Naval Command,
Control and Ocean
Surveillance Center RDT&E Division San Diego, CA
92152–5000

Technical Report 1538
December 1992

Surface Computer System Architecture for the Advanced Unmanned Search System (AUSS)

M. E. Kono

Approved for public release; distribution is unlimited.

Technical Report 1538
December 1992

Surface Computer System Architecture for the Advanced Unmanned Search System (AUSS)

M. E. Kono

NAVAL COMMAND, CONTROL AND
OCEAN SURVEILLANCE CENTER
RDT&E DIVISION
San Diego, California 92152–5000

J. D. FONTANA, CAPT, USN
Commanding Officer

R. T. SHEARER
Executive Director

ADMINISTRATIVE INFORMATION

The work was performed by members of the Ocean Engineering Division (Code 94), Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA 92152–5000. This work was funded under program element 0603713N, project SO397.

Futher information on this subject is available in related reports that represent NRaD efforts through FY 1992. The bibliography is found at the end of this report.

Released by
N. B. Estabrook, Head
Ocean Engineering Division

Under authority of
I. P. Lemaire, Head
Engineering and
Computer Sciences
Department

DM

EXECUTIVE SUMMARY

OBJECTIVES

The original Advanced Unmanned Search System's (AUSS) surface console computer was a unique system built by a contractor and based on Multibus I bus architecture. The contractor had to be consulted before functional changes to the console could be made, and hardware add-on options for the system were limited by the Multibus I bus architecture. These disadvantages defined the following objectives for the AUSS surface console redesign effort: increase system reliability by basing the design on hardware and software components that were widely used (and thus tested) in the commercial sector; minimize software development and maintenance costs; and allow for the evolution of hardware and software components as technology advances.

RESULTS

The redesigned AUSS surface control van has been based on the IBM 7552 PC, a commercially available computer. The PC architecture has an enormous installed market base, making hardware add-on components widely available. The market dominance of Microsoft's DOS has standardized the operating system software. These two factors provided a huge base of software tools and add-on hardware peripherals, which have in turn allowed personnel to build highly specialized systems that could meet almost any functional need from low cost, high volume commercial products. The surface control van computers' functions have been split among several machines to provide for some modularity of the software design and accommodate possible increased processing demands. The current AUSS surface computer architecture has readily accommodated changing requirements; tasks have been decoupled such that the resultant software is easier to maintain and evolve.

RECOMMENDATIONS

The basic AUSS surface computer should be redefined to be an X Window Platform. Multiple, specific display cards must be replaced by a single virtual display to eliminate the dependency of the display system on a proprietary display card and its software library. The virtual display must be able to manage multiple graphics windows on a single screen to reduce the wiring supporting the AUSS's large number of monitors. This virtual display must also be able to remap windows related to an application. The X Window System, which is an architecture that promotes machine independence, can provide this virtual display environment. The X Window System can supply graphical interfaces locally at a single machine or across a network.

CONTENTS

FIGURES

w
ethe
DOS s
of a hug
allowed pe
functional ne
with this appro

BACKGROUND

The original Advanced Unmanned Search System (AUSS) used a variety of computers and approaches to address the needs of the unmanned undersea vehicles (UUV)/surface ship search system (figure 1). The primary computer was the surface console computer. This computer was based on the Multibus I bus and its central processing unit (CPU) was a 10-MHz Intel 8086. Intel RMX was used for an operating system. The system was a one-of-a-kind unit because it was built to specification by a contractor. Contained in the system were several custom cards. The program was custom written by the contractor to Naval Ocean Systems Center (NOSC) specifications in PLM, a very efficient, high-level language designed to provide speed and small code size benefits similar to those derived from assembly language. This console design served the original AUSS through its first life. As is typical, actual use and time on the system made numerous new features not originally envisioned desired updates. The main shortcoming of the original design became evident: it was extremely difficult to upgrade and essentially frozen. I believe that the decision to procure the unit on contract was made because of expediency and limited in-house manpower resources. However, this decision meant any functional changes to the console necessitated returning to the original contractor. Since even simple engineering changes to the software became a procurement process, updating the console in a timely and cost effective manner was impossible. In addition, since the hardware design was based on the Multibus I bus architecture, few options for adding hardware functionality to the system were available.

The lessons learned from the original AUSS surface console design defined the following objectives for the redesign effort:

- Increase system reliability by basing the design on hardware and software components that were widely used (and thus tested) in the commercial sector;
- Minimize software development and maintenance costs; and
- Allow for the evolution of hardware and software components as technology advances.

To address these objectives, it was accepted that one-of-a-kind hardware or software decisions had to be avoided. Thus from the beginning, there was a strong desire to build the system around a well-known, commercially available, "standard" computer. The obvious choice at the time was the IBM PC/AT standard, which was an Intel 80286 CPU-based microcomputer. Its enormous installed market base meant that hardware add-on components, such as display cards, memory cards, serial I/O cards, and ethernet networking cards, were widely available. The market dominance of Microsoft's DOS standardized the operating system software. These two factors allowed the growth of a huge base of software tools and add-on hardware peripherals, which in turn allowed personnel to build highly specialized systems that could meet almost any functional need from low cost, high volume commercial products. The one problem with this approach was that AUSS had experienced reliability problems with an

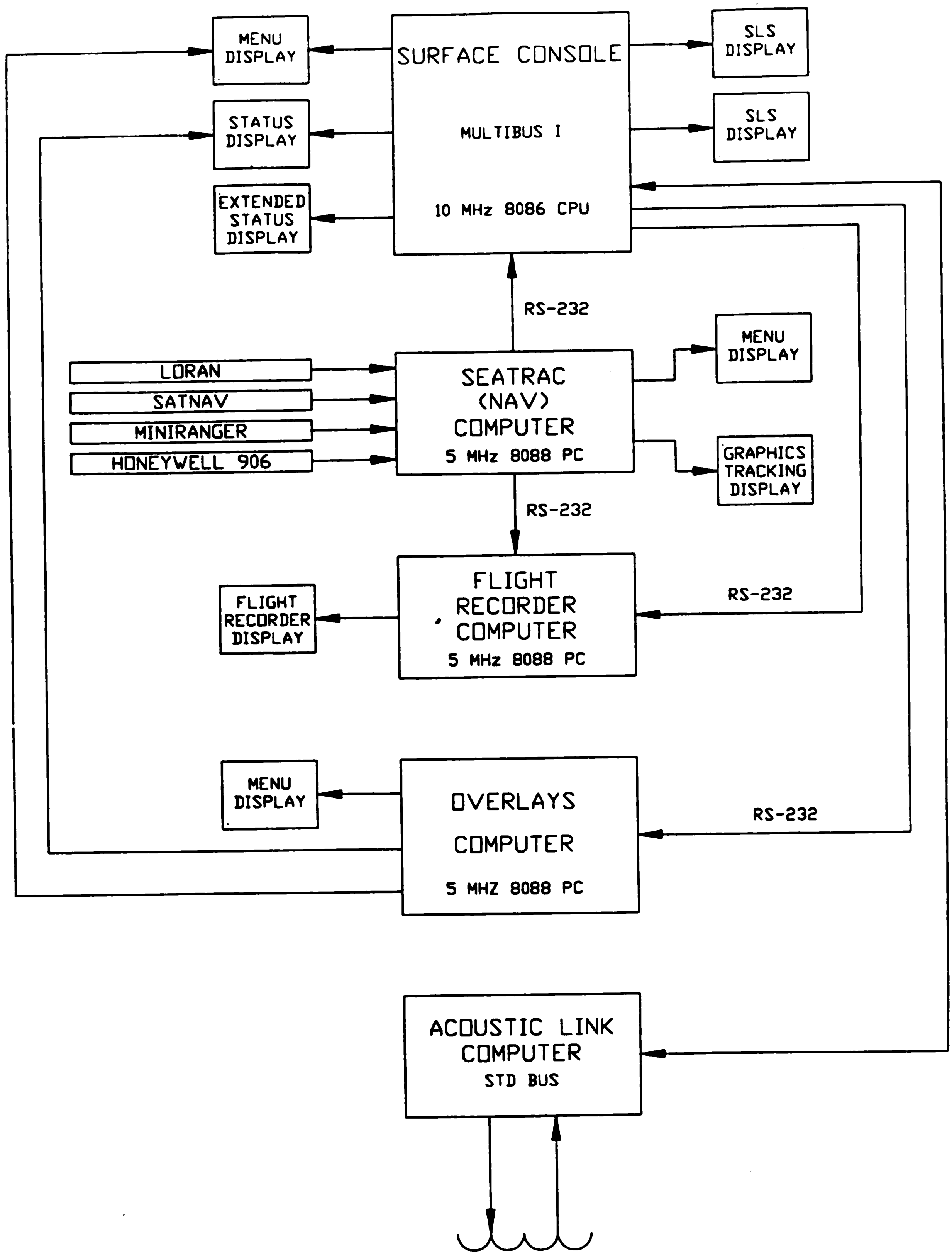


Figure 1. Original AUSS computer system.

original IBM PC because of edge card connections in the system backplane bus. The motherboard designs and the individual cards proved to be very reliable. However, in at-sea conditions, the system proved to be troublesome due to vibration or corrosion problems at the backplane connector. As a result, militarized Multibus I and a newer Multibus II system were also considered.

The current AUSS surface control van design is based on the use of IBM 7552 industrial computers. These units use a passive backplane with 192 lines organized via two 96-pin DIN connectors. The standard 7552 is based on a 10-MHz 80286 CPU design and has been advertised by IBM to be "IBM AT compatible" because it accepts expansion cards. Compatibility is theoretically achieved by inserting AT type cards into an interfacing cradle that essentially maps AT bus signals onto designated 7552 bus lines allocated for this purpose. An extensive analysis examined the pros and cons of using commercial Multibus I, militarized Multibus I, commercial Multibus II, and IBM 7552 based hardware (see NOSC Memo Ser 941/32–87 [Kono, 1987] for a redesign of the AUSS surface control van). This analysis convinced us to use a 7552-based design.

In summary, it was concluded that the 7552 option should improve reliability better than the commercial Multibus I configuration just because it uses better connectors and its system is specifically designed for a harsh industrial environment. Therefore, less NOSC packaging and configuration of cards would be required to create the final system. The system would have a single bus architecture: IBM 7552. The software would be a mix of C and compiled dBASE III for data logging. Sparing would not be excessive and the software development hardware cost would be minimal. Software support would be significantly improved because desktop PCs could be used in the lab and executable code would be disk based rather than Programmable Read Only Memory (PROM) based. This configuration would have the best long-term supportability because of the hardware and software environment on which the code would be based. New cards were designed to support the acoustic link function, and they have worked well. The command computer is interfaced to the old STD-bus-based acoustic link computer via an interface card in the 7552. This interface connects to the acoustic link computer via a connector that will remain the same on the new Multibus II based acoustic link computer. Video cards that support the display requirements are commercial units and are less expensive than the Multibus versions.

In theory, the 7552 PCs were supposed to be "AT compatible." After receiving our first unit, we discovered that the machines were in fact only partially compatible. The bus design turned out to be a hybrid of the AT Industry Standard Architecture (ISA) design and the new PS/2 microchannel (MCA) design. The biggest compatibility problem with the ISA specifications was that some signals were left off the bus. In addition, certain details, such as the address location of the keyboard port, were changed in such a manner that the 7552 was incompatible with some operating system software. These incompatibilities are covered in detail in appendix A, which discusses bus modifications. After the initial evaluation of the first 7552 was completed, it was decided to remove the 10-MHz 80286 CPU and memory cards and replace them with third-party

20-MHz 80386 CPU/memory cards. The net result gave us ISA bus compatibility and software compatibility with our desktop machines, plus a significantly higher processing speed. The 7552 enclosure, passive backplane, power supply, and card shrouds and adapters have been kept and have proven to be reliable and trouble free.

Architecturally, the functions to be provided by the surface control van computers were split up among several machines, rather than concentrated in a single monolithic machine. Splitting up the functions provided some modularity of the software design and allowed for accommodating increased processing demands as requirements changed. Another machine could be added in the worst case to accommodate more processing. This was a 180-degree departure from the original Multibus I design, which depended on RMX, a realtime multitasking operating system.

The layout of the planned surface computer architecture is depicted in figure 2. As figure 2 shows, these five 7552's are labeled according to their functionality. The main vehicle control console is named the command (CMD) machine. The 7552 responsible for assembling image data and displaying them on various screens is called the images (IMG) machine. The AUSS integrated navigation system 7552 is referred to as the AINS machine. The surface data logger 7552 becomes the LOG, and the file server is referred to as FS. Besides the 7552-based machines, another machine, called the data docker, is planned, which is basically a single board, 80286-based machine from Ampro Computers, Inc., intended to receive the vehicle on-board data-logging disk after a dive. The vehicle is outfitted with a similar Ampro system (running DOS) and is programmed to log data sent to it by the vehicle sensor computer during a dive. This raw data can be uploaded to the surface computer network by physically moving the SCSI disk drive to the data docker computer, which is networked into the control van system. The drive is packaged in a carrier that makes the drive act like a plug-in cartridge. You pull it out of one system and plug it into the other. Once the data is offloaded via a network transfer, the disk can be erased and plugged back into the vehicle. The current data capacity of the disk is about 100 Mbytes, but it can be expanded to a 300 to 500 Mbyte capacity by simply replacing the drive with a current technology model.

The surface control van currently uses four 7552/386 PCs for the CMD, IMG, NAV, and AINS computers. The navigation/SEATRAC 7552, referred to as the NAV machine, runs the SEATRAC navigation software. Two 7552/486 PCs are used for the LOG and dedicated Novell FS machines. The 7552/386 PCs are currently based on Texas Microsystems 20- MHz 80386 CPUs. The LOG and FS machines have been upgraded with 33-MHz Diversified Technology 80486 CPUs because the processing load has been taxing the 20-MHz 386 units. The Novell FS has also been upgraded with a Digital Audio Tape (DAT) drive for backing up the FS to 4-mm tape and quickly offloading data files from the server disk between dives. This is a convenient feature since our current FS capacity is limited to approximately 200 Mbytes.

Figure 3 illustrates our current architecture for the control van computers as of July 1992. Figure 4 is an operator's view of the control console as currently configured in the van. The main differences are a temporary laptop computer for displaying

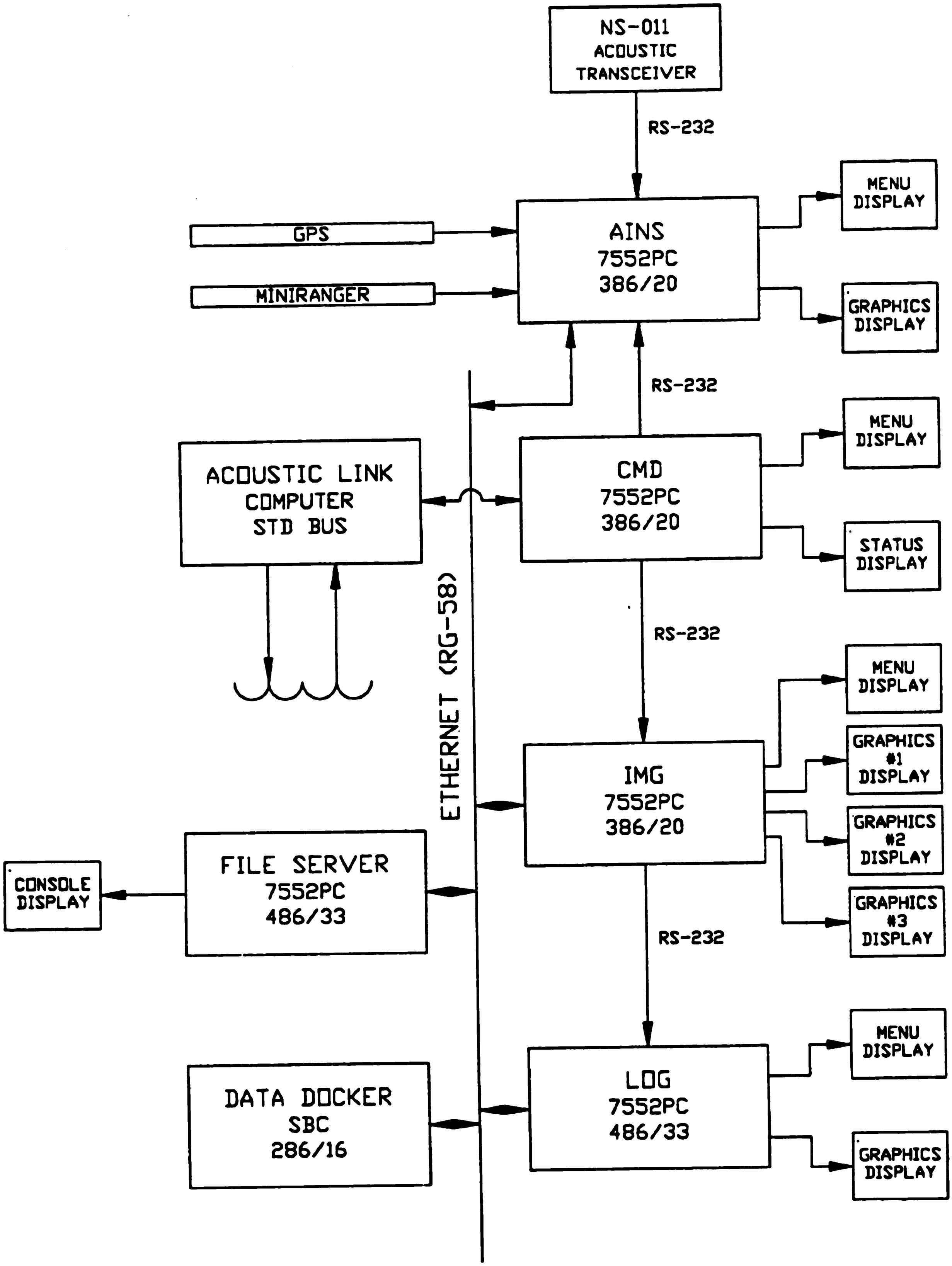


Figure 2. Planned AUSS surface computer system.

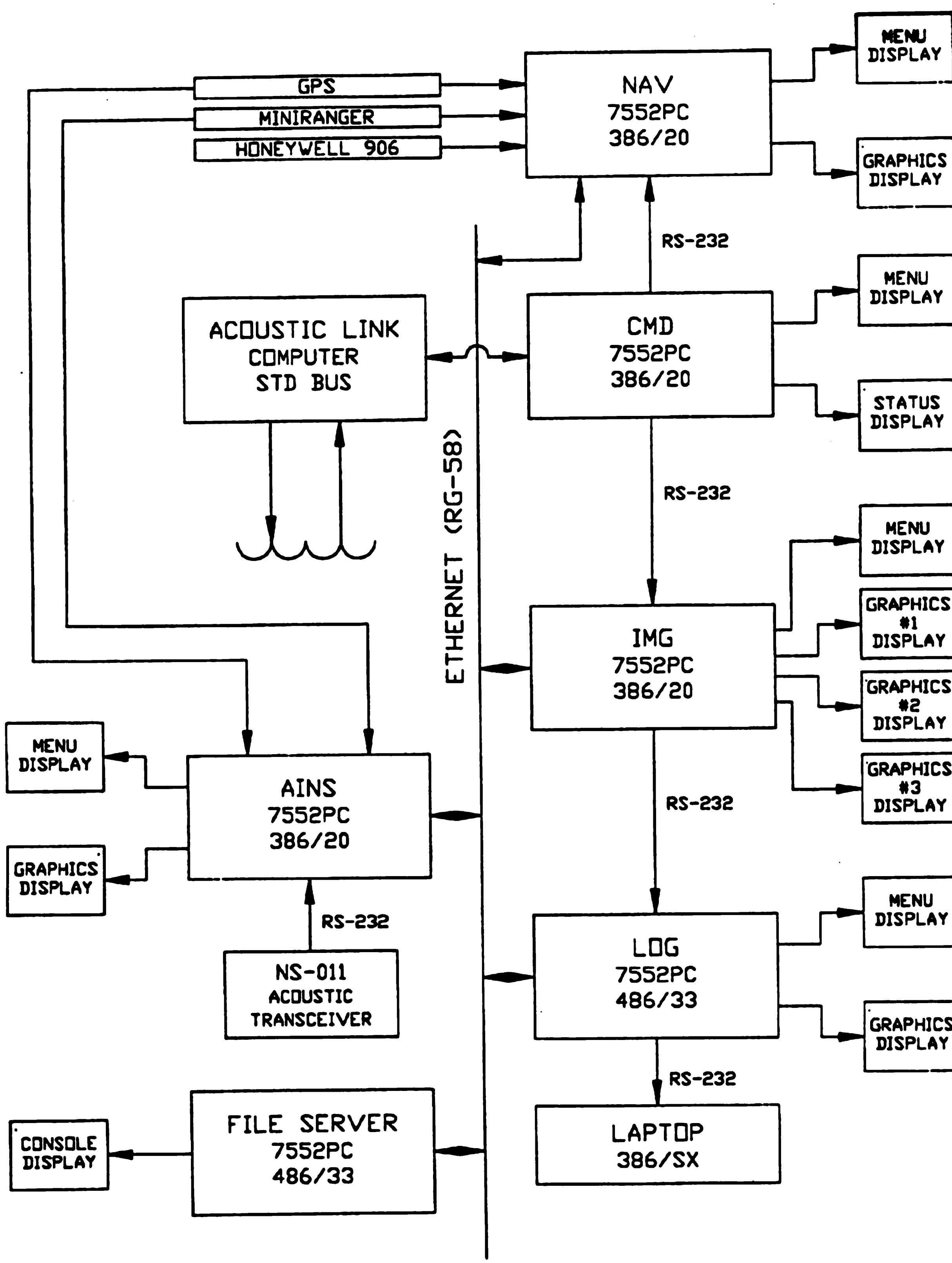


Figure 3. Current AUSS surface computer system.

