DEVELOPMENT OF AN AUTOMATED MICROCOMPUTER
KNOWLEDGE-BASED INTEGRATED CON. (U) NAVAL POSTGRADUATE
SCHOOL MONTEREY CA R L BEARD MAR 86

F/G 9/2
THESIS

DEVELOPMENT OF AN AUTOMATED MICRO-COMPUTER KNOWLEDGE-BASED INTEGRATED CONFIGURATION MANAGEMENT SYSTEM FOR THE STOCK POINT LOGISTICS INTEGRATED COMMUNICATIONS ENVIRONMENT (SPLICE) PROJECT MANAGEMENT STAFF

by

Robert Lee Beard III

March 1986

Thesis Advisor: Norman R. Lyons
Co-Advisor: Barry A. Frow

Approved for public release; distribution is unlimited
This thesis documents the development of a micro-computer knowledge-based integrated configuration management system for use by Naval Supply Systems Command (NAVSUP) Stock Point Logistics Integrated Communications Environment (SPLICE) Project Management Staff. A myriad of configuration heuristics associated with the configuration of a SPLICE site are identified. It also provides SPLICE project staff personnel a more accurate, reliable and efficient method of performing the configuration process and managing the overall project. The development of this integrated configuration management system employs both a prototype and software engineering methodology. The integrated configuration management system will be developed using custom generated software and the logical integration of several off-the-shelf commercial software packages.
Approved for public release; distribution is unlimited.

Development of an Automated Micro-computer Knowledge-based Integrated Configuration Management System for the Stock Point Logistics Integrated Communications Environment (SPLICE) Project Management Staff

by

Robert L. Beard III
Lieutenant Commander, Supply Corps, United States Navy
BE CS, University of New Mexico, 1975

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
March 1986

Author: Robert L. Beard III
Approved by: Norman R. Wyons, Thesis Advisor
Barry A. Frew, Co-Thesis Advisor
Willis R. Greer, Jr., Chairman, Department of Administrative Sciences
Kneale T. Marshall, Dean of Information and Policy Sciences
ABSTRACT

This thesis documents the development of a micro-computer knowledge-based integrated configuration management system for use by Naval Supply Systems Command (NAVSUP) Stock Point Logistics Integrated Communications Environment (SPLICE) Project Staff. A myriad of configuration heuristics associated with the configuration of a SPLICE site are identified. It also provides SPLICE project staff personnel a more accurate, reliable and efficient method of performing the configuration process and managing the overall project.

The development of this integrated configuration management system employs both a prototype and software engineering methodology. The integrated configuration management system will be developed using custom generated software and the logical integration of several off-the-shelf commercial software packages.
THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.
## TABLE OF CONTENTS

I. INTRODUCTION ................................................. 7  
   A. PURPOSE ................................................. 7  
   B. BACKGROUND ............................................. 8  
   C. SCOPE ................................................... 9  

II. CONFIGURATION RULES ......................................... 12  
   A. BASIC CONFIGURATION RULES ............................... 12  
   B. UNIQUE CONFIGURATION RULES ............................. 16  
      1. Hardware ............................................. 16  
      2. Software ............................................. 18  
      3. Manuals and Documentation ........................... 19  
      4. Training ............................................. 20  
      5. Maintenance ......................................... 20  
      6. Other ................................................ 21  
      7. Discount and Escalation Rates ...................... 21  

III. METHODOLOGY USED TO DEVELOP THE SYSTEM ................. 23  
   A. PROTOTYPE .............................................. 25  
   B. SOFTWARE ENGINEERING METHODOLOGY ................... 27  
      1. Planning ............................................. 28  
      2. Development ......................................... 32  
      3. Maintenance ......................................... 34  
   C. SUMMARY ................................................. 37  

IV. SYSTEM EXECUTION DIALOGUE .................................. 39  
   A. SYSTEM INITIATION ...................................... 40  
   B. CONFIGURE A SITE ....................................... 40
C. PERFORM FINANCIAL ANALYSIS ON SITE DATA -------- 41

D. INTERACT WITH THE CONFIGURATION MANAGEMENT AND REPORT GENERATION SUB-SYSTEM -------------- 42
   1. Load New Delivery Order Data ----------------- 42
   2. Load Serial Number and Manual Data --------- 43
   3. Generate a Maintenance Delivery Order ------ 45
   4. Generate a Report --------------------------- 46

