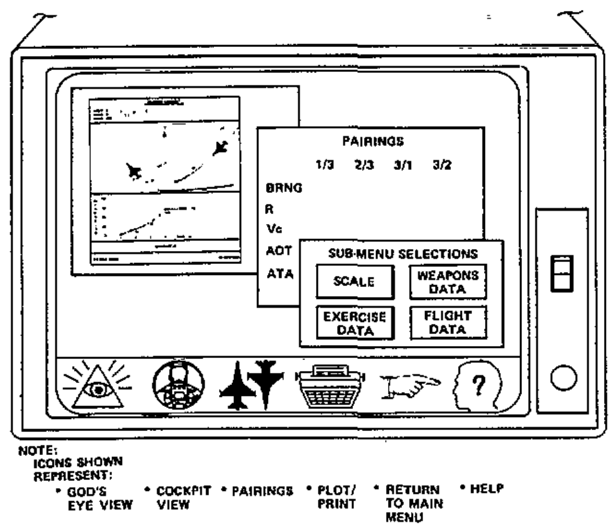

Figure 1. Windowing and Icons

Only one primary input device per work station should be necessary. The instructor should not have to search for the control menu. In this case, one touchpanel per station will be used for input. The number of inputs required to invoke a function will be severely limited. During real-time mission control, two menu levels should be the deepest an instructor will have to go from the primary runtime menu. The legends for the touchpanel should be in user-oriented English or graphics symbols. There should always be a return touchpad on each menu to allow the instructor to recover gracefully from unintentional inputs. Invalid functions should not be available on any menu and touchpads should always be the same size and color for quick recognition.

Color should be used for clarification and differentiation. For example, on a menu, different colored backgrounds might reflect different menu levels. Thus, the primary menus would have blue backgrounds, secondary menus green, and tertiary menus would have yellow backgrounds. Color on both the input and output displays should be relevant. Used properly, color can provide the instructor information quickly and effectively. Aircraft flight data might be presented in green when the aircraft is engaged in normal flight, in yellow when its stress boundaries are neared, and in red for critical conditions. Blinking hypercritical data could provide a secondary signal to alert the instructor to problems. In extreme cases, the speech digitizer could be used to alert both the instructor and the student to critical situations.

Color should not be used as a substitute for well designed displays. It can and should be used to enhance such displays.

CONCLUSIONS

Previous flight simulator/IOS hardware configuration designs have been restricted by the inherent limitations of the hardware used. Hardware advancements have reduced or eliminated many of these restrictions, but IOS design has remained the same. Future IOS designs must take new technologies into account to ensure a usable, state-of-the-art IOS.

After a hardware configuration has been determined, its functional application in the IOS should be carefully evaluated in terms of benefit to the end user, control requirements, and instructor support. Selecting and using appropriate state-of-the-art hardware doesn't have to be difficult. It requires creative thought, front end evaluation and design, and an understanding of the end user's requirements. With these considerations, a hardware configuration can be provided which will be used for many years.

ABOUT THE AUTHOR

Cathy Meyn graduated from San Diego State University with a double degree in Computer Science/Psychology. She has worked for Logicon, Inc., in San Diego as a Computer Scientist for the past six years, concentrating on instructor support in the flight simulator. Prior to that, she worked for the Navy Personnel Research and Development Center in San Diego designing and implementing adaptive testing and computerized counseling software.