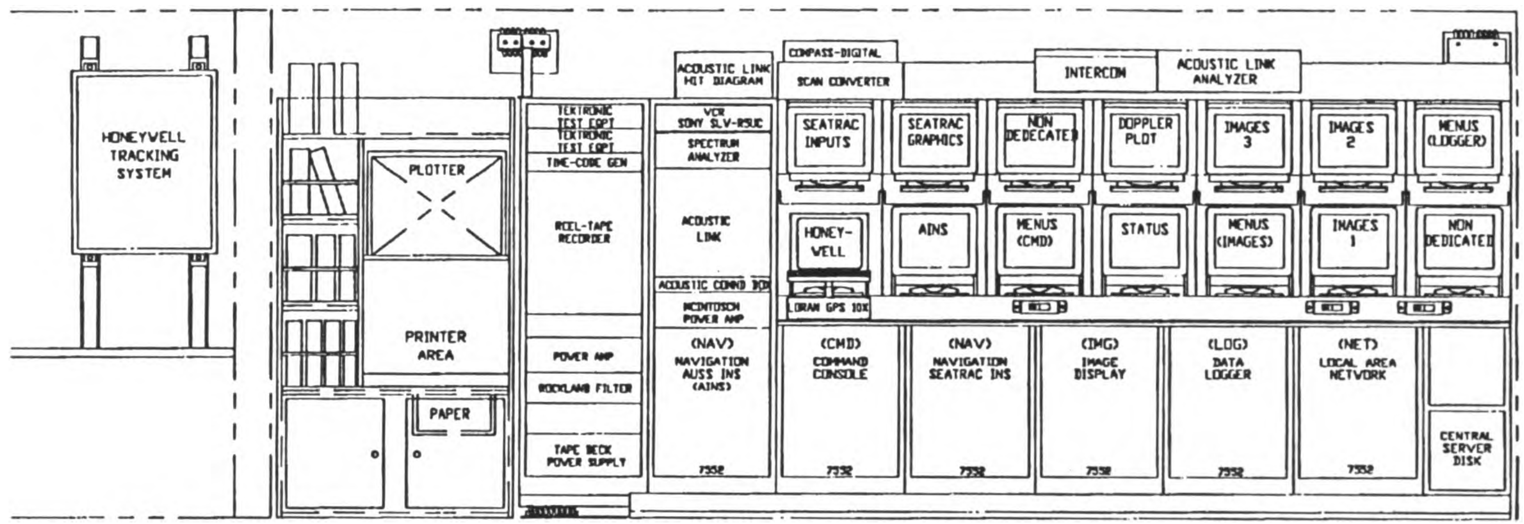


Figure 4. Current AUSS control console.

Doppler-generated plots of vehicle track, no computer to receive the vehicle data-logging disk, and an excessive number of keyboards and monitors. The laptop machine was spliced into the system to fill operational needs that evolved as AUSS was used more and more at sea. Before using a Doppler plot, the only way to provide an updated vehicle position was via either short or long baseline acoustic tracking fixes. As the at-sea operations moved from system shakeouts to actual search-like maneuvers, it became apparent that the operator had to have more frequent and consistent position fixes on the vehicle to issue follow-on commands. This was especially evident when he or she tried to execute contact and evaluation or target closure maneuvers. With a supervisory-controlled submersible, driving becomes a-point-and-shoot-and-see-what-you-get exercise. With the acoustic tracking system, a typical position update could be achieved at best in 10 to 15 seconds, but these fixes could display the vehicle as moving erratically due to bad acoustic conditions. With the Doppler tracking system, updates were limited by the rate status packets were commanded to be sent by the vehicle operator. A 15-second update rate was typically used. However, unlike the acoustic fixes, each new position was highly consistent with each previous and each successive fix. In other words, the Doppler position data eliminated "spikes" in the vehicle position display, providing the vehicle operator with the accurate relative motion of the vehicle. This feature was valuable to the operator during target closure and contact evaluation maneuvers. The Doppler data was then plotted on the laptop screen and relayed to a regular CRT in front of the vehicle operator. The laptop was used for expediency. It was easier and faster to generate a program to implement this display on a dedicated DOS machine using a VGA display card than to implement it on the LOG machine. The LOG machine was a more difficult platform to implement this plotting function from because its graphics display card was a Number Nine Pepper SGT Plus card. Ironically, the Pepper card is a more powerful display card, but the programming library does not lend itself to certain graphic functions needed by the Doppler plotting task, while the graphics library for the VGA card has the desired functions. The Number Nine library lacks functions to draw pie-shaped sectors, rectangles, clipped window areas, and scale screen images. The sector and rectangle shapes are used to depict forward-looking sonar (FLS) coverages and the rectangles to depict charge-coupled device (CCD) video coverages. The scaling and clipped window functions are needed to size the display area and exclude plots outside of the desired area. We plan that this same Doppler plot task will become a DESQview task running on the LOG machine concurrently with the data logging tasks. When this happens, the laptop will be removed from the system. The LOG machine can currently capture the X,Y Doppler data from the status packets and simultaneously display vehicle position while performing its data logging function, but the laptop's swath coverages and target marking capabilities are not yet implemented.

Using the laptop computer to provide an interim solution to an operational need illustrates how readily the current surface computer architecture accommodates changing requirements. It also illustrates how the surface computer software components can be developed as a quick response by drawing on the many options available for the MS-DOS/PC platform. Since the architecture has moved from a monolithic to a more distributed nature, tasks are more decoupled from each other and processes are more modular. More will be said on this topic in the summary section of this report.

OPERATING SYSTEM SOFTWARE

The six 7552-based PCs are either 20-MHz 386DX or 33-MHz 486DX based DOS machines. The 7552 is an industrialized version of the generic "PC" designed by IBM to run in harsh industrial environments. The actual box diverges from the conventional PC design, however, because it is a 19-inch rack mount package that uses a passive backplane. Active components plug into the backplane via two or, optionally, three 96-pin DIN connectors. To create a system via commercially available components, individual card housings are supplied that hold a cradle or "feature" adapter into which a standard AT form factor card is inserted. The cradle maps the AT card edge (bus) connections to the 7552 DIN connectors. Details of this mapping are in appendix A and the *IBM 7552 Industrial Computer Technical Reference 1.0 System Level* (International Business Machines Corporation, 1987). The CPU/memory functions are provided by commercially available cards designed for AT-style passive backplanes. Once inserted into the 7552 backplane via the cradle/housing, make/break operations are done via the DIN connectors. Additional cards, such as I/O or video cards, are installed in a similar manner until a complete MS-DOS machine is configured. The DIN connections have proven to work reliably as the primary make/break point.

MS-DOS, DESQview 386, and Novell Netware v3.11 are used for the operating system software for these systems. MS-DOS provides the basic services for using the disk/file resources of the computer. DESQview provides the ability to simultaneously multitask several programs on a single machine. Novell Netware lets us create a dedicated file server for disk storage that is accessible by all surface computers—basically allowing us to have a large (200 Mbyte or more), cached file storage. In addition, with appropriate programming, files are accessible at the same time by more than one computer.

All six 7552-based PCs in the surface control van use MS-DOS as an operating system in some way. The FS uses MS-DOS v5.00 just on power-up to execute the Novell Netware bootup programs. Once Netware is running, DOS is not used by the FS: Netware v3.11 becomes the operating system software for the FS at this point. The bootup configuration files are in appendix N. The setup for the other PCs is detailed below.

The CMD 7552 uses DOS 3.3 instead of the current DOS 5.0 because its software is burned in Read Only Memory (ROM) as a ROMDISK and it has never needed the features offered by DOS 5.0. Likewise, the CMD machine's application program is written to use DESQview's multitasking capabilities, but it uses an out-of-date version, DESQview 2.25, because it is sufficient. The 7552 for CMD has 2 Mbyte of memory, but it uses no memory management software to make all this memory available under DESQview because the CMD programs do not need the memory. A copy of this machine's bootup configuration files is in appendix I. All necessary code for operation is contained on the ROMDISK, which emulates a 1.2-Mbyte floppy disk.

The IMG 7552 uses DOS 3.3 and DESQview 2.25 like the CMD machine, because it does not need the features of the current DOS and DESQview. Unlike the CMD machine, however, the IMG box boots off a boot file stored on the network FS rather than off a disk drive. To do this, the ethernet card in the IMG machine has a boot ROM installed that connects to the FS and loads a boot file into memory upon power-up, creating a virtual floppy drive A:. This virtual drive contains the programs necessary to connect and login to the server and start-up the IMG programs running under DESQview. A copy of this machine's bootup configuration files are in appendix J. Like the CMD machine, the IMG PC has 2 Mbyte of memory.

The LOG 7552 runs with DOS 5.0, DESQview 2.3, and QEMM 6.02. QEMM is the memory manager software that allows DESQview to take advantage of the entire 8 Mbyte of memory in the machine. This machine has a 1.44-Mbyte floppy and an 80-Mbyte hard disk installed. Booting is done off the hard drive. Copies of the boot configuration files are in appendix M.

The NAV/SEATRAC 7552 runs with DOS 5.0. DESQview is not used since SEATRAC is a program designed to be run directly from DOS. This machine has a 1.44-Mbyte floppy installed to allow a convenient method of loading files into the network server and to provide a backup boot device. The primary boot method for this machine is via a boot file stored on the network FS, as is done for the IMG machine. Copies of the boot configuration files are in appendix K.

The AINS 7552 runs with DOS 5.0. This machine has a 1.44-Mbyte floppy installed and is the only boot device. Copies of the boot configuration files are in appendix L.

HARDWARE ARCHITECTURE

The AUSS surface computers are based on IBM's 7552 industrialized PC. When the decision was made to redesign AUSS, an analysis was done to evaluate various "platform/bus" options. Among the options were the

- Commercial Multibus I based as used in the original AUSS;
- Multibus II based system;
- Militarized Multibus I based system; and
- Ruggedized IBM AT (PC) based system (Model 7552).

The pros and cons of the various alternatives are detailed in NOSC Memo Ser 941/32–87 (Kono, 1987). The decision was made to use the IBM 7552 design. The hardware implemented in each of the 7552 "boxes" used in the surface control van is discussed in this section.

BACKPLANE/BUS

The IBM 7552 is a 19-inch rack mount, passive backplane, PC–AT equivalent system ruggedized for industrial applications. When originally procured, it was offered as a 10-MHz 80286 AT system that was supposed to be compatible with standard IBM AT expansion cards and software. The passive backplane bus presented two 96-pin DIN connectors into which the AT bus signals were physically mapped. The *IBM 7552 Industrial Computer Technical Reference 1.0 System Level* (International Business Machines Corporation, 1987) is the technical manual for the IBM 7552; it contains the signal definitions of the bus and the mapping of 7552 to IBM AT bus signals.

Shortly after the first 7552 was delivered, however, it was learned that the 7552 bus was not truly compatible with the IBM AT standard. Only some AT expansion cards worked with the bus and cradle adapters available to interface the AT cards to the new bus. Regarding hardware, we learned that the 7552 bus was a hybrid of the AT standard and the new microchannel bus. Regarding software, we learned that the CPU card design's keyboard controller design had been changed, making the hardware incompatible with the DESQview software that was to provide our multitasking capability. The bus incompatibility problem was solved by soldering connections from unmapped AT bus signals to 7552 bus pins that were not used by the AT mapping within the cradle adapter. For more details on the cradle adapter see the *IBM 7552 Industrial Computer Technical Reference 1.0 System Level* (International Business Machines Corporation, 1987). These modifications are detailed in appendix A.

The software incompatibility with DESQview 386 was rectified by removing the IBM 80286 CPU/memory cards and replacing them with third party 80386 CPU/memory cards designed for passive backplane AT systems. By doing this, we picked up compatibility, speed, and multiple sourcing. The actual cards used in each 7552 system are detailed in the sections below.

ACOUSTIC LINK COMPUTER

The computer that processes the received uplink acoustic signals or generates the downlink acoustic signals is actually a STD-bus-based, Intel 8085 CPU based unit developed for the original AUSS vehicle. This computer converts uplink acoustic data to digital data and passes it to the CMD computer. Similarly, downlink digital data originating in the CMD machine are sent to the acoustic link computer and converted to acoustic signals that are then broadcast through the seawater. The STD bus acoustic link computer was to be replaced with a Multibus II based 80386 CPU system similar to the one used in the vehicle. The difference between the surface and vehicle units was to be firmware changes. The hardware is currently ready but the software changes are not yet in place.

NAVIGATION (NAV)/SEATRAC COMPUTER

The NAV computer is one of six surface computers built around an IBM 7552 chassis as described above. The CPU is a 20-MHz 80386 with no memory caching. Main memory consists of 2 Mbyte of 32-bit memory located directly on the CPU board. This CPU/memory card is model B386 from Texas Microsystems. Up to 8 Mbyte of main memory can be installed on the card, and an additional 8 Mbyte can be added via a daughter card. A keyboard connector and a special cable to interface this connector to a standard AT-style keyboard cable are provided.

Video output in this system is handled via two different displays. The menu display for the program is provided via a generic EGA card. Graphics display of position and track type data is routed to a Control Systems ARTIST II display card. The interface to the navigation peripheral devices is done via RS-232 serial ports provided by a DigiCHANNEL COM/Xi 8 port serial I/O card. A Western Digital WD8003EBT ethernet card allows for connection to a Novell network. The primary bootup method of the computer is via a boot ROM located on the network adapter card. A secondary boot method is provided by a 3-and-1/2-inch 1.44-Mbyte floppy drive. This disk drive is interfaced to the system via a Western Digital WD1003-WA2 MFM floppy+hard disk interface card. The disk drive controller is scheduled to be replaced with an IDE disk + I/O card.

AUSS INTEGRATED NAVIGATION SYSTEM (AINS) COMPUTER

A second navigation computer system, whose objective is to eventually replace the SEATRAC system, is installed in the van. The heart of this system is the in-house developed software. A navigation system based on NRaD software could more practically and rapidly implement updates and modifications since NRaD, and not a contractor, would control the software. Features that are valuable to AUSS could be implemented in a timely and cost-effective manner.

Like the NAV/SEATRAC system, the CPU is a 20-MHz 80386 with no memory caching. Main memory consists of 2 Mbyte of 32-bit memory located directly on the CPU board. This CPU/memory card is model B386 from Texas Microsystems.

Video is provided by two different display cards. Menus are handled by an ATI Wonder 800+EGA card and plots are directed to a Number Nine Pepper SGT Plus display card. A Western Digital WD8003E ethernet card is installed, giving this machine access to the Novell network. A custom interface card interfaces to the control van gyro and takes synchro signals in, converting them to digital inputs for the software to process. Serial I/O consisting of two RS-232 ports is installed via a combination IDE disk controller and I/O card in order to interface the GPS-LORAN box and take acoustic tracking data from the NS-11. Bootup of this machine is via a 3-and-1/2-inch 1.44-Mbyte floppy drive interfaced to the IDE floppy+hard disk controller card.

COMMAND (CMD) COMPUTER

The CMD computer is the main vehicle operator interface machine. Commands are issued from this machine and routed down to the vehicle via the acoustic link. In a similar manner, vehicle uplink data come to this machine and are processed and displayed or relayed to other machines in the control van. The CMD machine provides the interface to the acoustic link computer via a custom interface card that establishes a direct digital port connection between the two computers.

The base machine is again centered around a Texas Microsystems model B386 20-MHz CPU/memory card. Memory in the system is 2 Mbyte. Two video cards are used: a standard IBM compatible monochrome card for the menus and a Number Nine Pepper SGT Plus card configured as a CGA display card for the status screens. A SGT Plus card is used for the status display because its video output is an analog RGB signal compatible with a standard VGA signal. Various displays from surface computers are repeatered to nondedicated monitors or a scan converter in order to be recorded on a S-VHS recorder. An acoustic link interface card, designed in-house and fabricated to the IBM 7552 form factor, provides a connection into the acoustic link computer. This interface allows uplink data packets to reach the surface computers by using the CMD machine as a gateway. Conversely, downlink packets are sent from the CMD machine to the vehicle via this interface. A generic dual serial I/O card is installed to allow a serial link between the CMD machine and the IMG machine. Uplink sensor image data, as well as other ASCII uplink and downlink data, are relayed from the CMD computer to the IMG machine via a RS-232 serial link. There is no ethernet connection to the network on this machine. The second serial port is used to connect the CMD machine to the NAV machine. The CMD uses this connection to pass Doppler X,Y position data to the navigation software for plotting vehicle position. Bootup is accomplished via an Industrial Computer Source ROMDISK model PCE/2 card. This card provides a self-contained disk drive emulator that stores DOS disk files into PROM. The card with its software allows the operating system of the computer to read these files and actually boot the computer up. Operationally, whenever changes are made to the CMD program, the ROMDISK card must be removed from the 7552 and brought up to the lab and PROMs erased and then reprogrammed with the new files. The ROMDISK card has proven to be very reliable at sea, as we had hoped. However, the reprogramming cycle for even the smallest changes has been inefficient during the development of the project. Therefore, for the duration of development, the ROMDISK should be replaced with a floppy/hard disk combination. The ROMDISK would work well for the delivery configuration, where it would offer a slightly faster bootup time and probably higher reliability.

IMAGES (IMG) COMPUTER

The IMG computer is the main sensor display machine. Sensor data, side-looking sonar (SLS), FLS, and CCD TV are routed to this machine via the serial link from the CMD computer. The IMG computer then processes the data, assembling the packetized data into bit maps for up to three display cards. For example, a port SLS, starboard SLS, and a FLS display could simultaneously be up and updated as the data packets arrive from the vehicle. Targets can be marked on each display or enlarged in scale in a small window for closer inspection. Once the screen is filled, the image is stored to the network FS as a binary image file. A messaging scheme is yet to be implemented that would have the IMG machine send an interprocessor message to the LOG machine, alerting it to the new image file so it could be cataloged in the master database for future recall.

Like the other machines, the heart of this system is a Texas Microsystems model B386 20-MHz CPU/memory card. Memory in the system is 2 Mbyte. Four video cards are used: a standard IBM compatible monochrome card for the menus and three Number Nine Pepper SGT Plus cards for the sensor displays. This machine boots up from the network FS. To provide network connectivity, a Western Digital 8003EBT ethernet card is installed with a boot ROM. The boot ROM invokes the loading of bootup files from the FS into the IMG machine upon power-up. A generic serial I/O card is installed to provide two RS-232 serial ports. COM1 is set up to connect to the CMD machine and COM2 is set up to connect to the LOG machine.

LOGGER (LOG) COMPUTER

The LOG computer is the main computer for managing the storage and retrieval of ASCII and sensor image data. ASCII data is routed from the IMG machine to the LOG via a serial RS-232 link. The ASCII data include uplink vehicle data and downlink command data streams. Uplink vehicle data include status packets and flight recorder dumps. All ASCII data is captured (written) to a disk file that can be viewed by any text editor. This ASCII text file is equivalent to the Crosstalk® generated text files that were created by running Crosstalk®, the commercial program, on a dedicated PC in the original AUSS system. With the LOG software, in addition to the ASCII capture file, status data packets are parsed and used to update a formatted display window in realtime and are also stored away in dBase III compatible file format. The status data and other ASCII data is to be used by the recall portion of the software to re-create vehicle track plots. The LOG software is also set up to receive interprocessor message packets from the IMG machine. These message packets are to be stored in a time-keyed manner so that vehicle position can be tied to sensor image files created by the IMG machine. Track plots are currently available on the graphics display in realtime as the status data packets arrive.

The LOG machine is based on a 33-MHz 486 CPU/memory card from Diversified Technology, model CAT1000. Memory in the system is 8 Mbyte. A Maxtor model 7080A 80-Mbyte IDE interface hard disk and a 1.44-Mbyte 3-and-1/2-inch floppy drive are installed in the system via an IDE+I/O card. In addition to supporting the drives, this card provides two serial ports. The primary boot device is from the hard disk, and the floppy acts as a backup device. The first serial port, COM1, is connected to the IMG machine, and the second serial port, COM2, supports an optional mouse pointing device. Connectivity to the net is handled via a Western Digital WD8003EBT ethernet card. This machine was originally based on a Texas Microsystems B386, and it was set up to boot from the network FS like the IMG machine. The change to the Diversified Technology CAT1000 was necessary because the logging software could not keep up with the continuous capture of 9600 baud serial data (the extreme test scenario) in realtime while performing the other processing tasks mentioned above. In addition, bootup was originally done via the network, but this method was replaced by conventional disk-boot methods to make changes to the system configuration easier during the development cycle. The disks have proven to be trouble free during sea tests. There are two displays on this system. The main console/menu display is handled by an ATI

Basic VGA card. A graphics display is provided by a Number Nine Pepper SGT Plus card equivalent to those used in the IMG computer. The VGA card handles program menus and the Pepper card takes care of sensor image displays and vehicle track plots.