E. REVIEW THE ON-LINE USER'S MANUAL --------------- 46

F. TERMINATE SYSTEM EXECUTION ------------------- 47

V. COST BENEFIT AND EFFECTIVENESS ------------------ 48

LIST OF REFERENCES --------------------------------- 53

APPENDIX A: USER'S MANUAL -------------------------- 54

APPENDIX B: MAINTENANCE MANUAL --------------------- 138

BIBLIOGRAPHY -------------------------------------- 423

INITIAL DISTRIBUTION LIST -------------------------- 424
I. INTRODUCTION

A. PURPOSE

The Naval Supply Systems Command (NAVSUP) Stock Point Logistics Integrated Communications Environment (SPLICE) Project Manager is tasked with the responsibility to oversee, direct and review all aspects of the SPLICE project. More specific responsibilities of the SPLICE Project Manager include:

1. ADP equipment acquisition
2. ADP software development
3. Coordination of installations and implementations with field activities

In order to perform the latter of the above responsibilities, the Project Manager must maintain a complete history of all configuration components and component changes. This requirement applies to each component of hardware, software and documentation for the complete fifteen year life cycle of the project. [Ref. 1]

This thesis is designed to provide the Project Manager the capability to perform these functions in an automated manner. A micro-computer knowledge-based integrated configuration management system is seen as the means to accomplish the task. To aid in the development of such a system and reduce development time and difficulty,
functional off-the-shelf commercial packages, where feasible, were used. The system was also designed as a user-friendly interactive system.

B. BACKGROUND

In 1977, NAVSUP conceived and developed the SPLICE project to accomplish the following goals:

1. Provide state-of-the-art local and long haul telecommunications capabilities to sixty-two NAVSUP Stock Points

2. Provide interactive and distributed automated data processing (ADP) capabilities to SPLICE sites

3. Provide capacity relief to aging Burroughs hosts at the Stock Points

4. Standardize and upgrade, via mass replacement, the myriad of minicomputers existing at Stock Points

To achieve these goals, NAVSUP initiated a competitive solicitation for "fault-tolerant" hardware and software. The solicitation was completed in November 1983 and the contract was awarded to Federal Data Corporation (FDC). FDC proposed TANDEM hardware and software to meet most of the solicitation processing and local communication requirements. Network System Corporation hardware and software were proposed to meet the local inter-host communication requirements.

Shortly after the SPLICE contract award, hardware and software components had to be ordered. NAVSUP faced a dilemma. Only a few SPLICE personnel had worked closely with the SPLICE acquisition benchmark and negotiations.
These few people were the only personnel that had sufficient knowledge of the system to configure and generate delivery orders. These personnel developed initial orders by hand to meet the immediate need. Numerous minor errors were encountered with initial orders. FDC corrected the orders to the government and received additional compensation for their efforts.

This manual configuration process was later automated using a software product called SUPERCALC2. It has subsequently transitioned to LOTUS 1-2-3. The basic method of developing these orders remained virtually manual. These few SPLICE personnel, with FDC assistance, developed a series of "rules of thumb" used to configure individual site systems. Many of the original SPLICE group have moved on, taking their knowledge of the system with them.

C. SCOPE

A knowledge-based integrated configuration management software system designed to run on a micro-computer was proposed by a former Fleet Material Support Office¹ (FMSO)

¹FMSO is the Central Design Agency for all NAVSUP software development projects. As such, FMSO is responsible for the project development of the SPLICE project under the guidance and direction of the Systems Commander Project Manager, NAVSUP.
SPLICE project officer\(^2\) to codify these "rules of thumb."