NETWORK FILE SERVER (FS)

The network FS is another 7552 PC set up and running under Novell Netware v3.11. The purpose of this machine is to provide a central file storage location for control van computers. This centralized storage is intended to facilitate easy data sharing among the control van computers and minimize hardware requirements in each machine, i.e., eliminate disk drive requirements. Under Netware, all machines, except the acoustic link computer and CMD computer, bootup and "attach" to the FS. Once attached or logged into the server, virtual disk drives become available to each computer, compliments of the server machine and the network operating system software. Files or data on the server can be made available to every machine, or conversely, access can be denied. The original concept for the surface computer architecture called for the LOG machine to read navigation and image data from the server in order to catalog and retrieve the data. Current LOG software accesses IMG-generated data, but it does not yet access SEATRAC or NRaD navigation software data.

The FS is based on a 33-MHz 486 CPU/memory card from Diversified Technology, model CAT1000. Memory in the system is 8 Mbyte. File storage is provided by a 213-Mbyte Maxtor LXT213S SCSI hard disk. The SCSI drive is interfaced to the computer via an Adaptec 1542B SCSI controller card. The Adaptec controller can control up to two floppy disk drives and seven SCSI devices. The on-board BIOS ROM on the controller provides the capability to create a small DOS partition on the SCSI hard disk, from which boot files to bring up Netware are invoked. Once the server is booted up, control switches from DOS to Netware. To provide a means to offload large amounts of data quickly and conveniently from the server disk between dives, an Archive model 4520NT DAT drive is also attached to the Adaptec controller. Each DAT tape has a 1.3-Gbyte capacity and transfers data at a 6 to 7 Mbyte per minute rate. The Maxtor drive is set up with an SCSI ID of zero and the DAT drive is set up with an SCSI ID of two. An ATI VGA Basic card is installed as the console display for the server. The ethernet interface card is a Western Digital WD8013E 16-bit card.

NETWORK WIRING

As mentioned, all the 7552-based PCs, except the CMD machine, are networked together. The network wiring is a simple daisy-chained RG-58 coax that runs from the AINS machine to the NAV, to the IMG, to the LOG, and finally to the server. This is illustrated in figure B-1. Each machine is connected to the coax via a BNC T-connector. Each end of the RG-58 coax MUST be terminated with a 50-ohm terminating resistor, and there cannot be an open break in the coax daisy chain.

RS-232 Serial Wiring

The CMD machine is a logical starting point for documenting the serial interconnections between machines. A serial connection is made from the CMD machine to the IMG and NAV machines. As explained earlier, the CMD machine passes uplink packets and downlink CMD packets to the IMG machine. The downlink packets are passed just for the purpose of relaying them to the LOG machine for archiving. The IMG machine processes sensor image binary data and loads the display cards in such a manner that the sensor images are displayed on the monitors. Image data packets are not relayed to the LOG machine. Instead, the sensor images, once displayed, are saved as binary files on the FS. At this point, it is planned that the IMG machine will generate an ASCII message packet destined for the LOG machine, alerting it of the creation of a new image file that needs to be cataloged. Status packets containing the vehicle's Doppler X,Y position data are relayed from the acoustic link computer to the NAV computer on the CMD machine's second serial port.

Serial ports are set up to operate at 9600 baud, no parity, and one stop bit. No handshaking protocols are used and thus machines must be able to process serial capture in realtime. Hardware handshaking is avoided because a hang in one machine hangs the other machines. Software handshaking methods are more forgiving but require the use of special characters as handshake signals. Since binary image data may contain these characters, software protocols are also avoided.

Display Switching/Wiring

The original AUSS control van had a custom Intel Multibus I based computer system as its operator command console. Greyscale composite video display monitors were part of this console. The composite video output for the displays made it easy to switch or relay the video signal to any other composite video monitor in the control van. In addition, the composite video signal was directly recordable on a VHS tape recorder. With the redesign of AUSS, it was decided to rethink how the topside computers and displays would be implemented. In short, the decision was made to use standard PC components, computers, and software tools whenever possible. After we conducted numerous experiments and considered what display cards were available, we selected the Number Nine Pepper SGT display card to be our primary image and graphics display card. A major consideration influencing this choice was that multiple display cards would be required in the IMG computer. Three image display cards in this machine currently coexist with a standard monochrome display card for a total of four display cards. The Pepper card puts out an analog RGB signal versus a composite video signal. This analog RGB signal provides a 640×480 pixel resolution with 256 colors or shades of grey. The signal is output on a 15-pin, D-subminiature connector with a pin out that is compatible with a standard VGA display card. Pepper cards are also used in the LOG and CMD machines to standardize the hardware for sparing and video signal routing. Standardizing on a single graphics display card also minimizes the number of software libraries that have to be used in developing surface computer software. To provide switching capability to different monitors, the output signal from each display card of interest is routed to repeater boxes that amplify and condition the original signals before splitting to a one to two or one to eight output as shown in figure B-1. The repeater units are Vopex-2V and Vopex-8V models designed for repeatering standard VGA output signals to multiple monitors. Switch boxes are used to select which display signals from these repeaters are routed to spare display monitors.

SCAN CONVERTER/S-VHS TAPE RECORDER

The video output signal from the display cards can be switched to alternate monitors and an S-VHS recorder. However, since the signal is in VGA RGB format, it must be converted to composite video before being fed into the recorder. The conversion is completely handled in hardware with the YEM model CVS-910 scan converter shown in figure B-1.

SUMMARY

RELIABILITY ISSUES

One of the initial objectives of the surface computer system redesign was to improve the reliability of the overall system. To achieve this objective, the approach below was followed:

- Address the question of edge card connector reliability by selecting an IBM 7552 bus/enclosure that uses DIN connectors for the make/break interface of the cards to the bus for the new systems;
- Use a well-defined and standardized bus architecture, IBM AT bus, along with a well-defined and standardized operating system, Microsoft DOS, to provide a stable operating platform for development and a target environment;
- Make certain all computer cards, except one interface card to the acoustic link computer, are high volume, commercially available products whose designs were subjected to mass market testing;
- Use commercially available, high volume software tools to develop application code;
- Separate application software requirements to multiple machines in such a manner that no single machine becomes overloaded; and
- Use multitasking DESQview software wherever possible within each machine to let us write software as event-driven modules—this process tends to decouple software functions from one another similarly to the effect achieved by separating software functions to separate machines, rather than concentrating them to a single, monolithic machine.

The card edge reliability problem has not been an issue since the conversion was made to the IBM 7552 bus/enclosures. No system failures have been attributable to card connections to my knowledge. Cards in each system must be seated properly prior to operation, but once seated, the DIN connectors perform well. The AT compatible cards are housed in individual enclosures that mate the card to an adapter cradle. The adapter cradle–AT card interface is an edge card connector. However, since these connections are rarely broken, connector problems seem nonexistent.

An IBM PC/AT bus architecture with Microsoft DOS as the operating system (OS) was chosen to be the basis of the surface computer platform. NOSC Memo Ser 941/32–87 (Kono, 1987) analyzed four platform options:

- Commercial grade Multibus I (sanitized AUSS I) using Intel RMX OS;
- Militarized Multibus I using Intel RMX OS;
- Multibus II using Intel RMX OS; and
- IBM 7552 (industrialized PC/AT) using MS-DOS for OS.

The four alternatives were analyzed from the perspective of total delivery cost: the analysis focused on normalizing the comparison by defining costs for all aspects of each choice. Hardware and software procurement costs plus software and hardware development costs were totaled for each option, and the IBM 7552 turned out to be the best quantitative choice. On a more subjective level, the 7552 was preferable because the OS and hardware architecture were well defined and tested in the commercial market. For the other three options, special cards had to be designed and tested in-house, and the system configuration would have been unique to our application. It would have been difficult to isolate operational problems to the application software versus the system software or hardware configurations, because the platform would NOT have had a baseline reference.

Failures at the card level in the various 7552 platforms have been nonexistent. Using cards designed for the well-defined PC platform provided low cost, flexibility, and excellent reliability relative to other platform options that were considered early in the redesign phase of AUSS.

The application software was generated with an approach consistent with hardware selection philosophy. Software tools, languages, libraries, linkers, etc., selected for use were items in wide-scale commercial use. As a result, unexplained software problems in the application code were kept to a minimum. A conscious decision was also made to move away from PLM to a higher level language, e.g., 'C.' Since C is generally accepted as the most popular development language in the marketplace, a multitude of third-party software libraries is generally available. These libraries, like the the multitude of PC adapter cards on the hardware side of the system, are viewed as a resource pool.

As stated earlier, a design goal for the new surface computer architecture was to move away from the tightly coupled, monolithic design of the original AUSS I. That first design relied on a programming language called PLM that produced executable code similar in size and efficiency to that achieved via assembly language programming. PLM was used out of necessity as microprocessor technology at that time considered a 10-MHz Intel 8086 processor state-of-the-art. This early design heavily emphasized making the code as fast as possible: the surface console was asked to do many tasks in realtime. The downside of this approach was that changes to the code almost always disrupted the stability of the previously working program. To address this problem, the surface architecture was set up to load share the processing among various machines—in essence attacking the big problem by subdividing it into smaller problems. The result was the CMD, NAV+AINS, IMG, and LOG computers.

Within each machine, application code for the CMD, IMG, and LOG machines was further subdivided into subtasks running under DESQview. Philosophically, the concept was to continue breaking down the programming problem even further. The net effect was to further modularize the software by creating many small software modules running as independent tasks and communicating with each other when necessary via DESQview intertask mailboxes. In this way, the final application code could be developed and tested incrementally with minimal dependency on other pieces of the system. Wherever external interaction (from the task) was needed, it was relatively easy to simulate it during development. This architecture let us focus solely on the CMD machine code to start up AUSS II. Once it was stable, work progressed on the IMG code. When it was ready, the IMG machine was plugged into the CMD machine via a serial RS-232 link.

SOFTWARE DEVELOPMENT AND MAINTENANCE ISSUES

A second major objective of the surface computer system redesign was to minimize software development and maintenance costs. Software development and maintenance became a visible issue while AUSS was evolving from AUSS I to AUSS II. As mentioned earlier, the analysis in NOSC Memo Ser 941/32-87 (Kono, 1987) to assess which computer platform would be better for AUSS II showed the PC (7552 bus)+MS-DOS system to be better from a cost standpoint. On close review, the factors that significantly raised costs for the Multibus options were software development and debugging. The reasons were simple. First, the target Multibus systems were generally high in procurement costs. Second, you had to have development platforms that cloned the target system for the software developers to test code, and this platform was naturally just as expensive. Third, the software development process using RMX as an OS and PLM as the language and loading the executables into ROM did not lend itself to being efficient. Everything about these Multibus approaches suggested high risk: the developer had to integrate hardware and software to produce the basic system, and the developer was responsible for hardware or software extensions to the system.

Software development and maintenance have been very successful for the surface computer systems using the PC+MS-DOS platform for development and target systems. One problem, mentioned earlier, has been recently noted regarding the Pepper display cards and Doppler plot requirement. In taking advantage of the rich selection pool offered by the PC adapter card market, we selected a video card, the Pepper SGT Plus, as our standard graphics card, because it had many desirable features designed into it and a library of software functions that made using these features in the code relatively easy. On the one hand, this selection has helped, but on the other hand, graphics display code has become dependent on this card and its associated software library. This is an area of the current system design that must be re-evaluated.

Information on the software tools used for development and the resultant AUSS surface software can be found in NRaD TN 1705, "Advanced Unmanned Search System (AUSS) System SW Description: Vol. 1 Vehicle SW/Vol. 2 Surface SW" (see Schwager reference in Bibliography.

ADAPTING TO TECHNOLOGY ADVANCES

The final major objective of the redesign was to provide an architecture that could easily adapt to hardware and software technology advances. This objective has been successfully met. At the hardware level, our bus system is currently based on the IBM PC/AT ISA bus. However, since the backplane is passive and uses a card adapter cradle to map AT bus signals to the DIN connectors on the backplane, it is conceivable that the card sets could be changed to another bus standard at a later time. IBM, in fact, has evolved the model 7552 into a model 800, which is basically a system that uses a 25-MHz 80486 processor running on a bus mastering microchannel bus. I have been told that the passive backplane has been slightly modified for 100 percent microchannel orientation and the cradle adapters have been changed to accommodate microchannel cards commonly found in IBM PS/2 desktop computers.

With our modified 7552 bus, system CPUs have evolved from a 10-MHz 80286, to a 20-MHz 80386, and finally to a 33-MHz 80486. Updating has been as simple as plug and play. No programming changes have been required. Some of our menu video cards have been updated from TTL monochrome to analog VGA in such a manner that these displays could be distributed to multiple monitors and recorded by a VCR. Operating system software has been updated from MS-DOS 3.3 to 5.0. The LOG machine has had its DESQview software upgraded twice. The networking software has also gone through a major revision upgrade.

RECOMMENDATIONS

In the early phases of the AUSS surface computer system redesign, a need for multiple video displays was identified. In response to this need, the Number Nine Pepper SGT display card was eventually selected to provide us with up to four auxiliary display screens on any single platform. At the time this decision was made, it seemed logical to orient the system design to be hardware intensive to minimize the requirements of the software development effort. In making that decision, however, the design process failed to recognize the long-term effect of locking into a proprietary display card and a one-display-per-video-card-mapping philosophy while writing the application programs. For reasons detailed below, this section argues for redefining the basic AUSS surface computer to be an X Window Platform in the long term.

Since each Pepper SGT video card was viewed as a peripheral device in a given machine, the software graphics tasks were written to talk directly to the video card.

Each task could then run under DESQview as a subtask. Even with three to four Pepper cards, a system could provide simultaneous updating of multiple displays. The Doppler plot requirement mentioned earlier gave us the first indication why this might NOT be a good system design decision. Software applications that will use the Pepper SGT as a graphics display media are currently limited by the capabilities of the associated vendor-supplied and -supported software libraries. Furthermore, it was assumed that these cards would be available and supported indefinitely. The fact is that these cards are no longer produced and have been superseded by new models that do not use the original software libraries.

Each display card in the surface computer system generally has a video monitor hooked to it at some point (see appendix B). The exceptions are the acoustic link computer, which is really more of a controller, and the FS, which shares a monitor because it does not generally need an active display. The net result is 13 PC-type monitors currently installed in the control van and typically supporting 640 × 480 pixel resolutions on 14-inch diagonal screens. The number of monitors combined with the requirement to switch display sources to two nondedicated monitors and a VHS recorder has created a horrendous wiring problem behind the monitors. This is displayed in figure B-1. To minimize this problem in the near term, some specialty devices that will allow keyboards and monitors to be shared will be installed as depicted in figure B-2. In addition, the Doppler plot software must be updated for the LOG graphics display (using the Pepper card) in such a manner that the laptop computer can be removed and the monitor that is used returned to the LOG machine's graphics display.

For the long term, the graphics display of the AUSS surface computer systems must be re-evaluated. It appears that the obvious solution is to reduce the number of display monitors. To do this, the design must consider using 19- to 20-inch diagonal monitors coupled with a minimum display resolution 1024 × 768 × 8 bits deep. With a larger physical screen size and a higher pixel resolution, it would be feasible to incorporate multiple scalable windows onto a single monitor, thereby providing the functionality of the multiple card/monitor system currently in place. In the IMG machine, a single monitor would provide four windows: one for menus and three for sensor images. The 13 monitors of the AUSS's current 4-workstation setup—CMD, NAV, IMG, and LOG—would be replaced by 4 larger ones. The CMD, NAV, and LOG machines currently require only two 640 × 480 pixel displays, and therefore, with the larger monitor/display resolutions, they would present surplus display area that auxiliary display windows could be overlaid onto. In effect, this would provide the functionality of the two monitors currently used for display switching. Note that it is assumed that the AINS machine becomes the NAV platform and the SEATRAC software is phased out.

Remember that the objective is twofold: (1) to eliminate the dependency of our display system on a proprietary display card and its software library and (2) reduce the wiring required to support a large number of monitors. Thus, the long term solution means considering changes to the software architecture. AUSS currently uses DOS as the OS for each workstation, and in the CMD, IMG, and LOG machines, this OS is extended via DESQview to provide multitasking support. Tasks are grouped within each platform by function. All resources of each machine, other than file storage, are dedicated to the platform's tasks. To achieve the first objective, the multiple, specific display cards must be replaced by a single virtual display. To achieve the second objective, this virtual display must be able to manage multiple graphics windows on a single screen. Each window then replaces one of the current hard-wired display card/monitor subsystems. In addition to the ability to manage multiple graphics windows, this virtual display mechanism must be able to remap windows related to an application from one physical screen to another (on a different computer) to replace our current switched monitors.

The virtual display environment described above can be provided by the X Window System. The X Window System is an architecture that promotes device and machine independence and provides a means of supplying graphical interfaces locally at a single machine or distributed across a network. In a traditional application requiring graphics output, the program makes a call to a library or system software graphic subroutine. This subroutine in turn causes the desired output to appear on the display screen. Typically, this subroutine is some special function, like draw rectangle with rounded corners. This function or subroutine is in turn written dependent on other graphic subroutines, and eventually some very tight link (low-level interface) to a specific display card is made in this library of graphics routines. If the display card is changed and the new card is incompatible with the low-level interface functions used in the library, your application software is no longer functional unless you rewrite the graphics library to support the new card.

In the X Window System, graphics displayed on the screen are done by a task called the X server. In essence, the X server task is a graphics display engine. To invoke some graphical entity, an application sends messages to the X server, then the X server puts it on the screen. For this process to work, the system must support multitasking because the X server must run concurrently with the application or X client program(s). Note that this lets more than one application use the X server and thus provides multiple local and remote application display outputs to the screen controlled by the X server.

This messaging scheme allows graphical applications to be distributed across a network. The traditional approach is procedure oriented, transferring control of the computer's resources to the subroutine that is jumped to. The X Window approach just sends a message to another application (the X server) running concurrently. Figure 5 contrasts the traditional approach versus the X Windows event-driven approach.

Figure 6 diagrams the X Windows scheme being used over a network as envisioned for a future AUSS. In this diagram, the NAV graphics display is directed to both the NAV X server (and thus its display screen) and the LOG X server. The NAV client might be monitoring ship's position in this example. The IMG machine may have a CCD video image being displayed via the IMG #1 client application, and the operator could choose to output this same display to both the NAV and LOG machines via their X servers. This architecture eliminates the excessive display monitor problem and the wiring required for redirecting displays to nondedicated monitors. The computer architecture for the surface control van simplifies from that depicted in figure B-2 to figure B-3.

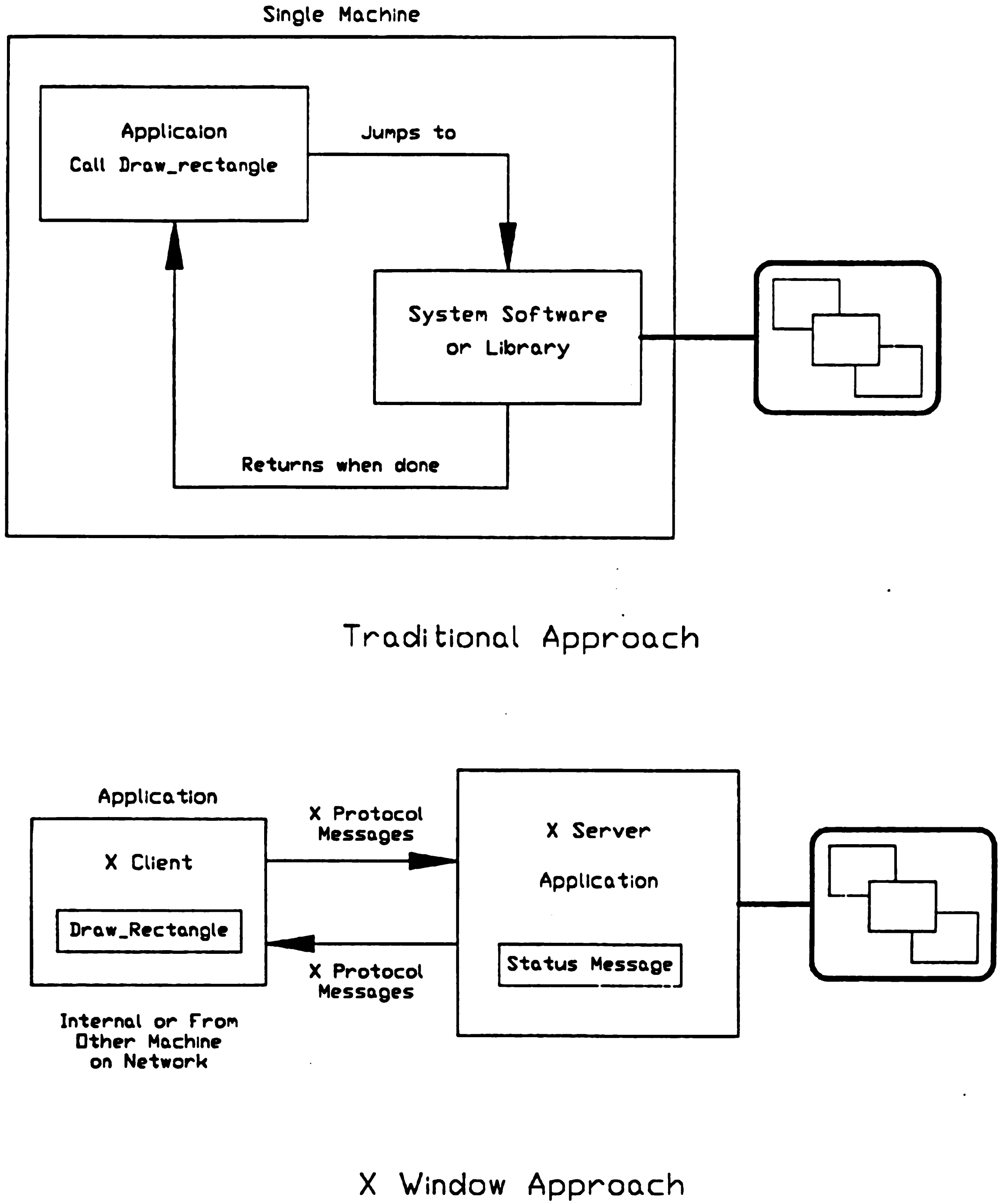


Figure 5. Traditional versus X Windows approach.

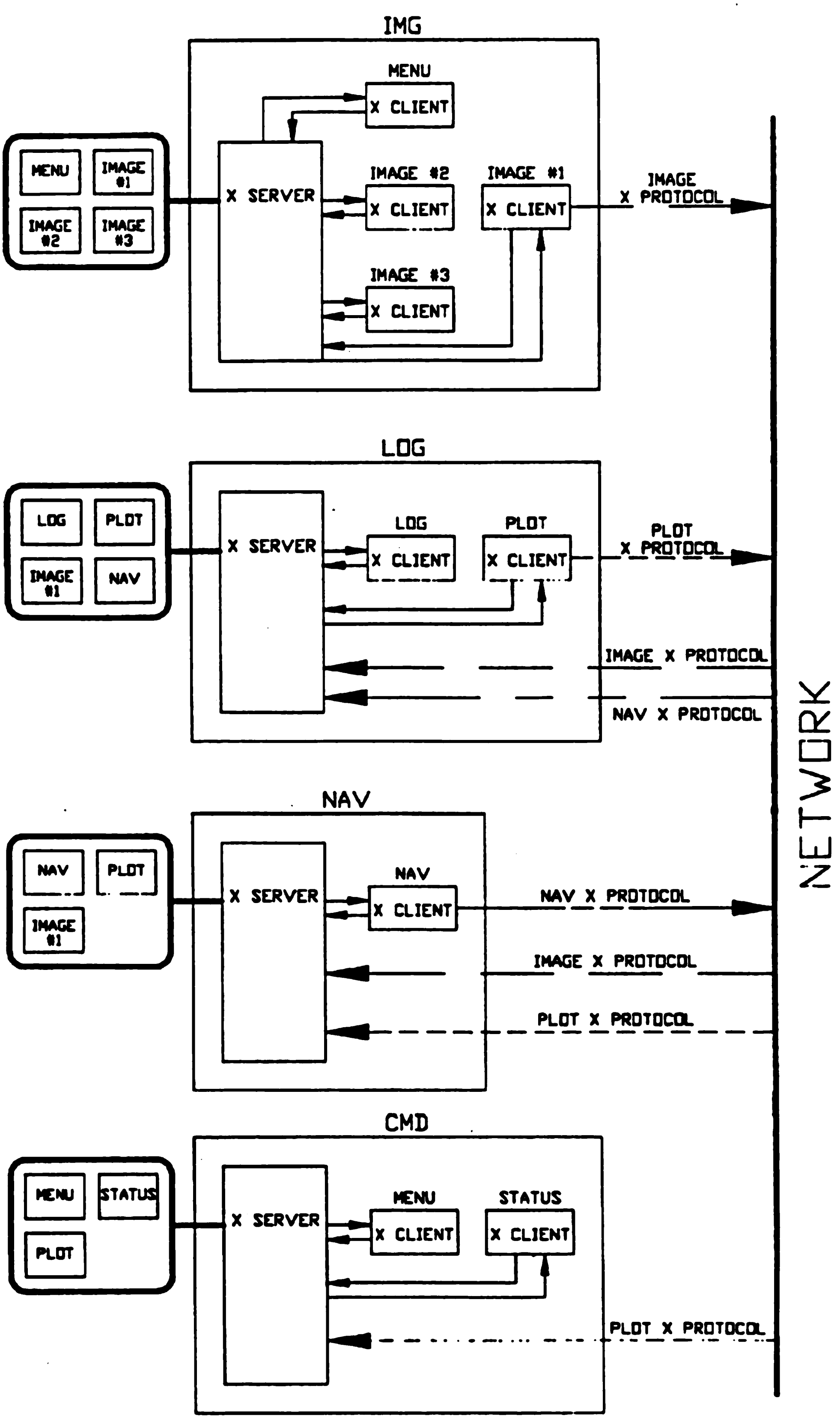


Figure 6. X Windows concept for AUSS.

Within the X server code, there must obviously be some low-level interface to display cards that is supported by the server just like the setup in the conventional subroutine library code. However, a difference exists because whenever a display card is added, low-level functions necessary to support the entire X Window System are well defined and thus easily addressed. Updating the X server code is the responsibility of the developer of the X Window Server software for a particular platform, and it is not the responsibility of the application developer or card manufacturer. The application developer can thus take advantage of technological advances for virtually no cost. He or she just installs the updated X server code and a newer display card to gain the new benefits. The application developer should note that the software becomes portable to any X Window Platform. The platform could be a SUN Sparc workstation running UNIX, a DEC Microvax running VMS, or a PC running DESQview/X on top of DOS.

To convert to an X Window System architecture, the procedure is as follows:

- Stay with DOS-based 7552 PCs as the foundation of the X Window Platform by upgrading DESQview to DESQview/X;
- Set up an X Window Platform in the laboratory for software development;
- Rewrite graphics display tasks used in the IMG, LOG, and NAV to use X Windows;
- Upgrade the display hardware to large, high-resolution, single monitor displays when the rewrite is completed; and
- Rewire the van for X-Window-System-based architecture.

By staying with the 7552 PCs, AUSS can continue running all existing software until the transition is complete. Furthermore, changes required to support the new software are minimal. Existing development platforms remain essentially unchanged. Finally, a hardware architecture that has proven to be extremely adaptable and capable of evolving with technological advances is used. This decision does not prevent AUSS from moving to a more powerful platform or from using a full UNIX OS instead of MS-DOS at a later date. The point is that the investment made to develop the application code would be automatically moved with minimal additional cost.

By staying with the 7552 PCs, AUSS can continue running all existing software until the transition is complete. Furthermore, changes required to support the new software are minimal. Existing development platforms remain essentially unchanged. Finally, a hardware architecture that has proven to be extremely adaptable and capable of evolving with technological advances is used. This decision does not prevent AUSS from moving to a more powerful platform or from using a full UNIX OS instead of MS-DOS at a later date. The point is that the investment made to develop the application code would be automatically moved with minimal additional cost.

The display card used to provide the multiple windows that are mapped to a 1024 × 768 × 8 bit or higher resolution must conform to some widely accepted commercial standard. Ideally, this single replacement display card should be available from multiple sources. The display standard should be a part of the platform standard. In our current configuration, our platform is based on a PC ISA bus architecture design that uses MDA, CGA, EGA, and VGA display card standards augmented by two third-party proprietary design cards: the Pepper SGT and the Artist II. The new platform concept would standardize to a single display card that uses a single display standard. If a display standard had to be currently selected, a Super VGA or a 8514 display would be used. Linking the display card standard to the platform would support porting of the application software to the newer standards as hardware or software technology advanced.

To set up an X Window Platform for software development, the current platforms must be upgraded to an OS supporting X Windows, its development tools, and a supported high-resolution display card.

Rewriting the graphics tasks is clearly the most difficult step. It is a step, however, that will have to be done whether the move is made to X Windows or just another display card. It is currently believed that AINS will completely replace the SEATRAC navigation software someday. An area where the AINS software is currently lacking is the user interface. Thus, it would be logical to now define the specification for the interface to be X Windows based. For the IMG and LOG programs, the main graphics tasks to be re-done handle loading sensor image data from file input or uplink packets. Menu screens on the surface computers are DOS-text-based applications and can thus be run as is since DV/X can translate DOS text screen writes on-the-fly to X protocol messages to an X server. This becomes an interim benefit of staying with the existing 7552 DOS platforms rather than switching to a UNIX-based platform.

To upgrade the surface computers once the graphics tasks are rewritten, IMG, CMD, and NAV machines must have their CPU/memory cards replaced by 486-33 MHz CPUs with 8 Mbyte of memory. The LOG machine is already upgraded. In addition, the multiple display cards need to be removed and replaced by a single high-resolution display card. In the IMG machine, four video cards would be replaced by a single card. In the other machines, two display cards would be replaced by a single card. Instead of having to spare four different types of video cards, only a single card design would be required. The AINS 7552 would be similarly upgraded, but instead of being used as a navigation computer, its function would be changed to simply being an output device to the scan converter/VHS recorder. Any display window from any application could be invoked on this machine's display for purposes of feeding the scan converter. The wiring of the computers would then simplify to that depicted in figure B-3. When the wiring diagram in figure B-3 is compared with the wiring diagram in figure B-1, it is obvious that a dramatic simplification of the wiring becomes possible. Multiple monitors on each machine are eliminated because the distributed processing and display capability plus the window manager capability built into the X Window System allow redirection and multiple display windows at each platform's monitor. There can be a one-to-one or a one-to-many relationship between an X client (application program) and windows on a display monitor. The final recommendation for cleaning up the architecture of the surface computers is to get rid of the RS-232 serial connections linking the CMD to the NAV, the CMD to the IMG, and the IMG to the LOG. The X Window System has mechanisms built into it that pass data from one X client process to another, even if the processes live on different platforms attached to the net.

REFERENCES

International Business Machines Corporation. 1987. *IBM 7552 Industrial Computer Technical Reference 1.0 System Level.* Revision 1.1. International Business Machines Corporation, Boca Raton, FL.

Kono, M. 1987. "Control Van Microcomputer Alternatives Analysis," NOSC Memo Ser 941/32-87 (May). Naval Ocean Systems Center, San Diego, CA.

BIBLIOGRAPHY

Acoustic Systems, Inc. 1992. "Definition of the Advanced Unmanned Search System (AUSS) Sonar Characteristics." NRaD TN 1704 (Sep). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.*

Bryant, S. B. 1979. "Advanced Unmanned Search System (AUSS) Performance Analysis." NOSC TR 437 (Jul). Naval Ocean Systems Center, San Diego, CA.

Cooke, M. W. 1992. "Advanced Unmanned Search System (AUSS)." NRaD TD 2348 (Dec). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Endicott, D. L. Jr., and G. R. Kuhl. 1992. "Fast Area Search System (FASS): Feasibility Study Appendices." NRaD TN 1703 (Sep). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.*

Endicott, D. L. Jr., and G. R. Kuhl. 1992. "The Fast Area Search System (FASS): A Feasibility Study." NRaD TR 1526 (Sep). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Grace, D. R. 1992. "Brownian Reber Search Theory for the Advanced Unmanned Search System." NRaD TR 1534 (Oct). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Gunderson, C. R. 1978. "Advanced Unmanned Search System (AUSS), Preliminary Search Systems Analysis." NOSC TR 375 (Dec). Naval Ocean Systems Center, San Diego, CA.

Held, J. L. 1992. "Automatic Hovering Algorithms for the Advanced Unmanned Search System." NRaD TR 1535 (Sep). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Held, J. L. and H. B. McCracken. 1993. "Automatic Transit Algorithms for the Advanced Unmanned Search System (AUSS)." NRaD TR 1536 (Jan). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Jones, H. V. 1992. "Advanced Unmanned Search System (AUSS) Description." NRaD TR 1528 (Nov). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

* NRaD Technical Notes (TNs) are working documents and do not represent an official policy statement of the Naval Command, Control and Ocean Surveillance Center (NCCOSC), RDT&E Division (NRaD). For further information, contact the author(s).

Keil, T. J. 1992. "Advanced Unmanned Search System (AUSS) Deep Ocean Floor Search Performance Computer Model: Executive Summary." NRaD TN 1702 (Sep). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.*

Kono, M. E. 1992. "Surface Computer System Architecture for the Advanced Unmanned Search System (AUSS)." NRaD TR 1538 (Dec). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Mackelburg, G. R., S. J. Watson, and W. D. Bryan. 1992. "Advanced Unmanned Search System (AUSS) Acoustic Communication Link Development." NRaD TR 1531 (Nov). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

McCracken, H. B. 1992. "Advanced Unmanned Search System (AUSS) Supervisory Command, Control and Navigation." NRaD TR 1533 (Nov). Naval Command, Control, and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Osborne, P. D., and C. C. Geurin. 1992. "Advanced Unmanned Search System (AUSS) Surface Navigation, Underwater Tracking, and Transponder Network Calibration." NRaD TR 1532 (Oct). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Rasmussen, M. E. 1992. "Advanced Unmanned Search System (AUSS) Battery Monitor/Charging Systems." NRaD TR 1539 (Sep). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Schwager, M., and J. Stangle (SAIC). 1992. "Advanced Unmanned Search System (AUSS) Software Description: Vol I Surface SW/Vol II Vehicle SW." NRaD TN 1705 (Dec). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.*

SEACO, Inc. 1992. "Development of the Acoustic Telemetry System." NRaD TD 2336 (Sep). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Stachiw J. D. 1984. "Graphite-Reinforced Plastic Pressure Hull for the Advanced Unmanned Search System (AUSS) (U)." NOSC TR 999 (Oct). Naval Ocean Systems Center, San Diego, CA.

Stachiw J. D. 1986. "Graphite-Fiber-Reinforced Plastic Pressure Hull Mod 1 for the Advanced Unmanned Search System (AUSS)." NOSC TR 1182 (Dec). Naval Ocean Systems Center, San Diego, CA.

* NRaD Technical Notes (TNs) are working documents and do not represent an official policy statement of the Naval Command, Control and Ocean Surveillance Center (NCCOSC), RDT&E Division (NRaD). For further information, contact the author(s).

Stachiw J. D. 1988. "Graphite-Fiber-Reinforced Plastic Pressure Hull Mod 2 for the Advanced Unmanned Search System (AUSS)." NOSC TR 1245 (Aug). Naval Ocean Systems Center, San Diego, CA.

Uhrich, R. W., J. Walton, and S. J. Watson. 1978. "Portable Test Range and its Application to Side-Looking Sonar." NOSC TR 258 (Jan). Naval Ocean Systems Center, San Diego, CA.

Uhrich, R. W., and S. J. Watson. 1992. "Deep-Ocean Search and Inspection: Advanced Unmanned Search System (AUSS) Concept of Operation." NRaD TR 1530 (Nov). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Uhrich, R. W., S. J. Watson, and G. R. Mackelburg (Eds.). 1992. "Advanced Unmanned Search System (AUSS) Surface Acoustic Link Description." NRaD TN 1706 (Oct). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.*

Vought Corporation. 1992. "Design Analysis and Operations Research for the Advanced Unmanned Search System (AUSS)." NRaD TD 2337 (Sep). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Walton, J. 1992. "Advanced Unmanned Search System (AUSS) At-Sea Development Test Report." NRaD TR 1537 (Dec). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Walton, J. 1992. "Advanced Unmanned Search System (AUSS) Testbed: FY 1987 Development Testing." NRaD TR 1525 (Nov). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Walton, J. 1992. "Advanced Unmanned Search System (AUSS) Testbed: Search Demonstration Testing." NRaD TR 1527 (Nov). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

Walton, J. 1992. "Evolution of a Search System: Lessons Learned with the Advanced Unmanned Search System." NRaD TR 1529 (Nov). Naval Command, Control and Ocean Surveillance Center, RDT&E Division, San Diego, CA.

* NRaD Technical Notes (TNs) are working documents and do not represent an official policy statement of the Naval Command, Control and Ocean Surveillance Center (NCCOSC), RDT&E Division (NRaD). For further information, contact the author(s).

APPENDIX A: IBM 7552 BACKPLANE MODIFICATIONS

IBM 7552 BACKPLANE

The IBM 7552 industrial computer uses a passive backplane design. Feature cards plug into the bus via two 96-pin (3 x 32) DIN connectors. The signals assigned to the backplane bus are composed of signals from three sources: IBM microchannel (16 bit), the IBM PC AT bus subset, and the IBM 7552 unique signals.

To work around software and hardware incompatibilities, the standard IBM 7552 bus was modified to make it capable of being 100 percent compatible with the full IBM PC AT Bus standard. To accomplish this, modifications were made to the IBM PC feature adapter. A side view of the adapter is shown below in figure A-1. This adapter accepts a standard IBM PC AT accessory card and then maps its bus signals to lines on the 7552 bus. A standard AT card mates to a 98-line bus via a 62-pin and a 36-pin edge card connector. The feature adapter has the appropriate mating edge connectors on one side into which the AT card is inserted. On the other end of the feature adapter are two 96-pin DIN connectors that mate to the 7552 backplane. In between, traces are etched that map AT bus signals to 7552 bus signals. For whatever reason, IBM chose not to support all the AT bus signals on the 7552 bus. The signals left off were

- DMA level 2 (DRQ2 and -DACK2);
- Interrupt levels 6 and 14 (IRQ6 and IRQ14);
- AT adapter cards that are bus masters (use the -MASTER line); and
- Zero wait state bus cycles.

To reinstate these signals and achieve sufficient compatibility to the IBM PC AT standard for our hardware and application software, the original 7552 CPU and memory cards were removed and replaced with third-party CPU/memory cards designed for operating in AT-compatible passive backplanes. The standard 7552 CPU cards incorporated changes that used some AT signals and some microchannel signals. This hybrid architecture created hardware and software incompatibilities with our surface computer designs, necessitating replacement of the 7552 CPU and memory cards. Once replaced, the 7552 bus was made compatible with the AT standard by adding the necessary missing signals back onto the bus via changes to the feature adapter signal mappings.

The bus signal definitions as delivered for the 7552 are shown in figure A-2.

The second of the two DIN connectors is mapped as shown in figure A-3.

The following two tables in figure A-4 and figure A-5 detail the mapping of AT bus signals to 7552 bus signals that occur on the feature adapter cradle cards.

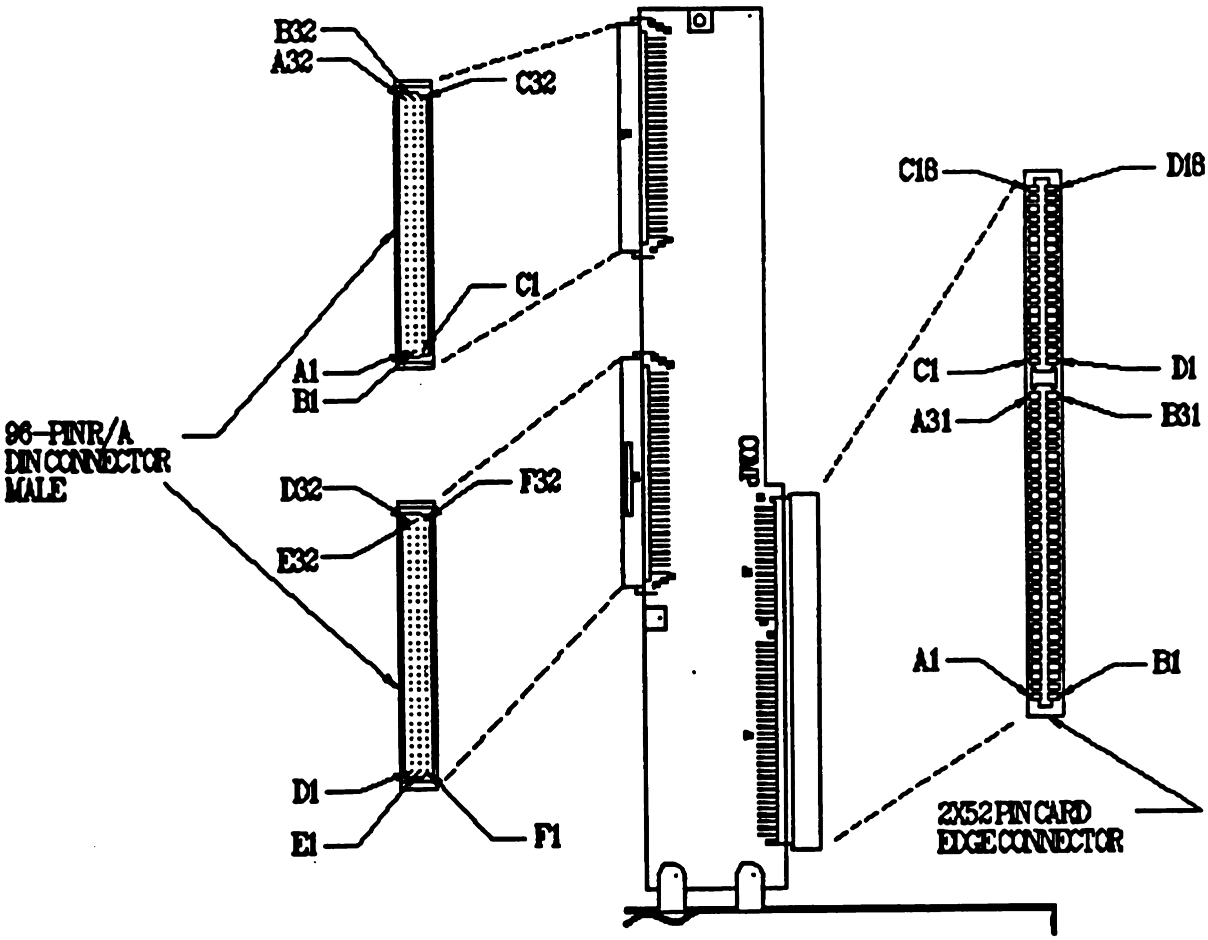


Figure A-1. IBM 7552 feature adapter.