The proposed integrated configuration management system will provide NAVSUP with the capability to develop and maintain SPLICE configurations and delivery orders and to perform configuration management for the overall project. The proposed integrated system will be composed of three software modules designed to:

1. Configure initial SPLICE site systems by answering a series of configuration related questions

2. Restructure the system configure output file into a format compatible for financial and "what-if" analysis

3. Restructure the financial module output file into a format compatible for entry into a data base management system

4. Generate a series of configuration management reports to:
   a. obtain an overall project report
   b. obtain a report for a particular site
   c. obtain a report for a delivery order issued on a particular date

5. Generate a maintenance delivery order for a specific SPLICE site

6. Generate a set of mailing labels for all designated SPLICE sites

---

\(^2\)Lieutenant Commander Edward J. CASE, Supply Corps, United States Navy served as SPLICE project officer from September 1981 to August 1984. LCDR CASE was enrolled as a student at the Naval Postgraduate School from October 1984 to March 1986. Much of the research and development of the micro-computer knowledge-based integrated configuration management system is attributed to the prior knowledge, experience and efforts of LCDR CASE.
Development of the micro-computer knowledge-based integrated configuration management system and successful implementation of the configuration heuristics will provide the NAVSUP SPLICE project manager with the capability to perform all assigned configuration management tasks.
II. CONFIGURATION RULES

The success of the knowledge-based integrated configuration management system is largely dependent upon the accurate implementation of the numerous heuristics involved in the configuration of SPLICE site components. Heuristics which must be considered during the configuration process fall into two categories:

1. Basic configuration rules which apply to all contract line items under consideration
2. Specific configuration rules which apply only to selective contract line items

A breakdown and discussion of these two categories of heuristics is provided below.

A. BASIC CONFIGURATION RULES

A TANDEM processing system consists of a mainframe and its free standing peripherals. A small standard mainframe normally includes two cabinets:

1. Processor (CPU) cabinet
2. Tape cabinet

The processor cabinet houses the processing units (CPUs) and associated power supplies. The tape cabinet houses a magnetic tape unit, Diagnostic Link control panel, I/O patch panels, battery pack or I/O power supply modules. The I/O patch panels provide attachment points for the signal cables.
of various peripherals (ex: CRT terminals, line printers, large capacity disks, etc.). Patch panels are connected to the device controllers residing in the system cabinets through internal cabling.

Additional cabinets (ex: processor, tape, patch panel and expansion) may be added as necessary. Patch panel cabinets provide space for additional patch panels when tape cabinet capacity is inadequate. Generally, mainframe cabinets are fastened together side-by-side to form a single unit.

When two processor cabinets are used in a system and both cabinets contain I/O controllers, additional space for I/O only power supplies may be required. Additional I/O only power supplies may be housed in system expansion cabinets.

System expansion cabinets are required for systems with three or more processor cabinets (or with two processor cabinets connected as noted above). I/O only cabinets must be ordered when system composition reaches four system cabinets. I/O only cabinets may also be necessary to accommodate increased I/O device loads.

Twenty-four I/O slots (four identical backplane assemblies each containing six board slots) are available in a NonStop TXP processor cabinet. The placement of controller boards may result in the need to order additional system or I/O expansion cabinets.
Include one Operations and Service Processor (OSP) with each system.

Every processing unit is supplied with a standard power supply module. The power supply provides several DC voltage levels for use by the CPU, memory and I/O device controllers. No redundant power supply exists for the CPU. Redundancy at the processor unit is obtained with multiple processor units.

In a simple configuration all device controllers are connected to both I/O channels. A simple configuration may be two processors with limited memory and I/O capability.

The I/O channel for a processing unit can accommodate up to thirty-two I/O device controllers. Each device controller can control a maximum of eight devices.

Every I/O controller has two addresses, is dual-ported and is connected to two processor channels.

A one-to-one relationship exists between a controller address and the number of circuit boards it represents with the following exceptions:

1. One 3106 disc controller consists of two boards
2. The 6303 asynchronous controller board accounts for four controller addresses regardless of the number of communications lines it controls. The four controller addresses can represent from one to three boards: one 6303 plus one or two 6304 expansion boards

A fiber optic link (FOX) permits multiple configurations of up to sixteen TANDEM processors each to be directly interfaced. One 6700 FOX controller is required per node.
A special backplane upgrade and replacement is included with the 6700 controller. The FOX controller must reside in the first six (leftmost) I/O slots in the system directly under processor number zero. Any system configuration which includes FOX must consider this requirement. Some such systems may require an additional I/O cabinet to accommodate all controllers. The FOX controller consumes approximately forty-eight amperes of +5 VDC power and may impact the power configuration considerations.

A five strand one-hundred meter air plenum pre-terminated cable, model 7618, should be utilized. The 7618 cable is UL approved for use in air plenum spaces (under raised floors, above false ceilings, etc.) without need for installation in conduit (UL rating VW-1). The fifth strand is provided as an integral part of the cable and serves as a spare in case of breakage or intermittent voltage levels.

Terminal communications to the TANDEM hosts is accomplished via specific processor resident ASYNC or SYNC controllers or is off-loaded to a 6100 controller (communications processor).

Network Systems Corporation (NSC) HYPERchannel products enable two or more computer systems to communicate with each other at multi-megabit rates. A HYPERchannel network consists of one or more coaxial cables running the length of the computer room. HYPERchannel adapters are tapped into
the cable and connected to the applicable hosts at designated high speed I/O channel ports. User or NSC software creates the processing sessions among the hosts.

B. UNIQUE CONFIGURATION RULES.

Unique rules must be applied during the configuration process in addition to the basic configuration rules. These additional heuristics apply to all classes of available options (e.g., hardware, software, documentation, etc.). The discussions which follow highlight these additional considerations.

1. Hardware

Unique configuration heuristics described below apply to hardware line items.

1. One to four CPUs require one system cabinet and one patch panel. Each CPU is ordered with two megabytes of memory and is augmented with an additional two megabytes of memory.

2. Five to eight CPUs require two system cabinets, one patch panel and one expansion cabinet.

3. Nine to twelve CPUs require three system cabinets, two patch panels and one expansion cabinet.

4. Larger configurations are built using multiples of the above three rules.

5. The FLOATING POINT ARITHMETIC microcode for FORTRAN processing is only ordered for the two FMSO sites (Sites 02 and 03).

6. An Operations and Service Processor (OSP), with a TANDEM 6530 CRT attached, is ordered for each configuration of sixteen processors or portions thereof. The OSP must be capable of using an
attached Centronics Printer with a printer interface unit that permits switching among two OSPs.

7. Each system cabinet requires three I/O power modules.

8. Each system cabinet has twenty-four slots. Each controller (ex: disk controller, LP/CR controller, etc.) occupies two slots.

9. One disk controller is needed for every two disk units ordered.

10. Disk controllers must be ordered in pairs.

11. One disk patch panel is required for every four disk controllers.

12. HYPERchannel adapters may only be ordered by sites designated as stock points. Available HYPERchannel adapters are listed as follows:
   a. A140 - UNIVAC host interface.
   b. A150 - Burroughs B4800 host interface. An EBCDIC-to-ASCII Conversion RAM board is ordered with each A150 adapter to facilitate TANDEM-to-Burroughs communications.
   c. A220 - IBM host interface.
   d. A400 - Standard minicomputer interface used for TANDEM and PERKIN-ELMER hosts. Each adapter can support up to four CPUs. This is the only adapter which can exceed the one-to-one relationship between processors and adapters.
   e. A510 - FIPS Standard host interface. HYPERchannel component pricing is based upon the assumption that the maximum number of components to achieve the maximum discount have already been ordered.

13. Each HYPERchannel cabinet will accommodate up to three adapters. If TANDEM and Burroughs machines are greater than fifty feet apart, a HYPERchannel cabinet is needed for each machine. Coaxial cables in lengths from 500 to 5000 feet may be ordered as needed.

14. One patch panel cabinet is required for every ten patch panels (any type).
15. 6100 Communications Subsystem Base units come with a cabinet with room to accommodate fifteen Line Interface units (LIUs) and two Subsystem Base Add-on units. Each Subsystem Base Add-on unit can accommodate an additional fifteen LIUs. Three cable size options are available for connecting the 6100 Subsystem to hosts. Only the 60M option is ordered. Each Subsystem Base unit and Add-on unit requires two cables.

16. One TANDEM HYPERchannel patch panel is required for every four TANDEM HYPERLINK controllers.

17. One tape controller is needed for every tape drive unit.

18. One LP/CR controller is required for every line printer, card reader or card reader punch unit.

19. All TANDEM 6530 CRTs are ordered with the word processing option.

20. One ASYNC patch panel is required for each ASYNC controller. An ASYNC controller supports two asynchronous ports. At least two ASYNC controllers are required for the OSP and for redundancy. Up to two ASYNC extension boards may be added to each ASYNC controller, if needed.

21. One SYNC patch panel is required for each BYTE SYNC controller. SYNC controllers are ordered in pairs for redundancy.