I/0 PIN	SIGNAL NAME	I/O PIN	SIGNAL NAME	I/O PIN	SIGNAL NAME
C32	GND	B32	CHRD Y ADP (1)	A32	GND
C31	DC PWR GOOD	B31	DRQ7 (1)	A31	BKUP DISC
C30	+5 VDC	B30	-DACK7 (1)	A30	-DO 1
C29	-PR PWR CK	B29	DRQ6 (1)	A29	-IRQ8
C28	+12 VDC	B28	-DACK6 (1)	A28	-IRQ1
C27	-TEMP CK	B27	DRQ5 (1)	A27	-IRQ15
C26	-12 VDC	B26	-DACK5 (1)	A26	-IRQ14
C25	-P/S CK	B25	-XMEM W (1)	A25	-IRQ12
C24	+5 VDC	B24	DRQ0 (1)	A24	-IRQ11
C23	RESER VED (2)	B23	-XMEM R (1)	A23	-IRQ10
C22	GND	B22	-DACK0 (1)	A22	GND
C21	-CD DS 16	B21	IRQ15 (1)	A21	DPAR 1
C20	RESER VED (2)	B20	IRQ12 (1)	A20	D15
C19	-SBHE	B19	IRQ11 (1)	A19	D14
C18	- REFRE SH	B18	IRQ10 (1)	A18	D13
C17	-DS 16 RTN	B17	-IO CS16 (1)	A17	D12
C16	-SFDBK RTN	B16	-MEM CS16 (1)	A16	D11
C15	RESER VED (2)	B15	-BHE (1)	A15	D10
C14	CHRE SET	B14	LA0 (1)	A14	D9
C13	GND	B13	LA1 (1)	A13	D8
C12	CD CHRD Y	B12	LA2 (1)	A12	D7
C11	GND	B11	BALE (1)	A11	GND
C10	-CD SFDBK	B10	LA3 (1)	A10	D6
C09	+5 VDC	B09	T/C (1)	A09	D5
C08	CHRD YRTN	B08	LA4 (1)	A08	D4
C07	+5 VDC CONT	B07	LA5 (1)	A07	D3
C06	M/-IO	B06	IRQ3 (1)	A06	D2
C05	+12 VDC	B05	LA6 (1)	A05	D1
C04	-CMD	B04	IRQ4 (1)	A04	D0
C03	+5 VDC	B03	LA7 (1)	A03	DPAR 0
C02	-CHK	B02	IRQ5 (1)	A02	-DPAR EN
C01	GND	B01	LA8 (1)	A01	GND

(1) IBM PC AT Unique Signals
(2) Reserved

Figure A-2. IBM 7552 system bus backplane connector J01–J09.

I/O PIN	SIGNAL NAME	I/O PIN	SIGNAL NAME	I/O PIN	SIGNAL NAME
F32	GND	E32	LA9 (1)	D32	GND
F31	-TC	E31	LA10 (1)	D31	-S1
F30	+5 VDC	E30	IRQ7 (1)	D30	-S0
F29	ARB/- GNT	E29	CLK (1)	D29	A0
F28	+12 VDC	E28	LA11 (1)	D28	A1
F27	ARB3	E27	LA12 (1)	D27	A2
F26	-12 VDC	E26	DRQ1 (1)	D26	A3
F25	ARB2	E25	LA13 (1)	D25	A4
F24	+5 VDC	E24	-DACK1 (1)	D24	A5
F23	ARB1	E23	LA14 (1)	D23	A6
F22	GND	E22	DRQ3 (1)	D22	GND
F21	ARB0	E21	LA15 (1)	D21	A7
F20	RESER VED (2)	E20	-DACK3 (1)	D20	A8
F19	-IRQ7	E19	LA16 (1)	D19	A9
F18	-IRQ6	E18	-IOR (1)	D18	A10
F17	-IRQ5	E17	LA17 (1)	D17	A11
F16	-IRQ4	E16	-IOW (1)	D16	A12
F15	-IRQ3	E15	LA18 (1)	D15	A13
F14	-IRQ9	E14	-SMEM R (1)	D14	A14
F13	GND	E13	LA19 (1)	D13	A15
F12	- BURST	E12	-SMEM W (1)	D12	A16
F11	GND	E11	AEN (1)	D11	GND
F10	-PREEM PT	E10	IRQ9 (1)	D10	A17
F09	+5 VDC	E09	-SETUP 1 (1)	D09	A18
F08	-ADL (ALE)	E08	-SETUP 2 (1)	D08	A19
F07	-5 VDC	E07	-SETUP 3 (1)	D07	A20
F06	MADE 24	E06	-SETUP 4 (1)	D06	A21
F05	+12 VDC	E05	-SETUP 5 (1)	D05	A22
F04	-CD SETUP	E04	-SETUP 6 (1)	D04	A23
F03	+5 VDC	E03	-SETUP 7 (1)	D03	AUDIO GND
F02	OSC	E02	-SETUP 8 (1)	D02	GND
F01	GND	E01	-SETUP 9 (1)	D01	GND

(1) IBM PC AT Unique Signals
(2) Reserved

Figure A-3. IBM 7552 system bus backplane connector J10–J18.

AT PIN	SIGNAL NAME	7552 I/O	AT I/O PIN	SIGNAL NAME	7552 I/O PIN
A01	-CHCK	C02	B01	GND	A01 *
A02	D7	A12	B02	CHRESET	C14
A03	D6	A10	B03	+ 5 VDC	C03 *
A04	D5	A09	B04	IRQ9	E10
A05	D4	A08	B05	- 5 VDC	F07
A06	D3	A07	B06	DRQ2	F16 **
A07	D2	A06	B07	-12 VDC	C26 *
A08	D1	A05	B08	0WS	
A09	D0	A04	B09	+12 VDC	C28
A10	CHRDY ADP	B32	B10	GND	C01 *
A11	AEN	E11	B11	-SMEMW	E12
A12	LA19	E13	B12	-SMEMR	E14
A13	LA18	E15	B13	-IOW	E16
A14	LA17	E17	B14	-IOR	E18
A15	LA16	E19	B15	-DACK3	E20
A16	LA15	E21	B16	DRQ3	E22
A17	LA14	E23	B17	-DACK1	E24
A18	LA13	E25	B18	DRQ1	E26
A19	LA12	E27	B19 ***	-REFRESH	N/C
A20	LA11	E28	B20	CLK	E29
A21	LA10	E31	B21	IRQ7	E30
A22	LA9	E32	B22	IRQ6	F18 **
A23	LA8	B01	B23	IRQ5	B02
A24	LA7	B03	B24	IRQ4	B04
A25	LA6	B05	B25	IRQ3	B06
A26	LA5	B07	B26	-DACK2	F15 **
A27	LA4	B08	B27	T/C	B09
A28	LA3	B10	B28	BALE	B11
A29	LA2	B12	B29	+5 VDC	C30 *
A30	LA1	B13	B30	OSC	F02
A31	LA0	B14	B31	GND	D01 *

* Multiple 7552 pins used
** Signals added to mapping in ALL feature adapters
*** Signals removed from ALL feature adapters
**** Signal added to ONLY CPU and Adaptec feature adapters in FS machine

Figure A-4. AUSS IBM AT I/O pin to IBM 7552 I/O pin mapping.

AT PIN	SIGNAL NAME	7552 I/O	AT I/O PIN	SIGNAL NAME	7552 I/O PIN
C01	-BHE	B15	D01	-MEM CS16	B16
C02	A23	D04	D02	-IO CS16	B17
C03	A22	D05	D03	IRQ10	B18
C04	A21	D06	D04	IRQ11	B19
C05	A20	D07	D05	IRQ12	B20
C06	A19	D08	D06	IRQ15	B21
C07	A18	D09	D07	IRQ14	F14 **
C08	A17	D10	D08	-DACK0	B22
C09	-XMEMR	B23	D09	DRQ0	B24
C10	-XMEMW	B25	D10	-DACK5	B26
C11	D8	A13	D11	DRQ5	B27
C12	D9	A14	D12	-DACK6	B28
C13	D10	A15	D13	DRQ6	B29
C14	D11	A16	D14	-DACK7	B30
C15	D12	A17	D15	DRQ7	B31
C16	D13	A18	D16	+5 VDC	F03 *
C17	D14	A19	D17	-MASTER	F19 ****
C18	D15	A20	D18	GND	D32 *

* Multiple 7552 pins used

** Signals added to mapping in ALL feature adapters

*** Signals removed from ALL feature adapters

**** Signal added to ONLY CPU and Adaptec feature adapters in FS machine

Figure A-5. AUSS IBM AT I/O pin to IBM 7552 I/O pin mapping.

The IRQ6 signal was removed from the bus when IBM created the 7552 because it was the interrupt assigned to the floppy disk drive controller in the AT bus. In the 7552 system, IBM decided to use a microchannel controller that used a different architecture. Likewise, the same was true for the IRQ14 assigned to the hard disk controller for the AT bus. The DRQ2 and -DACK signals supported use of DMA channel 2 transfer of data to and from the floppy disk controller in an AT system. Since the 7552 used a microchannel controller for its factory disk subsystems, these two signals were likewise removed from the 7552 bus by IBM. The -MASTER signal is used in an AT system to allow a coprocessor on an I/O card to take over the bus from the main CPU. In the AUSS surface computers, the only card that uses this is the Adaptec 1542B in the FS. The 0WS signal is not needed by any of the cards in the system and therefore has not been reinstated into the AUSS 7552 bus. CPU cards used in the 7552 systems are CPU/memory cards whose system memory is located on the same card as the CPU. As such, no backplane bus is required to interface memory to CPU. A local bus on the card handles data transfer between CPU and main memory. The -REFRESH signal was removed from the 7552 backplane because it caused an incompatibility between an early CPU card and display card combination, and the CPU/memory card architecture eliminated the need to send a memory refresh signal onto the backplane to support a separate memory card.

APPENDIX B: CONTROL VAN WIRING

CURRENT CONTROL VAN WIRING

Figure B-1 illustrates the interconnection wiring required in the current AUSS surface computers. The wiring requirements are derived from two sources: (1) interconnections between computers and (2) wiring from computers to display monitors and video repeaters. Since AUSS uses 13 display monitors in the van and requires that display screens be switchable to a couple of auxiliary monitors and a tape recorder, the video display wiring has become unwieldly. The number of keyboards has also become a problem because the number of computers in the van has grown to six 7552 machines and a laptop.

PLANNED CONTROL VAN WIRING

Figure B-2 illustrates the near-term design goal of the control van surface computer architecture. This is the layout that AUSS has been moving to as a delivery configuration. Note that the number of keyboards and monitors has been reduced by using an electronic keyboard/monitor switch box. This is put to use to let one keyboard and monitor set be used for the LOG, FS, and data docker computers. The FS and data docker machines require only infrequent use of a keyboard and monitor, and therefore, the switchbox is an expedient way to address this problem. The requirement to redirect displays to auxiliary monitors and a VCR, however, still necessitates a lot of wiring given the number of monitors.

LONG-TERM RECOMMENDED CONTROL VAN WIRING

Figure B-3 depicts how the wiring would be simplified if the recommendations made in the body of the report were adopted and the software rewritten to use X Windows architecture. Note that the number of monitors is reduced from 13 to 5, there are no video repeater boxes, the serial RS-232 cables are eliminated, and there is only one keyboard/monitor switch. Not only is the wiring greatly simplified, but the spares problem is also significantly reduced.

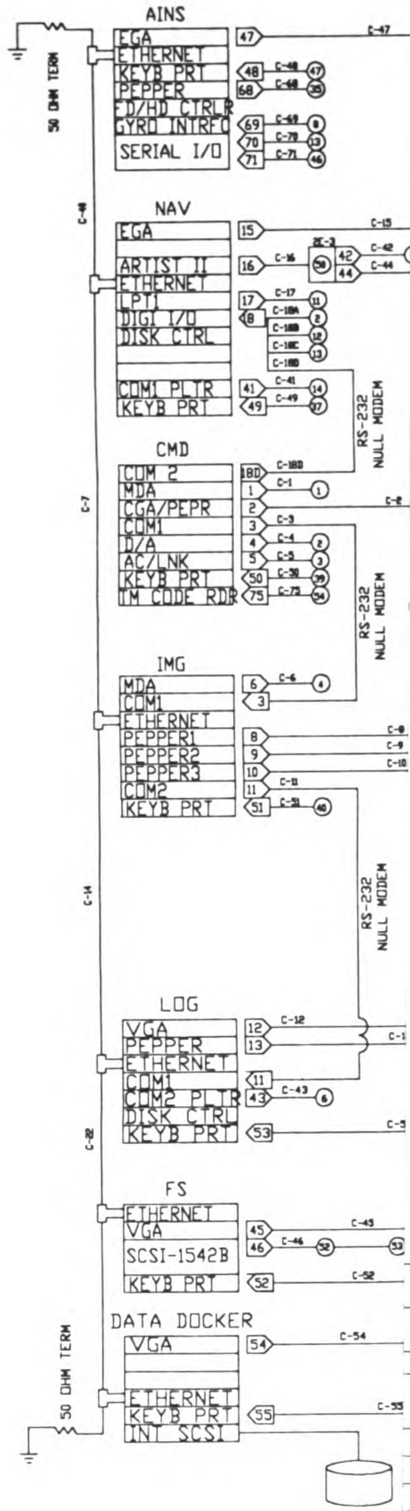
DWG NO 0129038 SH 1

REVISIONS			
REV	DESCRIPTION	DATE	APPROVED

THIS DRAWING HAS BEEN GENERATED AND IS MAINTAINED BY A CAD SYSTEM. CHANGES SHALL ONLY BE INCORPORATED AS DIRECTED BY THE GOVERNMENT DESIGN ACTIVITY.

NOTE:

1. SEE NCCOSC RDT&E DIVISION TECHNICAL REPORT REGARDING COMPUTER CIRCUIT CARD SWITCH SET



CABLES

1. EGA TO MULTISYNC - 12' CABLE
2. VGA-M TO VGA-F 6' CABLE
3. RS-232 NULL MODEM CABLE
4. TWO-WIRE ANALOG DEPTH CABLE TO HONEYWELL
5. 50 CONDUCTOR RIBBON CABLE TO AC/LNK COMPUTER
6. EGA TO MULTISYNC - 12' CABLE
7. NAV TO IMAGES: ETHERNET, RG-58
8. VGA-M TO VGA-F 6' CABLE
9. VGA-M TO VGA-F 6' CABLE
10. VGA-M TO VGA-F 6' CABLE
11. RS-232 NULL MODEM CABLE
12. VGA-M TO VGA-F 6' CABLE
13. VGA-M TO VGA-F 6' CABLE
14. IMAGES TO LOGGER: ETHERNET
15. EGA-M TO EGA-F 6' CABLE
16. EGA-M TO EGA-F 6' CABLE
17. PARALLEL PRINTER CABLE
18. DIGI-CARD 'OCTOPUS' CABLE
    - 18A. HONEYWELL RS-906
    - 18B. MINI-RANGER III
    - 18C. TRIMBLE GPS/LORAN
    - 18D. RS-232 NULL MODEM CABLE
19. VGA-M TO MULTISYNC - 12'
20. VGA-M TO VGA-F 6' CABLE
21. VGA-M TO MULTISYNC - 12'
22. LOGGER TO FS: ETHERNET RG-58
23. VGA-M TO MULTISYNC - 12'
24. VGA-M TO VGA-F 6' CABLE
25. VGA-M TO MULTISYNC - 12'
26. VGA-M TO VGA-F - 6' CABLE
27. SHARED KEYBOARD EXTENSION CABLE
28. VGA-M TO VGA-F - 6' CABLE
29. VGA-M TO VGA-F - 6' CABLE
30. -> 39. VGA-M TO VGA-F - 6' CABLE
40. NAV TO AINS: ETHERNET RG-58
41. CABLE SEATRAC TO PLOTTER
42. EGA TO MULTISYNC - 12' CABLE
43. (TO BE DETERMINED)
44. EGA-M TO EGA-F - 100' CABLE
45. VGA-M TO VGA-F - 6' CABLE
46. 50 CONDUCTOR CABLE
47. EGA-M TO EGA-F - 6' CABLE
48. AINS KEYBOARD CABLE
49. NAV KEYBOARD CABLE
50. CMD KEYBOARD CABLE
51. IMG KEYBOARD CABLE
52. FS KEYBOARD EXTENSION CABLE
53. LOG KEYBOARD EXTENSION CABLE
54. VGA-M TO VGA-F 6' CABLE
55. DATA DOCKER KEYBOARD EXTENSION CABLE
56. EGA-M TO MULTISYNC - 12' CABLE
57. EGA-M TO EGA-F - 100' CABLE
58. VGA-M TO MULTISYNC - 12' CABLE
59. VGA-M TO 15 PIN D-SUB CONNECTOR
60. COAXIAL CABLE - RG-57
61. EGA-M TO EGA-F - 100' CABLE
63. EGA-M TO MULTISYNC - 12' CABLE
64. EGA-M TO MULTISYNC - 100' CABLE
68. VGA-M TO MULTISYNC - 12' CABLE
69. RS-232 CABLE
70. RS-232 CABLE
71. RS-232 CABLE
73. 4 CONDUCTOR SUPER-VIDEO CABLE
74. 7 CONDUCTOR SUPER-VHS TO 4 CONDUCTOR DIN CABLE
75. COAXIAL CABLE - RG-58

DESCRIPTION OR NOMENCLATURE	MATERIAL/SPEC/REF DESG	FIND NO

PARTS LIST

NAVAL OCEAN SYSTEMS CENTER
SAN DIEGO, CA 92152-5000

NEARTERM PLANNED INTERCONNECTION DIAGRAM

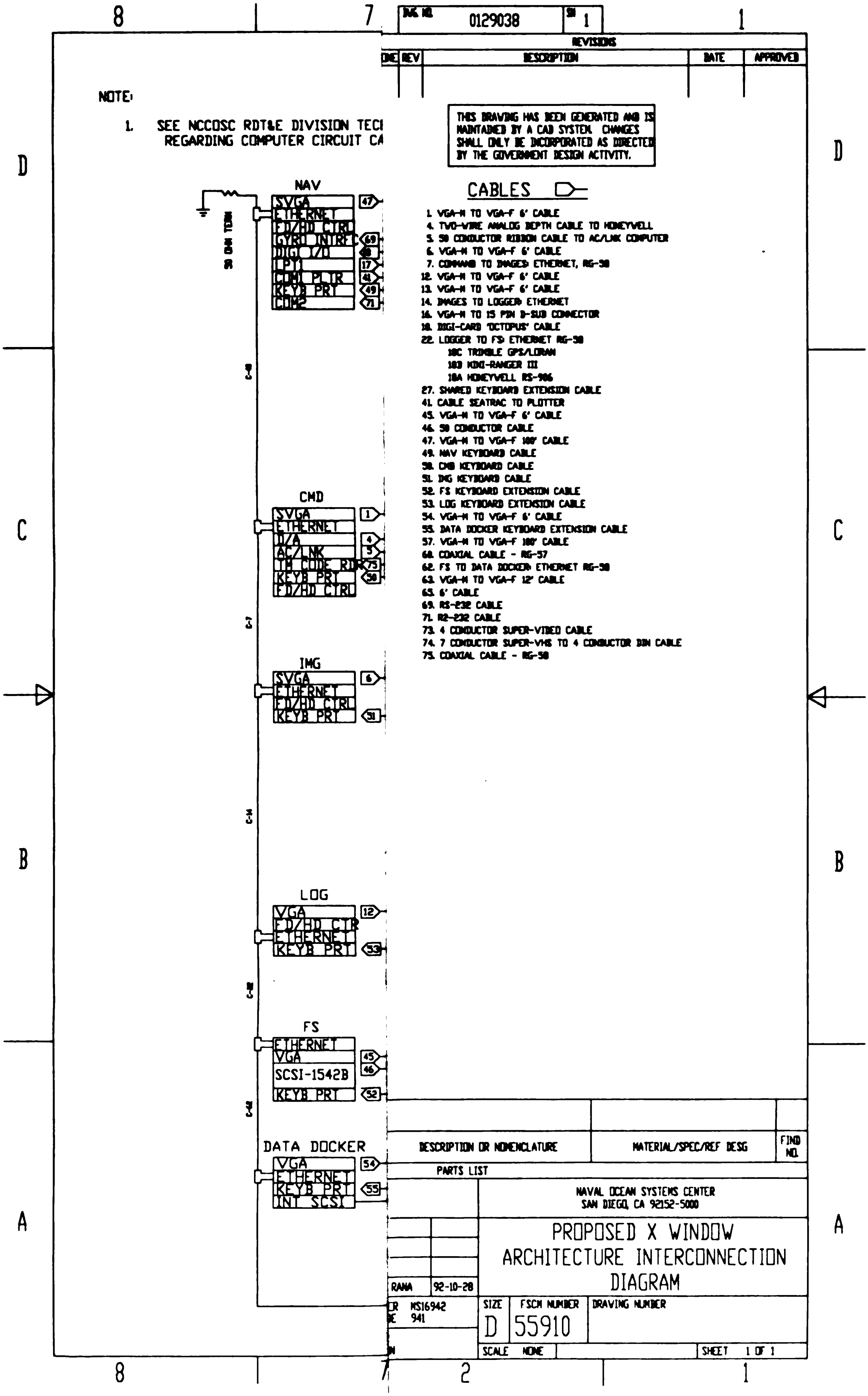
NA 92-10-28

MS16942 941

SIZE	FSCM NUMBER	DRAWING NUMBER
D	55910	

SCALE NONE SHEET 1 OF 1

...ned interconnection diagram.



…rchitecture interconnection diagram.

APPENDIX C: COMMAND (CMD) COMPUTER BOARD JUMPERS/SWITCH

CMD LIST OF ADAPTERS

1. CPU card: Texas Microsystems model B386/20 with 2-Mbyte memory
2. Menu display card: Generic Monochrome Display Adapter (MDA)
3. Graphics display card: Number Nine Pepper SGT Plus with 1-Mbyte memory
4. ROMDISK card: Industrial Computer Source ROMDISK model PCE/2
5. I/O card: Kouwell model KW-524H dual serial port + parallel port
6. Acoustic link interface card: custom NRaD 7552 form factor card
7. Time code reader card: Bancomm, division of Datum, model PC03XT

JUMPER/SWITCH SETTINGS

1. CPU card:
   Switch #1: 1 to 4 OFF; normal operation versus manufacturing test

   Reference: *B386 Central Processor Unit Card User Manual*, (c) 1987
   by Texas Microsystems, Inc.

2. Menu Display card:
   No jumpers or switches to set

3. Graphics Display card:
   Switch #1: 1 CLOSED; CGA emulation, 640 × 480 × 8 noninterlaced, SGT PLUS
   no self test, separate sync signals, interface address at C700
   2 to 10 OPEN

   Reference: *Pepper SGT Plus Quick Installation Card and User's Guide*, (c) 1989
   by Number Nine Computer Corp.

4. ROMDISK CARD:
   Switch #1: 1 to 4 ON
   Single ROMDISK installation w/UV or flash EPROM
   Switch #2: 1 OFF
   2 ON
   3 OFF
   During operation, autoboot as drive A:
   Switch #2: 1 to 3 ON
   Set during reprogramming of UV EPROMs
   Switch #3: 1 to 2 OFF
   3 to 7 ON
   8 OFF

1.2-Mbyte 5-and-1/4-inch floppy emulation
UV EPROMS
1.2-Mbyte diskette emulation
Floppy versus hard drive emulation
DMA-1 memory transfer
Autoset of floppy designator

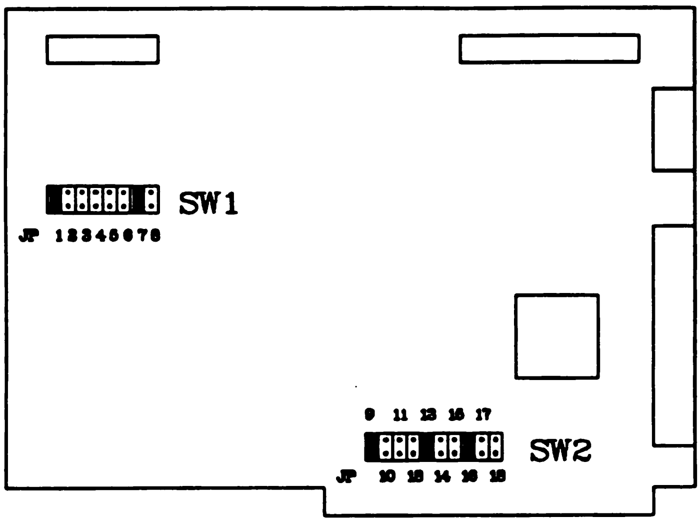
Reference: *Model PCE2 Reference Manual*, (c) 1989 by Industrial Computer Source

5. I/O card:

Switch #1:	1 ON	Switch #2:	9 ON
	2 to 6 OFF		10 to 12 OFF
	7 ON		13 ON
	8 OFF		14 to 15 OFF
			16 ON
			17 to 18 OFF

Jumpers as shown below:

KOUWELL KW-624H I/O CARD



Reference: *Kouwell KW-524H User's Manual*

6. Acoustic link card:
No jumpers or switches to set

Reference: NRaD drawings

0122705	CONT CONSOLE/ACOUSTIC LINK INTERFACE ASSY
0122706	CONT CONSOLE/ACOUSTIC LINK INTERFACE PWB
0122707	CONT CONSOLE/ACOUSTIC LINK INTERFACE SCHEM

7. Time code reader card:
Switch U35: 1 ON
2 OFF

3 ON
4 OFF

Reference: *PC03XT Time Code Reader Module Operation & Technical Manual*, (c) 1989 by Datum, Inc.

APPENDIX D: IMAGES (IMG) COMPUTER BOARD JUMPERS/SWITCH

IMG LIST OF ADAPTERS

1. CPU card: Texas Microsystems model B386/20 with 2-Mbyte memory
2. Menu display card: generic monochrome display adapter (MDA)
3. Graphics display card #1: Number Nine Pepper SGT Plus with 1-Mbyte memory
4. Graphics display card #2: Number Nine Pepper SGT Plus with 1-Mbyte memory
5. Graphics display card #3: Number Nine Pepper SGT Plus with 1-Mbyte memory
6. I/O card: Kowell model KW-524H dual serial port + parallel port
7. Ethernet card: Western Digital 8003E 8-bit interface card

JUMPER/SWITCH SETTINGS

1. CPU card:

   Switch #1: 1 to 4 OFF; normal operation versus manufacturing test

   Reference: *B386 Central Processor Unit Card User Manual*, (c) 1987 by Texas Microsystems, Inc.

2. Menu display card:

   No jumpers or switches to set

   Reference: none

3. Graphics display card #1:

   Switch #1: 1 to 2 CLOSED; no emulation, 640 × 480 × 8 noninterlaced
   3 to 7 OPEN SGT Plus mode, self test, interface address
   8 CLOSED at C400, separate sync signals
   9 to 10 OPEN

   Reference: *Pepper SGT Plus Quick Installation Card and User's Guide*, (c) 1989 by Number Nine Computer Corp.

4. Graphics display card #2:

   Switch #1: 1 to 2 CLOSED; no emulation, 640 × 480 × 8 noninterlaced
   3 to 9 OPEN SGT Plus mode, no self test, interface address
   10 CLOSED at CF00

   Reference: *Pepper SGT Plus Quick Installation Card and User's Guide*

5. Graphics display card #3:

   Switch #1: 1 to 2 CLOSED; no emulation, 640 × 480 × 8 noninterlaced
   3 to 10 OPEN SGT Plus mode, no self test, interface address
   at C700

   Reference: *Pepper SGT Plus Quick Installation Card and User's Guide*

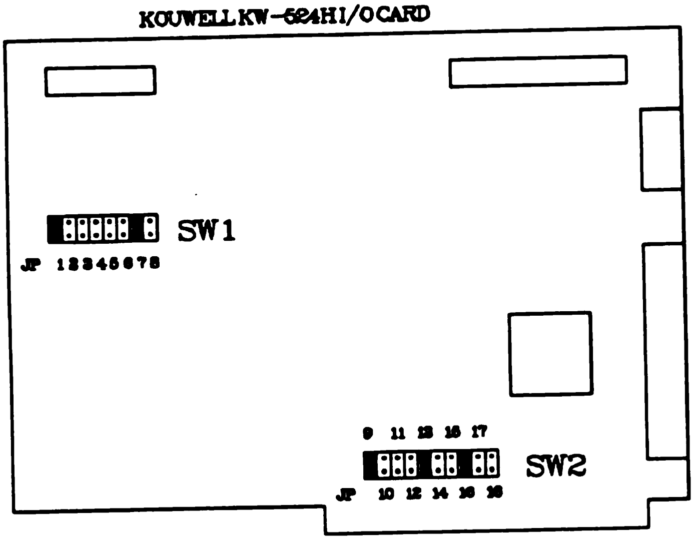
6. I/O card:

Switch #1: 1 ON
2 to 6 OFF
7 ON
8 OFF

Switch #2: 9 ON
10 to 12 OFF
13 ON
14 to 15 OFF
16 ON
17 to 18 OFF

Jumpers as shown below:

KOUWELL KW-524H I/O CARD



Reference: *Kouwell KW-524H User's Manual*

7. Ethernet card:

Jumper settings:

W1 _X X_X ; board address: 280H; 4 wait states for bus
W2 _ _ _X_ _ ; interrupt 5 used
W3 X X X X X X <—bottom ; thin ethernet
W4 _ ; IEEE 802.3 ethernet
W5 X ; standard thin ethernet segment
W6 _ _X X_ ; boot ROM enabled @ D8000
W7 X_ <—bottom ; 8-K RAM buffer
W8 X_ <—bottom ; 8-K RAM buffer
W9 _X ; 16-K boot ROM
W10 X_ ; 16-K boot ROM
W11 left and center pin jumpered

Note: _ —> OPEN
X —> JUMPERED

Reference: *EtherCard PLUS User Installation Guide*, (c) 1987
by Western Digital Corporation

APPENDIX E: LOGGER (LOG) COMPUTER BOARD JUMPERS/SWITCH

LOG LIST OF ADAPTERS

1. CPU card: Diversified Technology model CAT1000: 486/33 MHz with
   8-Mbyte memory

2. Menu display card: ATI Basic VGA display adapter

3. Graphics display card: Number Nine Pepper SGT Plus with 1-Mbyte memory

4. IDE disk and serial I/O card: Super IDE I/O card (2S, 1P, 1G ports) model PT604
   IDE hard disk: Maxtor model 7080 AT
   Floppy disk: 1.44-Mbyte generic 3-and-1/2-inch floppy drive

5. Ethernet card: Western Digital 8003EBT 8-bit interface card

JUMPER/SWITCH SETTINGS

1. DT 486/33 MHz CPU card:

Switch #1:	1 to 3 OPEN	; normal operation versus manufacturing test
	4 CLOSED	; color monitor
	5 to 6 OPEN	; AT keyboard, normal interrupt for mouse

   Reference: *Diversified Technology CAT1000-ASSY.NO.-912000975 Rev. 1.1 AT Compatible Configuration Guide*, (c) 1990

2. Menu display card:
   No jumpers or switches to set

   Reference: *VGA BASIC–16 User's Guide Ver 1.0*, (c) 1990 by Advanced Technologies, Inc.

3. SGT Plus graphics display card:

Switch #1:	1 to 2 CLOSED	; no emulation
	3 to 9 OPEN	; 640 × 480 × 8 noninterlaced, SGT PLUS, no self-test, separate horiz & vert sync
	10 CLOSED	; 8 OPEN + 10 CLOSED: interface address @ CF00

   Reference: *Pepper SGT Plus Quick Installation Card and User's Guide*, (c) 1989 by Number Nine Computer Corp.

4. IDE + I/O card:

JMP 1:	X_	; floppy interface enabled
JMP 3:	X_	; floppy port address: 3F1–3F7
JMP 4:	X_	; IDE fixed disk interface enabled
JMP 2:	X_	; IDE port address 1F0–1F7
JMP 12:	X_	; not used
JMP 11:	_X	; AT HDC

```
JMP  5:   X_   ; printer port address: 378-37F
JMP 10:   X_   ; printer port enabled
JMP  6:   X_   ; COM1 serial port address: 3F8-3FF
JMP  9:   X_   ; COM1 enabled
JMP  7:   X_   ; COM2 serial port address: 2F8-2FF
JMP  8:   X_   ; COM2 enabled
JMP 14:   X_   ; game port enabled
```

Note: each jumper block is composed of three pins in a horizontal row:
X_ means left and center pins are jumpered together
_X means right and center pins are jumpered together

Reference: *Super IDE I/O Card User's Manual Model PT-604*
Unknown manufacturer

5. Ethernet card:

Jumper settings:

```
W1   _ X X _ X                  ; board address: 280H; 4 wait states for bus
W2   _ _ _ X _ _                ; interrupt 5 used
W3   X X X X X X <---bottom     ; thin ethernet
W4   _                          ; IEEE 802.3 ethernet
W5   X                          ; standard thin ethernet segment
W6   _ _ X X _                  ; boot ROM enabled @ D8000
W7   X _ <---bottom             ; 8-K RAM buffer
W8   X _ <---bottom             ; 8-K RAM buffer
W9   _ X                        ; 16-K boot ROM
W10  X _                        ; 16-K boot ROM
W11  right and center pin jumpered
```

Note: jumper blocks are either dual row or dual column of pins
_ ---> OPEN across two pins
X ---> JUMPERED across two pins

Reference: *EtherCard PLUS User Installation Guide*, (c) 1987 by
Western Digital Corporation

APPENDIX F: NAVIGATION (NAV)/SEATRAC COMPUTER JUMPERS/SWITCH

NAV LIST OF ADAPTERS

1. CPU card: Texas Microsystems model B386/20 with 2-Mbyte memory
2. Menu display card: generic EGA display adapter
3. Graphics display card: Control Systems Artist II graphics display card
4. Floppy/hard disk card: Western Digital WD1003-WA2 MFM controller card
5. Ethernet card: Western Digital 8003EBT 8-bit interface card
6. Multiport serial I/O card: DigiCHANNEL COM/Xi 8 port serial I/O card
7. AT I/O card: Kowell model KW-524H dual serial and parallel port card

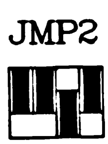
JUMPER/SWITCH SETTINGS

1. Texas Microsystems B386/20 CPU card:
   Switch #1: 1 to 4 OFF; normal operation versus manufacturing test

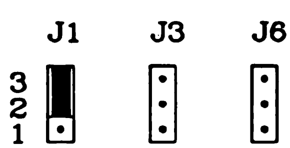
   Reference: *B386 Central Processor Unit Card User Manual*, (c) 1987 by Texas Microsystems Inc.

2. Menu display card:
   Switch #1: 1 OPEN
   2 to 3 CLOSED
   4 OPEN
   5 to 6 CLOSED

   Jumpers:



3. ARTIST II graphics display card
   Jumpers (J1, J3, J6):



   Reference: *ARTIST 2 Graphic Controller User's Guide*, (c) 1986 by Control Systems Inc.

4. Floppy/hard disk controller card:

Jumpers:

E2<-->E3
E5<-->E6
E7<-->E8

5. Ethernet card:

Jumper settings:

W1	_ X X _ X	; board address: 280H; 4 wait states for bus
W2	_ X_ _ _ _	; interrupt 3 used
W3	X X X X X X <---bottom	; thin ethernet
W4	_	; IEEE 802.3 ethernet
W5	X	; standard thin ethernet segment
W6	_ _ _X X	; boot ROM enabled @ CC000
W7	X _ <---bottom	; 8-K RAM buffer
W8	X _ <---bottom	; 8-K RAM buffer
W9	_ X	; 16-K boot ROM
W10	X _	; 16-K boot ROM
W11	right and center pin jumpered	

Note: jumper blocks are either dual row or dual column of pins

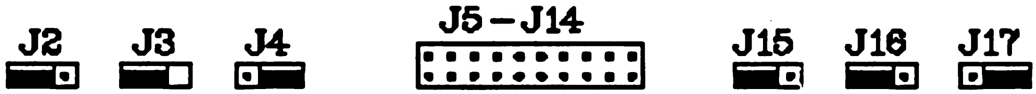
_ ---> OPEN across two pins

X ---> JUMPERED across two pins

Reference: *EtherCard PLUS User Installation Guide*, (c) 1987 by Western Digital Corporation

6. DigiCHANNEL COM/Xi Multiport serial card:

Jumpers:

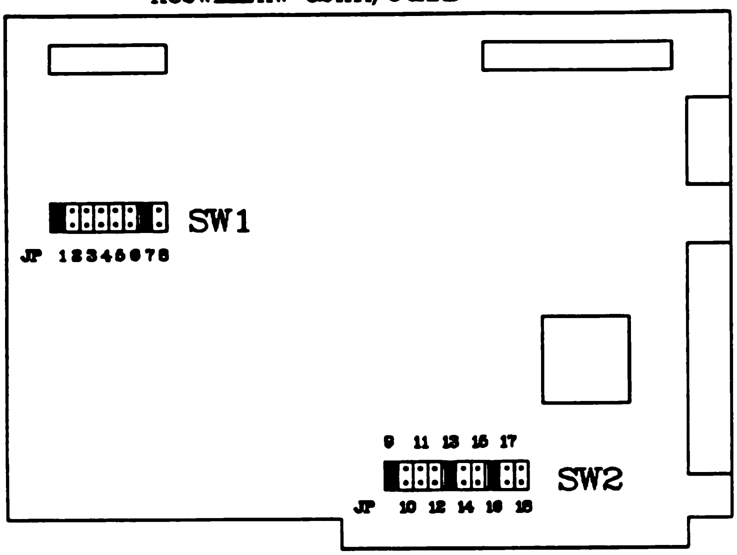


7. AT I/O card

Switch #1: 1 ON
2 to 6 OFF
7 ON
8 OFF

Switch#2: 9 ON
10 to 12 OFF
13 ON
14 to 15 OFF
16 ON
17 to 18 OFF

Jumpers as shown below:



Reference: *Kouwell KW-524H User's Manual*

APPENDIX G: AUSS INTEGRATED NAVIGATION SYSTEM (AINS)

AINS LIST OF ADAPTERS

1. CPU card: Texas Microsystems model B386/20 with 2-Mbyte memory
2. Menu display card: ATI Wonder 800 EGA display adapter
3. Graphics display card: Number Nine Pepper SGT+ graphics display card
4. IDE disk and serial I/O card: Identity model IDM10 IDE+I/O card
   Floppy disk: 1.44-Mbyte generic 3-and-1/2-inch floppy drive
5. Ethernet card: Western Digital WD8003E 8-bit interface card
6. Synchro to digital converter: ILC Data Device model SDC-36015 synchro to digital converter card

JUMPER/SWITCH SETTINGS

1. Texas Microsystems B386/20 CPU card:
   Switch #1: 1 to 4 OFF; normal operation versus manufacturing test

   Reference: *B386 Central Processor Unit Card User Manual*, (c) 1987
   by Texas Microsystems, Inc.

2. Menu display card:
   No switches or jumpers

   Reference: *EGA Wonder 800+ User's Guide*, (c) 1987 by ATI Technologies, Inc.

3. SGT Plus graphics display card:

Switch #1: 1 to 2	CLOSED	; no emulation
3 to 7	OPEN	; 640 × 480 × 8 noninterlaced, SGT PLUS, no self-test
8	CLOSED	; see #10 below
9	OPEN	; separate synch signals
10	CLOSED	; 8 OPEN + 10 CLOSED: interface address @ CC00

   Reference: *Pepper SGT Plus Quick Installation Card and User's Guide*, (c) 1989
   by Number Nine Computer Corp.

4. IDE + I/O card:
   Jumpers:

JP6- 1: A SIDE	; LPT1 for printer port
JP6- 2: A SIDE	; COM2 assigned to JP4
JP6- 3: A SIDE	; enable printer port
JP6- 4: A SIDE	; enable COM2
JP6- 5: A SIDE	; enable COM1
JP6- 6: A SIDE	; COM1 assigned to JP3

JP6- 7: A SIDE ; enable FDD controller
JP6- 8: A SIDE ; enable IDE controller
JP6- 9: A SIDE ; enable game port
JP6-10: B SIDE ; disable mouse port

JP7- 3: Jumper ; use IRQ5 for bus mouse if mouse enabled

JP8- 1: B SIDE ; IRQ3 for COM2
JP8- 2: A SIDE ; IRQ4 for COM1
JP8- 3: no jumper
JP8- 4: no jumper

JP9 : Jumper 2-3; IRQ7 for printer port

5. Ethernet card:

Jumper settings:

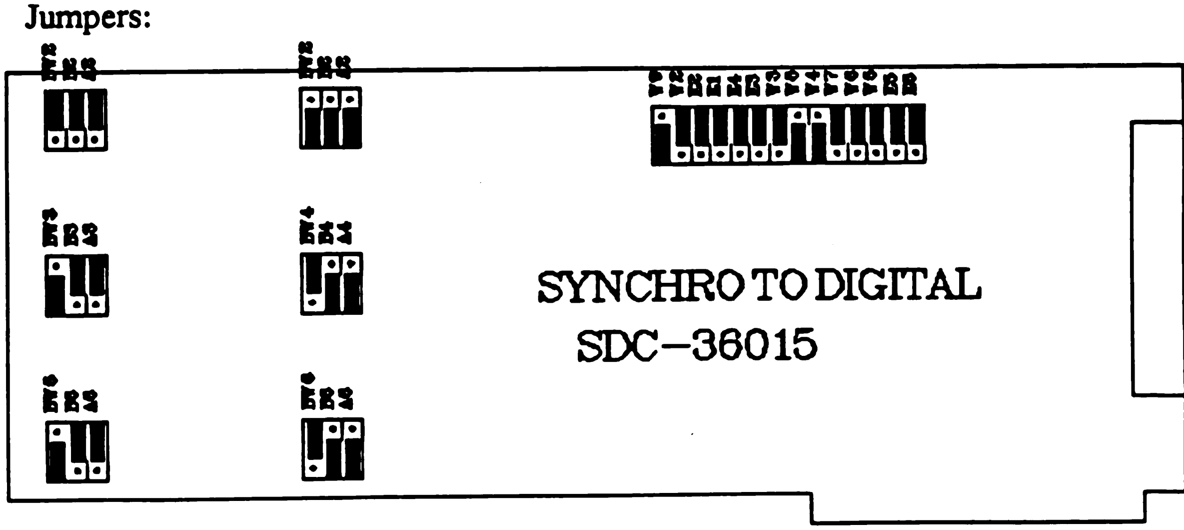
W1	_ X X _ X	; board address: 280H; 4 wait states for bus
W2	_ X _ _ _ _	; interrupt 3 used
W3	X X X X X X <--bottom	; thin ethernet
W4	_	; IEEE 802.3 ethernet
W5	X	; standard thin ethernet segment
W6	_ _ _ X X	; boot ROM enabled @ CC000
W7	X _ <--bottom	; 8-K RAM buffer
W8	X _ <--bottom	; 8-K RAM buffer
W9	_ X	; 16-K boot ROM
W10	X _	; 16-K boot ROM
W11	right and center pin jumpered	

Note: jumper blocks are either dual row or dual column of pins
_ —> OPEN across two pins
X —> JUMPERED across two pins

Reference: *EtherCard PLUS User Installation Guide*, (c) 1987
by Western Digital Corporation

6. DDC synchro to digital converter:

Jumpers:



APPENDIX H: FILE SERVER (FS) COMPUTER JUMPERS/SWITCH

FS LIST OF ADAPTERS

1. CPU card: Diversified Technology model CAT1000: 486/33 MHz with 8-Mbyte memory
2. Menu display card: ATI Basic VGA display adapter
3. SCSI Host adapter card: Adaptec 1542B SCSI controller
4. Ethernet card: Western Digital WD8013EBT ethernet card

JUMPER/SWITCH SETTINGS

1. DT 486/33 MHz CPU card:

   Switch #1: 1 to 3 OPEN ; normal operation versus manufacturing test
   4 CLOSED ; color monitor
   5 to 6 OPEN ; AT keyboard, normal interrupt for mouse

   Reference: *Diversified Technology CAT1000-ASSY.NO.-912000975 Rev. 1.1 AT Compatible Configuration Guide*, (c) 1990

2. Menu display card:

   No jumpers or switches to set

   Reference: *VGA BASIC-16 User's Guide Ver 1.0*, (c) 1990 by Advanced Technologies, Inc.