22. No SYNC patch panels are ordered for BIT SYNC controllers.

23. Communications patch panel/line monitor and ARCLI components are never ordered.

24. One FOX controller is required per node. A single FOX cable connects two nodes.

2. **Software**

   Unique configuration heuristics described below apply to software line items.

   1. All FDC software is purchased on a "per site" basis (i.e., pay for the first copy only at any site) and
ordered on a "per processor" basis. This requirement includes Batch, FDC System Utilities. FDC File Security System, FDC TPS SAS, System Card Reader Support and GFE Terminal Support packages.

2. TANDEM software is purchased and ordered on a "per processor" basis. This requirement includes GUARDIAN OS. ENCOMPASS, EXPAND and COBOL packages. TANDEM EXCHANGE RJE HASP software cannot be ordered.

3. All 6100 software is ordered on a "per processor" basis. 6100 software versions must be indicated when ordering since versions differ for each site.

4. DDN Service Interface software is ordered on a "per site" basis. DDN Interface Protocol software is ordered on a "per processor" basis.

5. NETEX software packages (feature numbers 550801 through 551302) do not have any warranty period. No maintenance uplift factor should be applied to these software packages. NETEX software ordered will correspond to the NSC HYPERchannel adapters ordered. Pricing for Burroughs NETEX software is set at the maximum discount level. Pricing for TANDEM NETEX software is set at the third level. Pricing for all other NETEX software products are set at the first level.

6. Software maintenance is computed on a "per site" basis.

7. Block Structured Language (PASCAL) and FORTRAN may only be ordered for FMSO Sites 02 and 03.

8. Software components which are part of a bundled package may not be ordered separately.

9. FMSO Configuration Management and Query software may not be ordered.

10. T-TEXT software must consciously be ordered.

3. Manuals and Documentation

Four sets of manuals are available on the SPLICE contract. A predetermined number of manuals has been identified for each site. This predetermined figure is an
element of the input configuration file. Nevertheless, the actual number of manuals desired for a site must be specified during configuration processing. This is necessary since sites may not require the predetermined quantity on the first delivery.

4. Training

Training was originally planned to be ordered on a group basis. Several individual courses may be ordered either in addition to or in lieu of the group package. Such an option is supported for the following courses:

1. Hardware Overview
2. Systems Resource Management
3. Systems Tuning and XRAY
4. Data Communications
5. TANDEM Applications Language (TAL)

The addition of courses in the future will require the modification of source code and the input cost data file. This action will only apply to courses ordered on a unit basis.

5. Maintenance

Maintenance is configured on a component and monthly unit basis with few exceptions. If the normal maintenance option is selected, preventive maintenance and on-call maintenance options have zero values for both quantity and cost. If the normal maintenance is not selected, preventive
and on-call maintenance options are assigned values according to the SPLICE contract. Emergency Per-Call maintenance is specified on an hourly basis. Months of component maintenance varies based upon the warranty period specified in the SPLICE contract.

6. Other

Site Preparation (initial site preparation and installation survey) charges must be specified during the configuration process if desired.

7. Discount and Escalation Rates

Discount and escalation rates specified in the SPLICE contract vary at predetermined levels. These rates vary based upon either elapsed time relative to the contract award date or the quantity of line items ordered. The discount and escalation rates applied to line items during the configuration process must be explicitly specified. The rates entered are added to a value of one to generate the appropriate multiplication factor. Discount rate entries must be entered as negative amounts. The multiplication factor is then applied to a basic rate obtained from an input cost data file.

The heuristics described above apply to contract line items of a fifteen year life cycle ADP contract. As ADP technology is ever and rapidly changing, new requirements and pricing options are negotiated between the
government and the vendor (FDC). Accordingly, modifications to these heuristics will be necessary on a continual basis.
III. METHODOLOGY USED TO DEVELOP THE SYSTEM

The idea to pursue the development of a micro-computer knowledge-based configuration system was fostered by the need to satisfy a group project for a course of instruction in decision support systems (DSS). A member of the group was the former FMSO SPLICE project manager. Familiar with the specifics of the SPLICE project and sensitive to the problems experienced by the NAVSUP SPLICE project management staff, he proposed the development effort. Development of the proposed system would satisfy two purposes:

1. the need to complete a group project for the DSS course
2. provide an automated micro-computer knowledge-based configuration system that would help alleviate some of the NAVSUP SPLICE project staff's work load. Additionally, the proposed system would yield a more accurate, consistent and reliable configuration process.

The initial proposal was to develop a knowledge-based configuration system. No follow on development was planned as part of the initial development. TURBO Pascal was selected as the programming language of choice for the following reasons:

1. all group members were familiar with the language as a result of exposure from a previous programming course
2. a structured programming language was desired for the development effort
3. a language which supported screen-oriented functions and color was desired

4. a language which provided quick response and ease of editing and compilation to reduce development effort and minimize frustration

Other programming languages could have satisfied item 2 through 4 requirements as well, but TURBO Pascal was chosen because of the overriding requirement of item 1. This requirement was felt to be of paramount importance due to the short development time frame involved for the course. Group members felt that familiarity with TURBO Pascal would allow the development effort to be modular and completed more rapidly. The system was completed and was forwarded to NAVSUP for evaluation and comment.

A follow on course of instruction dealing with software engineering methodologies was taken. A course requirement called for the development of a project using a structured software engineering approach to software development. Feedback from the NAVSUP SPLICE project staff was favorable. Comments received indicated a strong potential for the system to significantly improve the currently manual configuration process. Follow on group development of the project was initiated. The group discussed the merits of such a system and decided to pursue development employing the software engineering methodology taught in the course. Discussion for the remainder of this chapter will focus on the entire development effort from commencement of
development to completion of the integrated configuration management system.

A. PROTOTYPE

During the initial discussions and planning of the proposed configuration system, the major concern of group members was whether the vast number of heuristics involved in the configuration process could successfully be automated during the time frame of the course. In order to meet the completion deadline, the programming effort had to be divided between group members. The strategy employed was to break the system down into five basic functional areas. Each functional area would deal with each set of heuristics described in the previous chapter with only minor exceptions. The general heuristics had to be addressed for multiple areas and a few of the smaller areas were consolidated for development efficiency.

The group strategy was to start with the first group of heuristics (hardware) and proceed in an incremental fashion. Development effort would continue until either the prototype system was finished or until the project was due. Since there were so many heuristics involved and no formal structured design or engineering methodology was conducted, there was little certainty of how much of the system would be developed.
Development commenced with the general and hardware heuristics. Initially, development was extremely slow and difficult. General and hardware heuristics encompass the majority of the heuristics associated with the configuration process and are very complex. The incorporation of these areas into the system consumed the largest amount of time during the prototype development effort. Development continued sequentially by area until all areas had been addressed. As each area was implemented, development became easier as members gained confidence and heuristics became less complicated.

As mentioned in the introduction, the initial goal in the development effort was to make the system interactive and as user friendly as possible. The screen oriented features and functions of TURBO Pascal proved to be very beneficial in this endeavor. The use of colors for screen displays helped to differentiate input fields and prompts. The ability to move the cursor anywhere on the screen and control data entry, validation and error messages formats also aided in this effort.

Upon completion of the course, the prototype configuration system was forwarded to the NAVSUP SPLICE project staff for comments and recommendations. Project staff personnel expressed considerable interest in the prototype configuration system. While the configuration system was crude, project staff personnel were enthusiastic
about the potential benefits of the system. Discussions concerning their desire to incorporate other project management functions into the system were addressed.

B. SOFTWARE ENGINEERING METHODOLOGY

The software design course requirement to develop a software system using a structured methodology coincided closely with the receipt of the NAVSUP list of comments, recommendations and additional features. Further development of the system was accomplished using a programming team concept in conjunction with the software engineering methodology.

The software engineering methodology used in the development effort is a three phased structured approach encouraged by Pressman:

1. Planning - the definition, analysis, specification, estimation and review of a process. Planning provides a preliminary indication of project viability in relationship to cost and schedule constraints

2. Design - a process of applying various techniques and principles for the purpose of defining a device, a process, or a system in sufficient detail to permit its physical realization

3. Maintenance - the diagnosis and correction of errors (corrective); the modification of software to properly interface with a changing environment (adaptive); or the incorporation of recommendations for newer capabilities, modifications of existing functions, or general enhancements following the successful development of software (perfective)
Each phase of the structured methodology is designed to minimize the difficulties associated with the software development effort. [Ref. 2]

1. **Planning**

The first step of the software engineering methodology is the planning process. During this phase of software development, the group commenced the detailed planning of the functions that were to be incorporated into the system. Initial discussions centered around the level of complexity to be attempted for the course project. During these discussions, comments, recommendations and additional features provided by the SPLICE project staff were reviewed and scoped for level of complexity.