3. SCSI host adapter:

   Jumper J5:

   Pin 1 – OPEN ; synchronous transfer NOT enabled
   Pin 2 – OPEN ; diagnostics OFF
   Pin 3 – OPEN ; SCSI parity ENABLED
   Pin 4 – OPEN
   Pin 5 – OPEN
   Pin 6 – OPEN ; pins 4, 5, 6 set to OPEN sets SCSI address at 7
   Pin 7 – OPEN
   Pin 8 – CLOSED ; 7 and 8 set DMA channel to 5
   Pin 9 – OPEN
   Pin 10 – CLOSED
   Pin 11 – OPEN ; 9, 10, 11 set interrupt to 11
   Pin 12 – OPEN
   Pin 13 – OPEN ; DMA transfer speed set to 5 Mbytes/sec

   Jumper J6:

   Pin 1 – CLOSED ; BIOS ENABLED
   Pin 2 – OPEN
   Pin 3 – OPEN
   Pin 4 – OPEN ; 2, 3, 4 not used
   Pin 5 – OPEN ; auto sense ENABLED

Jumper J7:

Pin 1 – OPEN	; primary floppy address used
Pin 2 – CLOSED	
Pin 3 – OPEN	
Pin 4 – OPEN	; pins 2, 3, and 4: AT I/O port address set to 330 H
Pin 5 – OPEN	
Pin 6 – OPEN	; pins 5 and 6 set BIOS wait state to 0 nano sec.
Pin 7 – OPEN	
Pin 8 – OPEN	; 7 and 8 set BIOS base address @ DC000

Jumper J8:

Pin 1 – CLOSED	; floppy enabled
Pin 2 – CLOSED	
Pin 3 – OPEN	; 2 and 3 set DMA request to 2
Pin 4 – CLOSED	
Pin 5 – OPEN	; 4 and 5 set DMA ACK to 2
Pin 6 – CLOSED	
Pin 7 – OPEN	; 6 and 7 set INT request to 6
Pin 8 – OPEN	; dual speed disabled

Jumper J9 – DMA/Interrupt Selection:

Pin 1 – OPEN	; DMA request 0 NOT used
Pin 2 – CLOSED	; DMA request 5 selected
Pin 3 – OPEN	; DMA request 6 NOT used
Pin 4 – OPEN	; DMA request 7 NOT used
Pin 5 – OPEN	; DMA ACK 0 NOT used
Pin 6 – CLOSED	; DMA ACK 5 selected
Pin 7 – OPEN	; DMA ACK 6 NOT used
Pin 8 – OPEN	; DMA ACK 7 NOT used
Pin 9 – OPEN	; INT request 9 NOT used
Pin 10– OPEN	; INT request 10 NOT used
Pin 11– CLOSED	; INT request 11 selected
Pin 12– OPEN	; INT request 12 NOT used
Pin 13– OPEN	; INT request 14 NOT used
Pin 14– OPEN	; INT request 15 NOT used

Reference: *AHA-1540B/1542B Installation Guide*, (c) 1990 by Adaptec, Inc.

4. Ethernet card:

Jumper Settings:

WO	_ _	; 1 Wait state
W1	_ _ _ _ _ X _ _ _ _	; Board address: 280H
W2	_ _ _ _ _ _ _ X	; Interrupt 3 used
W3	right and center pins jumpered on both rows	; Thin Ethernet
W6	_ _ _ X	; No boot ROM

W9 No jumpers needed ; ROM size(ignored becaused of W6)
W15 X_X ; ROM address(ignored becaused of W6)

Note:

_ ---> OPEN across two pins
X ---> JUMPERED across two pins

Reference: *EtherCard PLUS16 User Installation Guide*, (c) 1989 by Western Digital Corporation

EXTERNAL DEVICES

There are three external drives connected to the FS in an external enclosure with a separate power supply. The external drives are a generic 3-and-1/2-inch 1.44-Mbyte floppy drive, an Archive model 4520 NT DAT drive, and a Maxtor LXT213 SCSI hard disk. Cabling from the Adaptec 1542B SCSI controller card runs from the 7552 enclosure to the external disk enclosure.

APPENDIX I: COMMAND (CMD) COMPUTER BOOT CONFIGURATION

CMD COMPUTER BOOT PROCESS

The CMD computer is a stand-alone workstation that boots up off a ROM disk configured to look like a floppy drive. Since there is no connection to a network from this machine, the boot process is a simple two-step process:

- Power up the two display monitors hooked up to the machine
- Power up the CMD machine

BOOT FILES

Since this is a stand-alone machine, there are only two files that control the bootup of this machine:

- CONFIG.SYS
- AUTOEXEC.BAT

The CONFIG.SYS file is executed first on power-up, then execution of the AUTOEXEC.BAT follows. An examination of the file AUTOEXEC.BAT shows that a program called 'setupcmd' is executed, then DESQview is invoked by the command 'dv.' 'Setupcmd' is a utility that eliminates the cursor from the screen. DESQview is set up to auto-invoke the command console applications programs under DESQview via the use of the ! macro script. As such, powering up the machine will get the user into the command console program without having to hit a single keystroke.

BOOT FILE LISTINGS

Listed below are the contents of both files.

CONFIG.SYS

```
shell=a:\COMMAND.COM /P/E:600
```

AUTOEXEC.BAT

```
echo off
cd\d
numon
mode co80
setupcmd
mode mono
dv
mode co80
cls
prompt $t$h$h$h$h$h$h $p$g
```

FILE LIST ON CMD BOOT DISK

```
  Directory of A:\

COMMAND  COM      25307 03-17-87  12:00p
RDCOMP2  EXE      29970 09-15-89   9:57a
RDCOPY2  EXE      82206 09-15-89   2:08p
RDFMT2   EXE      23234 09-15-89   4:36p
D            <DIR>      08-17-90  10:51a
CONFIG   SYS         31 10-26-90  10:40a
IRIGB    COM      12741 01-03-80   7:46p
AUTOEXEC BAT        103 08-17-90  10:57a
IO2      EXE      32422 08-15-90   2:06p
CONFIG   BAK         53 09-27-90   7:48a

  Directory of A:\D

CMD-CD   EXE     141990 06-09-92   7:07a
DV       COM       9505 03-22-89   2:25a
DV       EXE     128386 03-22-89   2:25a
DESQVIEW DVO        240 06-09-92   8:34a
ZIP      BAT         66 03-16-90  12:05p
CONFIG   SYS         31 08-17-90  10:50a
SETUP    DVP        416 03-22-89   2:25a
AUTOEXEC BAT        116 06-11-91   3:44p
D1-PIF   DVP        416 03-15-89  12:35p
DS-PIF   DVP        416 12-22-87   2:01a
MS-PIF   DVP        416 12-22-87   2:01a
SETUPCMD EXE       2722 09-15-89   9:31a
MODE     COM      15440 02-02-88  12:00a
CP-PIF   DVP        416 12-22-87   2:01a
AP-PIF   DVP        416 12-22-87   2:01a
CD-PIF   DVP        416 04-04-90   7:49a
INSTLCHG COM       8592 12-22-87   2:01a
INSTLADD COM      14800 12-22-87   2:01a
DVHERC   COM       1799 12-22-87   2:01a
DVANSI   COM       2003 12-22-87   2:01a
CD-SCRIP DVS       1066 03-19-90  12:32p
LEARN    DVR       9589 03-22-89   2:25a
DESQVIEW DVS       1024 12-27-89  12:41p
ST-PIF   BAK        416 06-09-92   8:33a
MS       COM        646 12-22-87   2:01a
DVSETUP  COM      12487 12-22-87   2:01a
AUTOINST COM       3608 12-22-87   2:01a
CONVSCR  COM       6068 12-22-87   2:01a
IO-PIF   BAK        416 06-19-92   9:00a
DVSETUP  DV         702 04-16-90  11:37a
NUMON    COM         12 01-18-90   9:04a
DESQVIEW DVH      31662 03-22-89   2:25a
GRFCGA   DVR       2222 03-22-89   2:25a
EMM      DVR       7620 03-22-89   2:25a
DOSBUF   DVR       3561 03-22-89   2:25a
SETUP    BAT         23 12-22-87   2:01a
AL-PIF   DVP        416 06-19-92   9:05a
DIALDIR  PRM        154 07-24-89   2:59p
SWAPBNBG DV      144272 03-28-90   8:19a
```

```
CMD-ST   EXE    47732 06-16-92   7:42a
CMD-IO   EXE    34504 06-09-92   8:23a
CMD-AL   EXE    21436 03-12-92  12:57p
ST-PIF   DVP      416 06-09-92   8:34a
CMD      LOG     4607 07-31-91   3:21p
IO-PIF   DVP      416 06-19-92   9:03a
AL-PIF   BAK      416 06-19-92   9:04a
```

APPENDIX J: IMAGES (IMG) COMPUTER BOOT CONFIGURATION

IMG COMPUTER BOOT PROCESS

The IMG computer is a networked workstation that boots up off a network FS. Since the IMG machine depends on the FS to boot, the first step in booting up the IMG platform is to make sure that the FS is up and running. For the bootup procedure for the FS, see appendix N.

Assuming that the FS is up and operational, the following procedure is for starting up the IMG workstation:

- Power up the four monitors connected to the IMG machine
- Power up the IMG machine

BOOT FILES

There are four user configurable files that control the bootup process for the IMG machine:

- CONFIG.SYS
- AUTOEXEC.BAT
- DOLOGIN.BAT
- User 'img' login script file

Note that since the IMG machine boots up off the network FS, there is no "real" disk drive on the system. Instead, a boot ROM is installed on the ethernet card in the system and this card is jumpered to cause the system to execute a program embedded in the ROM on power-up that establishes a connection between the IMG and FS machines. Once this connection is established, a file named BOOTCONF.SYS in the SYS:LOGIN subdirectory of the server is scanned for the name of a boot image file associated with the ID number of the ethernet card. In this particular case, the boot image file name for IMG is IMG_BOOT.SYS. Once found, the system loads this boot image into memory and simulates a floppy diskette loaded in a floppy drive A:. This boot image file is created using a real bootable floppy containing DOS files for the IMG machine and by running the Novell Netware utility DOSGEN on this diskette. The complete procedure to create the "Remote Boot Image" file can be found in the *Novell Netware Version 3.11 Installation Manual.*

The CONFIG.SYS file is executed first on power-up, then execution of the AUTOEXEC.BAT follows. Within the AUTOEXEC.BAT file a call is made to execute DOLOGIN.BAT, which executes commands to login formally to the server as user 'img.' There is no password required to login as 'img,' and as a result, bootup control then passes to the login script for user 'img.'

BOOT FILE LISTINGS

Listed below are the contents of all four files.

CONFIG.SYS

```
files=25
buffers=16
lastdrive=e
device=nnios.sys /m=A000:16;A400:16;A800:16;
device=serial.sys device=vdisk.sys 256 512 64 /E
```

AUTOEXEC.BAT

```
prompt=$t$h$h$h$h$h$h $p$g
cls
ver
COPY COMMAND.COM C:\
SET COMSPEC=C:\COMMAND.COM
copy ipx.com c:\
copy net3.com c:\
copy dologin.bat c:\
c: dologin
```

DOLOGIN.BAT

```
A:ipx
A:net3
f: login img
```

User 'IMG' Login Script

```
MAP INS S5:=SYS:PUBLIC\TMI386\MSDOS\V3.30
COMSPEC=S5:COMMAND.COM
MAP F:=SYS:DV
MAP G:=SYS:IMG
DRIVE F:
EXIT "dv"
```

FILE LIST ON IMG BOOT DISK

DIRECTORY of A:\

COMMAND	COM	25307	03-17-87	12:00p
SERIAL	SYS	10605	01-05-88	3:58p
IPX	COM	27900	04-04-89	12:42p
DOLOGIN	BAT	32	06-21-90	10:49a
NNIOS	SYS	16322	08-18-89	2:08p
NET3	COM	48544	05-08-90	3:37p
VDISK	SYS	3455	03-17-87	12:00p
AUTOEXEC	BAT	159	06-21-90	7:58a
CONFIG	SYS	133	06-21-90	7:25a

Note that this IPX.COM is Novell shell driver for a Western Digital WD8003E ethernet card configured for a port address of 280H and using interrupt 5.

START-UP INSTRUCTIONS FOR IMG PROGRAM

The bootup sequence will take the operator into the DESQview environment and present the DESQview 'Open window' menu. Type 'O' followed by 'II.' This will open the DESQview applications window and the 'II' key sequence will select and load the IMG application program running under DESQview.

APPENDIX K: NAVIGATION (NAV) COMPUTER BOOT CONFIGURATION

NAV COMPUTER BOOT PROCESS

The NAV computer is a networked workstation that boots up off a network FS. Since the NAV machine depends on the FS to boot, the first step in booting up the NAV platform is to make sure that the FS is up and running. For the bootup procedure for the FS, see appendix N.

Assuming that the FS is up and operational, the following is the procedure for starting up the NAV workstation:

- Power up the two monitors connected to the NAV machine
- Power up the NAV machine

BOOT FILES

There are four user configurable files that control the bootup process for the NAV machine:

- CONFIG.SYS
- AUTOEXEC.BAT
- DOLOGIN.BAT
- User 'nav' login script file

Note that since the NAV machine boots up off the network FS, there is no "real" disk drive on the system. Instead, a boot ROM is installed on the ethernet card in the system and this card is jumpered to cause the system to execute a program embedded in the ROM on power-up that establishes a connection between the NAV and FS machines. Once this connection is established, a file named BOOTCONF.SYS in the SYS:LOGIN subdirectory of the server is scanned for the name of a boot image file associated with the ID number of the ethernet card. In this particular case, the boot image file name for NAV is NAV_BOOT.SYS. Once found, the system loads this boot image into memory and simulates a floppy diskette loaded in a floppy drive A:. This boot image file is created using a real bootable floppy containing DOS files for the NAV machine and by running the Novell Netware utility DOSGEN on this diskette. The complete procedure to create the "Remote Boot Image" file can be found in the *Novell Netware Version 3.11 Installation Manual.*

The CONFIG.SYS file is executed first on power-up, then execution of the AUTOEXEC.BAT follows. Within the AUTOEXEC.BAT file a call is made to execute DOLOGIN.BAT, which executes commands to login formally to the server as user 'nav.' There is no password required to login as 'nav,' and as a result, bootup control then passes to the login script for user 'nav.'

BOOT FILE LISTINGS

Listed below are the contents of all four files.

CONFIG.SYS

```
files=30
buffers=16
lastdrive=e
device=vdisk.sys 256 512 64 /E
```

AUTOEXEC.BAT

```
prompt=$t$h$h$h$h$h$h $p$g
SET COMSPEC=C:\COMMAND.COM
copy ipx.com c:\
copy net3.com c:\
copy dologin.bat c:\
c:
c: dologin
```

DOLOGIN.BAT

```
A:ipx
A:net3
f: login nav
```

User 'NAV' Login Script

```
MAP INS S5:=SYS:PUBLIC\TMI386\MSDOS\V3.30
COMSPEC=S5:COMMAND.COM
MAP F:=SYS:DV
MAP G:=SYS:NAV
DRIVE F:
EXIT "run"
```

FILE LIST ON NAV BOOT DISK

```
  DIRECTORY of   A:\

COMMAND    COM            47845    04-09-91     5:00a
PACKET          <DIR>              11-05-91     2:58p
DOS             <DIR>              11-05-91     3:05p
12KSPOT    MIS             2578    03-06-90    10:42a
AUTOEXEC   BAK              259    11-19-91     2:43p
AUTOEXEC   BAT               59    11-20-91     1:32p
```

Note that this IPX.COM is Novell shell driver for a WD8003E ethernet card configured for a port address of 280H and using interrupt 3.

DIRECTORY of A:\DOS

HIMEM	SYS	6261	10-06-91	12:00a
FORMAT	COM	32911	04-09-91	5:00a
EMM386	SYS	87776	10-06-88	12:00a
FDISK	EXE	57224	04-09-91	5:00a
SYS	COM	13440	04-09-91	5:00a
XCOPY	EXE	15804	04-09-91	5:00a
PRINT	EXE	15656	04-09-91	5:00a
CHKDSK	EXE	16200	04-09-91	5:00a
DISKCOPY	COM	11793	04-09-91	5:00a
SORT	EXE	6938	04-09-91	5:00a
COMMAND	COM	47845	04-09-91	5:00a

DIRECTORY of A:\PACKET

COMMAND	COM	25307	03-17-87	12:00p
NET3	COM	41038	03-27-89	9:30a
OSBORNE	BAT	27	11-05-91	3:27p
3C502	COM	4509	03-28-91	4:33p
TEST	.BAT	62	08-13-90	11:07a
TEST	BAK	62	08-13-90	8:08a
OSBORNE	BAK	59	08-09-89	8:14a
IPX	COM	25974	03-27-89	9:30a
NETX	COM	51201	07-31-91	10:47a
XMSNETX	EXE	60307	07-31-91	10:52a
XMSNET5	EXE	59264	03-07-91	4:57a
EMSNETX	EXE	63838	07-31-91	10:49a
EMSNET5	EXE	62251	03-07-91	4:50a
EMSNET4	EXE	61578	02-06-91	4:55a
EMSNET3	EXE	61118	02-06-91	5:01a
NLOGIN	EXE	26680	03-21-91	8:35a
IPX304	COM	37610	08-08-91	8:50a
WD8003E	COM	6535	03-28-91	4:34a
IPX	DOC	35	11-05-91	3:04a

START-UP INSTRUCTIONS FOR NAV PROGRAM

Just turn power on as described above. The bootup sequence will take the operator into the SEATRAC program.

APPENDIX L: AUSS INTEGRATED NAVIGATION SYSTEM (AINS) COMPUTER BOOT CONFIGURATION

AINS COMPUTER BOOT PROCESS

The AINS computer is a networked workstation that boots up off a 3-and-1/2-inch floppy disk. Although the AINS machine does not boot off the FS, it still tries to log on to the network as part of its start-up process. Therefore, the first step is to verify that the FS is operational. The procedure for starting up the FS is detailed in appendix N.

Assuming that the FS is up and operational, the following is the procedure for starting up the AINS workstation:

- Power up the two monitors connected to the AINS machine
- Power up the AINS machine

BOOT FILES

There are four user configurable files that control the bootup process for the AINS machine:

- CONFIG.SYS
- AUTOEXEC.BAT
- OSBORNE.BAT
- Login script file for user 'osborne'

The CONFIG.SYS file is executed first on power-up, then execution of the AUTOEXEC.BAT follows. Within the AUTOEXEC.BAT file a call is made to execute OSBORNE.BAT, which executes commands to login formally to the server as user 'OSBORNE.' There is no password required to login as 'OSBORNE,' and as a result, bootup control then passes to the login script for user 'OSBORNE.'

BOOT FILE LISTINGS

Listed below are the contents of all four files.

CONFIG.SYS

```
device=a:\nnios.sys
files=25
buffers=25
```

AUTOEXEC.BAT

```
echo off
path=f:\lbsprog;f:\c600\bin;f:\c600\source
prompt $T$H$H$H$H$H$H $P$G
cd \packet
osborne
```

OSBORNE.BAT

```
ipx
netx
nlogin osborne
f:
```

Login Script For 'OSBORNE'

```
MAP INS S5:=SYS:PUBLIC\TMI386\MSDOS\V5.00
COMSPEC = S5:COMMAND.COM
MAP F:=SYS:LBSPROG
MAP G:=SYS:LOG
```

FILE LIST ON AINS BOOT DISK

```
 Directory of A:\

COMMAND  COM     47845 04-09-91    5:00a
PACKET       <DIR>     11-05-91    2:58p
DOS          <DIR>     11-05-91    3:05p
12KSPOT  MIS      2578 03-06-90   10:42a
AUTOEXEC BAK       269 11-19-91    2:43p
LBSPROG      <DIR>     02-11-92    3:22p
CONFIG   SYS        43 05-27-92   11:13a
NNIOS    SYS     16130 04-01-89   12:00p
QE       EXE     48096 11-08-88    8:44a
AUTOEXEC BAT       103 06-22-92    9:01a

 Directory of A:\PACKET

OSBORNE  BAT        31 04-24-92    1:23p
NET3     COM     41038 03-27-89    9:30a
MISSION  ZIP     32910 06-02-92    7:13a
3C501    COM      4509 03-28-91    4:33p
TEST     BAT        62 08-13-90   11:07a
TEST     BAK        62 08-13-90    8:08a
OSBORNE  BAK        59 08-09-90    8:14a
IPX      COM     25974 03-27-89    9:30a
NETX     COM     51201 07-31-91   10:47a
XMSNETX  EXE     60307 07-31-91   10:52a
XMSNET5  EXE     59264 03-07-91    4:57p
EMSNETX  EXE     63838 07-31-91   10:49a
EMSNET5  EXE     62251 03-07-91    4:50p
EMSNET4  EXE     61578 02-06-91    4:55p
EMSNET3  EXE     61118 02-06-91    5:01p
NLOGIN   EXE     26680 03-21-91    8:35a
IPX304   COM     37610 08-08-91    8:50a
WD8003E  COM      6535 03-28-91    4:34p
IPX      DOC        35 11-05-91    3:04p
```

NOTE: IPXCOM is a Novell shell driver for a 3COM 3C501 ethernet card with a port address of 300H and using interrupt 5. This should be upgraded to use a packet driver interface using the 3C501.COM, IPX304.COM, and the NETX.COM files. See appendix M for an example of how this was implemented for machine LOG.

Directory of A:\DOS

```
HIMEM    SYS      6261 10-06-88  12:00a
FORMAT   COM     32911 04-09-91   5:00a
EMM386   SYS     87776 10-06-88  12:00a
FDISK    EXE     57224 04-09-91   5:00a
SYS      COM     13440 04-09-91   5:00a
XCOPY    EXE     15804 04-09-91   5:00a
PRINT    EXE     15656 04-09-91   5:00a
CHKDSK   EXE     16200 04-09-91   5:00a
DISKCOPY COM     11793 04-09-91   5:00a
SORT     EXE      6938 04-09-91   5:00a
COMMAND  COM     47845 04-09-91   5:00a
```

Directory of A:\LBSPROG

```
NS11DRIV EXE     18158 02-26-91   3:37p
```

START-UP INSTRUCTIONS FOR AINS PROGRAM

Just type 'AINS.'

APPENDIX M: LOGGER (LOG) COMPUTER BOOT CONFIGURATION

LOG COMPUTER BOOT PROCESS

The LOG computer is a networked workstation that boots up off a local hard disk. Since the LOG machine expects to see FS during the start-up process, the first step in booting up the LOG platform is to make sure that the FS is running. For the boot-up procedure for the FS, see appendix N.

Assuming that the FS is up and operational, the following is the procedure for starting up the LOG workstation:

- Power up the two monitors connected to the LOG machine
- Power up the LOG machine

BOOT FILES

There are three user configurable files that control the bootup process for the LOG machine:

- CONFIG.SYS
- AUTOEXEC.BAT
- User 'LOG' login script file

Note that since the LOG machine boots off a local hard disk, this workstation is the easiest to re-configure. As such, it serves as a model of the preferred configuration for surface computers during system development phases. The NAV and IMG machines use remote booting off of the FS so as to not require a local disk drive. For the delivery configuration this would be fine, but during development any changes to the bootup procedures require regenerating the boot image files on the server. On the CMD machine, the ROM disk bootup method is very reliable, but it makes the implementation of changes to the software time consuming. For the delivery configuration, all machines except the CMD should have a local hard disk as the primary boot method with a floppy disk as a backup. For the CMD machine, primary boot should be via the ROM disk and the backup should be via a hard disk.

The CONFIG.SYS file is executed first on power-up, then execution of the AUTOEXEC.BAT follows. There is no password required to login as 'LOG,' and as a result, bootup control then passes to the login script for user 'LOG.'

BOOT FILE LISTINGS

Listed below are the contents of all three files.

CONFIG.SYS

```
DEVICE=C:\QEMM\QEMM386.SYS R:2 X=D000-D7FF RAM ST:M
files= 30
SHELL=C:\COMMAND.COM /P/E:500
STACKS=0,0
break=on
BUFFERS=1
DEVICE=c:\qemm\loadhi.sys /r:2 C:\nnios\nnios.sys   /m=a000:16;
DEVICE=c:\qemm\loadhi.sys /r:1 c:\mouse.sys /2
```

AUTOEXEC.BAT

```
c:\qemm\loadhi /r:1 c:\qemm\buffers=32
prompt=$t$h$h$h$h$h$h $p$g
PATH=C:\;C:\DOS5;C:\QE;
chkdsk c:
cls
echo off
cls
ver
echo on
cd\packet
c:\qemm\loadhi /r:2 wd8003e -n 0x62 0x05 0x280 0xD000
c:\qemm\loadhi /r:2 ipx304
c:\qemm\loadhi /r:2 netx
cd\dv
f:
login log
```

User 'LOG' LOGIN SCRIPT

```
MAP INS S5:=SYS:PUBLIC\TMI386\MSDOS\V5.00
MAP F:=SYS:LOG
EXIT "dv"
```

FILE LIST ON LOG BOOT DISK

```
 Directory of C:\

COMMAND  COM     47845 04-09-91   5:00a
MOUSE    SYS     28547 12-06-90   5:01a
MW50         <DIR>     02-24-92  10:49a
PACKET       <DIR>     11-21-91   8:17a
QEMM         <DIR>     11-21-91   8:18a
NNIOS        <DIR>     11-21-91   8:20a
DOS5         <DIR>     11-21-91   8:22a
DV           <DIR>     11-21-91   8:23a
QE           <DIR>     11-21-91   8:32a
OPT3     BAT       616 04-17-92  12:16p
NET          <DIR>     01-28-92   3:27p
MOUSE    COM     28274 12-06-90   5:01a
AUTOEXEC QDK       186 04-17-92  12:09p
```

```
OPTAUTO  BAT        187 04-17-92  12:14p
OPT1TEST BAT         68 04-17-92  12:14p
AUTOEXEC OLD        286 02-26-92  10:37a
CONFIG   QDK        191 04-17-92  12:15p
CONFIG   OLD        306 02-26-92  10:37a
CONFIG   SYS        242 04-17-92  12:16p
ARCSERVE     <DIR>      05-07-92   2:18p
AUTOEXEC BAT        280 04-17-92  12:27p
       21 file(s)     107028 bytes

 Directory of C:\QEMM

RSTRCFG  SYS        319 11-13-91   6:02a
HINTDATA OPT         18 04-17-92  12:17p
OPTIMIZE INF       1591 04-17-92  12:16p
MFT      EXE      76266 11-13-91   6:02a
VIDRAM   COM       7587 11-13-91   6:02a
QEMMREG  COM        414 11-13-91   6:02a
BUFFERS  COM       3170 11-13-91   6:02a
EMS      COM       6109 11-13-91   6:02a
FCBS     COM       3024 11-13-91   6:02a
FILES    COM       2808 11-13-91   6:02a
LASTDRIV COM       2924 11-13-91   6:02a
LOADHI   COM      21629 11-13-91   6:02a
LOGOPT   COM       2833 09-07-90   5:11a
OPTIMIZE COM      71493 11-13-91   6:02a
QEMM     COM       7512 11-13-91   6:02a
EMS      SYS       6291 11-13-91   6:02a
EMS2EXT  SYS       3097 11-13-91   6:02a
LOADHI   SYS      18652 11-13-91   6:02a
MCA      ADL      64226 11-13-91   6:02a
MFT      HLP      19808 11-13-91   6:02a
MF-PIF   DVP        416 11-13-91   6:02a
QEMM386  SYS      89831 11-13-91   6:02a
WINHIRAM VXD      10838 11-13-91   6:02a
TECHSUP  BAT        368 11-13-91   6:02a
READQ    ME       27786 08-23-91   6:00a
QWINFIX  COM       1964 11-13-91   6:02a
TESTBIOS COM       2040 11-13-91   6:02a
HOOKROM  SYS       1369 11-13-91   6:02a
WINSTLTH VXD      10841 11-13-91   6:02a
4DOS     CMD        587 11-13-91   6:02a
OPTIMIZE DAT          5 04-17-92  12:15p
LOADHI   OPT        547 04-17-92  12:16p
UNOPT    BAT        298 04-17-92  12:16p
READ     ME       30933 11-13-91   6:02a
STEALTH  LOG         19 04-17-92  12:15p
       37 file(s)     497613 bytes
```

```
 Directory of C:\PACKET

NETX      COM      51201 07-31-91  10:47a
WD8003E   COM       6535 03-28-91   4:34p
IPX304    COM      37610 08-08-91   8:50a
KONO      BAT         46 11-21-91   7:15a
NLOGIN    EXE      26680 03-21-91   8:35a
PSU       TXT      14560 06-08-88   5:40p
PSU       EXE      13853 05-30-89  11:02a
PM        COM      25791 04-22-86   8:52p
HIMEM     SYS      11304 05-01-90   3:00a
NET3      COM      49198 02-06-91   4:44p
NET4      COM      49625 02-06-91   4:39p
EMSNET4   EXE      61578 02-06-91   4:55p
EMSNET3   EXE      61118 02-06-91   5:01p
XMSNET3   EXE      58219 02-06-91   5:16p
XMSNET4   EXE      58635 02-06-91   5:10p
IPX302A   COM      29114 03-21-91   7:22a
3C501     COM       4509 03-28-91   4:33p
NET5      COM      50260 04-09-91   5:00a
EMSNET5   EXE      62251 03-07-91   4:50p
EMSNETX   EXE      63838 07-31-91  10:49a
XMSNETX   EXE      60307 07-31-91  10:52a
       23 file(s)     796232 bytes

 Directory of C:\DV

SAMPLE    PLB       1984 09-07-90   2:31a
LOGGER    PLB       1668 03-18-92  11:10a
AOBOAODL           12288 05-20-92   5:26p
AMBHBMED           16384 05-31-92   8:22p
        6 file(s)      32324 bytes
```

START-UP INSTRUCTIONS FOR LOG PROGRAM

Note that during bootup, the C: drive must be set to the \DV subdirectory prior to logging into the FS. Once the login script has terminated executing its command list and the user is left on drive F:, the following keys must be typed to start execution:

DV<CR>OLO

"DV" will start up DESQview on the drive/subdirectory F:\LOG, "O" will execute the Open window command from the DESQview menu, and finally "LO" will invoke the DESQview LOGGER application program. It is intended that once the LOGGER application is completed, the start-up sequence would be further automated such that the operator would not be required to type any keys.

APPENDIX N: FILE SERVER (FS) COMPUTER BOOT CONFIGURATION

FS COMPUTER BOOT PROCESS

The FS computer is a Novell-Netware-dedicated file server running Netware 3.11. The FS's SCSI hard disk, floppy disk, and DAT tape drive are housed in an external cabinet next to the main 7552 enclosure. This external enclosure should have power switched on, indicated by a green LED indicator on the front of the box. The power is on the rear panel of the enclosure. There is only a single monitor connected to this machine and it should also be turned on prior to powering up the computer.

BOOT FILES

There are three user configurable files that control the bootup process for the FS machine:

- AUTOEXEC.BAT
- STARTUP.NCF
- AUTOEXEC.NCF

Upon power-up, the file 'AUTOEXEC.BAT' is executed. Within the AUTOEXEC.BAT file a call is made to execute SERVER.EXE. The program SERVER loads the Novell Netware networking software. As part of the Netware start-up process, two script files are executed in the following order: STARTUP.NCF residing on the DOS boot disk and AUTOEXEC.NCF located in the SYS:SYSTEM directory of the Netware volume. STARTUP.NCF loads the SCSI disk controller driver that allows Netware to mount the Netware partition of the disk drive. Once that is loaded, AUTOEXEC.NCF takes over and loads the rest of the necessary Netware Loadable Modules (NLMs) for full operation of the server.

BOOT FILE LISTINGS

Listed below are the contents of all three files.

AUTOEXEC.BAT

```
prompt $p$g
path=c:\dos;c:\novell
pause "hit key to continue"
cd \server
server
```

STARTUP.NCF

```
load ASPITRAN
load AHA1540 port=330
set minimum packet receive buffers = 100
```

AUTOEXEC.NCF

```
file server name AUSS1
ipx internal net 2386941
mount all
load sys:system\wdplus\SMCPLUSS port=280 mem=D0000 int=3
    frame=ETHERNET_802.3 name=STD
bind IPX to SMCPLUSS net=1
load clib
load remote aussii
load rspx
load ipxs
load spxs
load patch311
```

FILE LIST ON FS BOOT DISK

```
 Directory of C:\

COMMAND  COM     47845 04-09-91   5:00a
NOVELL       <DIR>     09-13-91   8:52a
SERVER       <DIR>     09-13-91   9:23a
SOFTSET      <DIR>     12-09-91   8:59a
AUTOEXEC BAT        85 05-08-92  10:17a

 Directory of C:\SERVER

DISKSET  NLM     72530 02-14-91   9:02a
V_MAC    NLM      4914 12-10-90   7:06a
NE2-32   LAN     11639 01-21-91   6:38a
CLIB     NLM    232842 02-14-91   4:26p
3C503    LAN     11856 01-21-91   6:39a
PS2MFM   DSK      8759 01-31-91  11:01a
INSTALL  NLM    160613 02-20-91  11:59a
NE2000   LAN     11636 01-31-91   4:35p
OS2      NAM     16703 01-18-91  10:49p
NE2      LAN     11589 01-21-91   6:24a
PS2ESDI  DSK      7564 02-12-91   8:39p
DCB      DSK     18613 02-12-91   9:08p
ISADISK  DSK     10415 02-15-91  11:31a
NE3200   LAN     23043 02-06-91   9:12a
NE1000   LAN     11336 01-31-91   4:43p
FIRMLOAD COM      1628 01-04-91   8:57a
3C505    LAN     21677 01-17-91  11:49a
V_OS2    NLM      7642 12-10-90   7:10a
PCN2L    LAN      9868 02-07-91   3:20p
TOKEN    LAN      9959 02-11-91   1:44p
PS2SCSI  DSK      9577 02-12-91   9:03p
ETHERRPL NLM     14415 01-21-91  11:12a
MAC      NAM     14977 11-13-90   4:48p
TOKENRPL NLM     16823 01-21-91  11:29a
NMAGENT  NLM     33929 02-06-91   9:39a
TRXNET   LAN      8955 01-07-91   2:59p
TOKENDMA LAN      8172 02-05-91  12:13p
MATHLIB  NLM     12459 02-14-91   3:07p
```

```
FILEDATA DAT        2460 02-19-91    7:33p
MATHLIBC NLM       16822 02-14-91    3:06p
ROUTE    NLM        4508 08-10-90    8:55a
UINSTALL NLM        5038 12-14-90    2:40p
VREPAIR  NLM       86394 02-07-91    7:13p
3C523    LAN       12225 02-06-91    4:42p
SERVER   EXE      879783 02-27-91    9:56a
FILES    DAT       53522 02-27-91    2:18p
README   311       35408 02-20-91    4:59p
NUT      NLM       42484 12-20-90    8:18a
STARTUP  NCF          68 05-07-92    2:30p
WDPLUSSV LAN       15899 04-24-91   11:08a
ADAPTEC  NLM       63022 02-13-91    5:07p
WDPLUS        <DIR>      10-25-91    1:27p
RSPX     NLM       17023 02-09-91    6:35a
IPXS     NLM       6676  02-12-91   10:35a
SPXS     NLM       6194  02-14-91   11:55A
XRCISE   EXE       49943 07-18-89    2:14a
AHA1540  DSK       17641 03-27-91   12:00p
ASPITRAN DSK       1586  07-31-90   12:00p
STARTUP  NCP       82    07-31-90    1:17p

 Directory of C:\SERVER\WDPLUS SMCPLUSS LAN 16479 03-19-92
```

START-UP INSTRUCTIONS FOR FS PROGRAM

After powering up equipment, the auto boot programs will pause, asking the operator to hit a key to continue. If a normal bootup is required, hit any key. If the operator desires to interrupt the sequence and stay in DOS, hit Ctrl-C. Normal bootup of the Novell Netware operating system will leave the screen displaying a single ':' prompt.

APPENDIX O: SPARES LIST

SPARES LIST

The following spares list is based on the AUSS system as configured for June 1992, and as such, it does not represent sparing for the planned delivery configuration. The list represents what is recommended as spares and not necessarily what is on hand in AUSS inventory. The list of items will be divided into two parts: a multiple item list—items used in more than one location—and a single item list—items used only once in the system.

MULTIPLE ITEM LIST

ITEM	DESCRIPTION
IBM 7552 chassis	Enclosure, power supply for six main surface computers
IBM 7552 card shroud	Individual card enclosure for 7552 systems
IBM 7552 feature adapter	Individual card adapter to interface AT cards to 7552
Generic 101 key PC keyboard	Keyboard for each of the 7552 systems
NEC multisync II monitor	Display monitor for NAV and AINS computers
NEC multisync 3D monitor	Display monitor for CMD, IMG, LOG, and FS computers
Vopex-2V video repeater	Video repeater for display switching of Pepper and VGA
Vopex-2E video repeater	Video repeater for NAV and AINS EGA displays
386-20 MHz CPU board	Texas Microsystems model B386/20 CPU card
486-33 MHz CPU board	Diversified Technology CAT-1000 CPU card
Generic mono display card	Generic AT mono display adapter (MDA)
VGA display card	ATI Basic VGA display card
Number Nine Graphics card	Number Nine Pepper SGT Plus display card
Ethernet card	Western Digital WD8003EBT ethernet card
AT multiple I/O card	Kouwell KW-524H dual serial + parallel port card
RG-58 coax cable	Used for ethernet connections for network
4 position video switch box	DB-15 switch box for video switching from Vopex
RS-232 null modem cable	Used for interconnecting computers
EGA to multisync 12' cable	Used for NAV and AINS displays
VGA-M to VGA-F 6' cable	Used for Pepper and VGA displays
1.44-Mbyte 3-1/2" floppy drive	Generic 1.44-Mbyte floppy drive

SINGLE ITEM LIST

ITEM	DESCRIPTION
ROMDISK card	Industrial Computer Source model PCE/2
Acoustic link interface card	Custom NRaD design
Time code reader card	Bancomm, division of Datum, model PC03XT
IDE disk + I/O card	Super IDE I/O card model PT604
80 Mbyte IDE hard disk	Maxtor model 7080AT
EGA card	Generic EGA card used in NAV computer
Graphics display card	Control Systems Artist II display card for NAV
Disk controller card	Western Digital WD1003-WA2 MFM controller card
Multiport serial card	DigiCHANNEL COM/Xi 8 port serial card for NAV
EGA card	ATI Wonder 800 EGA card for AINS
IDE disk + I/O card	Identity model IDMIO controller card
Synchro to digital converter	ILC data device model SDC-36015 card for AINS
SCSI controller card	Adaptec 1542B SCSI controller card for FS
SCSI hard disk	Maxtor LXT213S SCSI hard disk for FS
DAT drive	Archive 4520NT DAT drive for FS
Ethernet card	Western Digital WD8013EBT ethernet card for FS
50 conductor cable	Adaptec 1542B to external drive enclosure
50 conductor ribbon cable	CMD to acoustic link computer

ITEMS NOT INCLUDED

The YEM Model CVS-910 scan converter and the Super VHS tape recorder were not included for sparing because these items were not mission critical and would be repaired rather than replaced.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1992	Final

4. TITLE AND SUBTITLE

SURFACE COMPUTER SYSTEM ARCHITECTURE FOR THE ADVANCED UNMANNED SEARCH SYSTEM (AUSS)

5. FUNDING NUMBERS

PE: 0603713N
Proj.: S0397

6. AUTHOR(S)

M. E. Kono

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Ocean Engineering Division
Naval Command, Control and Ocean Surveillance Center (NCCOSC)
RDT&E Division (NRaD)
San Diego, CA 92152–5000

8. PERFORMING ORGANIZATION REPORT NUMBER

NRaD TR 1538

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Assistant Secretary of the Navy for Research and Development
Code PMO-403
Washington, DC 20350

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The redesigned Advanced Unmanned Search System (AUSS) surface control van has been based on the IBM PC/AT, a well-known, commercially available computer. Its installed market base has made hardware add-on components widely available. The market dominance of Microsoft's DOS has standardized the operating system software. The surface control van computers' functions have been split among several machines to provide for some modularity of the software design and accommodate possible increased processing demands. Operating system software and hardware architecture for these six 7552-based PCs are detailed.

The author recommends redefining the AUSS surface computer to be an X Window Platform, an architecture that can provide a virtual display environment. The virtual display environment would protect the software development investment from obsolescence due to hardware technology advances, reduce sparing requirements, and simplify the control van wiring.

Appendices describe board jumpers, switch settings, and boot configurations for the six 7552-based PCs.

14. SUBJECT TERMS

IBM 7552 PC
X Window System
MULTIBUS I and II

15. NUMBER OF PAGES

89

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAME AS REPORT

21a. NAME OF RESPONSIBLE INDIVIDUAL	21b. TELEPHONE *(Include Area Code)*	21c. OFFICE SYMBOL
M. E. Kono	(619) 553–1884	Code 941