Initial planning efforts generated a proposal to develop an integrated interactive and user-friendly system that would be composed of three major functional modules:

1. Configuration module
2. Financial analysis module
3. Configuration Management System module that would support report generation

Detailed functions for each module were further specified. Individual member previous experience and strengths were evaluated. The group was organized into a programming team concept. Each member was assigned tasks which best corresponded to his level of experience and knowledge with respect to development tasks.
Once the system functional modules were identified, the next step involved the selection of software to implement the development effort. Based upon the effort that had been expended and the enthusiasm exhibited with the prototype development, a decision was made to continue development of the configuration module using TURBO Pascal. SCREEN SCULPTOR\(^3\) was selected for the purpose of developing customized screens for the configuration module. It also employed a data entry and validation feature that could be incorporated into the configuration module with little effort. LOTUS 1-2-3 was selected as the software package for development of the financial analysis package. This selection was based upon the fact that the package was owned by a member of the group who was familiar and experienced in its use. dBASE III was selected for development of the Configuration Management module. Reasons surrounding this choice were:

1. the package was owned and readily available

2. it could be used as a shell to call and run other software packages from as well as perform the functions of configuration management using data base technology

\(^3\)SCREEN SCULPTOR is a software product available from The Software Bottling Company of New York. 6600 Long Island Expressway, Maspeth, NY 11378 (718) 458-3700. SCREEN SCULPTOR is a programming productivity tool that enables programmers to design and create input screens in minutes in either BASIC, IBM Pascal or TURBO Pascal.
3. FLASH CODE,4 a commercial screen generation software package was available and could support the generation of customized screens and perform data entry validation for both dBASE II and dBASE III. The use of such a package would help minimize development effort and ensure correct data entry.

4. dBASE III could support ten open files concurrently.

5. no other data base management software package was available that either provided the capability to customize screens to the degree desired and support an interface to FLASH CODE.

WORDSTAR was selected as the word processing software package that would be used to enable the user to view the User's Manual on-line. All packages with the exception of the two screen generation development packages were currently being used by SPLICE project staff personnel and required little investment in time to learn new packages or the outlay of funds.

Selection of the software packages posed some problems which had to be overcome prior to further development. LOTUS 1-2-3 and dBASE III both required special file formats and interfaces between input and output of each functional module. Special conversion procedures had to be developed to overcome these interface difficulties.

FLASH CODE is a software product available from The Software Bottling Company of New York, 6600 Long Island Expressway, Maspeth, NY 11378 (718) 458-3700. FLASH CODE is a programming productivity tool that provides dBASE II or dBASE III programmers the capability to use either screens or pop-up windows/help menus that instantaneously flash up on the screen.
The Pascal configurer module had to be developed to generate an output file that would allow the viewing and processing of both text and numerical fields when imported into LOTUS 1-2-3. The output file from the LOTUS 1-2-3 financial analysis module stripped off all text and header data following financial verification and saved as a "PRN" data file. A dBASE III work data base had to be created using a structure that was compatible with the "PRN" data file. This "PRN" file was later appended to the dBASE III work data base and converted to a dBASE III data entry format.

With the module interfaces resolved, each functional module was further developed and refined to identify all data elements involved with the functional process. Data flow diagrams documenting all required data elements and processes were generated for each functional module. Two data flow diagrams are provided in Appendix B to serve as representative examples of this process. Each data flow, input file and functional process was further specified in detail through the use of various module descriptions. An example of each of these description modules is provided in Appendix B. The formats of each of the descriptions used in the definition process were modifications of formats specified in [Ref. 2] and [Ref. 3]. A Bachman diagram, supplied in Appendix B, was used to document the data base relationships associated with the configuration management.
module. The generation of all functional module data interdependency charts signaled the completion of the planning phase.

2. Development

With the definition of all data element relationships, interdependencies and functional interfaces defined, the group commenced the development phase of the methodology. Using the data flow diagrams, data flow and process descriptions generated during the planning phase, each data process or bubble was decomposed into more detailed sub-functional processes.

Sub-functional processes were developed by exploding each bubble from the data flow diagram and decomposing the process to its lowest functional level through several layers of abstraction. The lowest levels of abstraction are procedure oriented and are stated in terms that can be directly implemented. Several guidelines for the process are involved and are outlined in Pressman [Ref. 2]. The overall objective of this decomposition process was to arrive at a description of each functional process to a level that would support modular development. Appendix B contains a few structure charts which are representative examples of the decomposition process.

The idea behind decomposing each process to its lowest functional description is to ensure that the scope of
effect\textsuperscript{5} of a module is maintained within the scope of control\textsuperscript{6} of that module [Ref. 2]. Another concept of the engineering methodology designed to aid in the development and maintenance of software systems is that of information hiding\textsuperscript{7}. These concepts were applied to the design phase of development to ensure modularity of the system. The structure of the system was designed in a way that would facilitate future maintenance.

With all processes defined, team members began coding the various modules. Coding was accomplished in a top-down modular fashion to facilitate a phased implementation plan. As each module was completed, it was integrated into the overall system and tested to ensure accurate performance. Coding continued until the project was due for submission. At the end of the course, the configuration and financial analysis modules were complete. The third module, the data base configuration management system, had a basic structure that would support a minimal number of configuration reports. This module would be

\textsuperscript{5}Scope of effect of a module is defined as how other modules are affected by decisions which are made within the module. [Ref. 2: p. 170]

\textsuperscript{6}Scope of control of a module is the number and degree of control which is exerted on other modules by the controlling module. [Ref. 2: p. 170]

\textsuperscript{7}Information hiding is the concept whereby procedures and data information within a module are invisible to other modules. This concept helps achieve modularity during development. [Ref. 2: pp. 156-157]
finished as a follow on project under the maintenance phase. The system was forwarded to the SPLICE project staff for evaluation.

3. Maintenance

The structured design and development methodology employed in the development of the micro-computer knowledge-based integrated configuration management system proved to be very beneficial. Completion of the data base configuration management module was straight forward due to this design methodology.

The data base configuration management system was completed as a follow on project for a course of instruction in data base design. Since a foundation already existed as a result of the initial system development, continued development fell into the category of maintenance. The development of the configuration management module used three methods of maintenance. Each maintenance category is defined briefly in the methodology introductory discussion near the beginning of this chapter.

Continued development of the configuration management module was undertaken. Feedback from the SPLICE project staff highlighted errors which required correction - corrective maintenance. Also, due to contract negotiations and modifications, certain heuristics required modification - adaptive maintenance. Additionally, the data
base design course highlighted more efficient methods of accomplishing functional processes in lieu of methods used during the development phase of the system - adaptive and perfective maintenance.

The maintenance effort and system enhancements proposed by the NAVSUP SPLICE project staff were reviewed and evaluated for level of implementation difficulty. Each change was classified according to the type of maintenance involved. A development schedule was established and development effort continued.

The first maintenance actions addressed were corrective maintenance issues. Each potential error was evaluated in terms of its impact on the basic system structure. Errors were also evaluated in terms of whether the condition fell within the initial capabilities designed for the system. Some of the potential errors were found to be outside the scope of the initial design and were not attempted. SPLICE project staff personnel were informed of these conditions and were instructed on how to deal with the conditions.

Changes to the initial environment were addressed next. Contract negotiations are continuing and result in contract modification requirements. These modifications were evaluated to identify the degree of modification required to the basic system structure. While some modification was required, the majority of the changes
involved the configuration module. The decomposition of the logical functions to their lowest levels coupled with the high degree of cohesion\(^8\) and low degree of coupling\(^9\) of both modules and data made maintenance almost effortless.

The last maintenance area involved refining the methods by which tasks were performed. Knowledge gained from the data base design course identified more efficient means of accessing certain files. Also, certain initial relationships did not follow the relational normal forms associated with relational data base design [Ref. 4] and [Ref. 5]. Thus, certain files had to be restructured. Other changes involved eliminating unnecessary statements and optimizing certain functions, loops and file accesses. Modification of certain file accesses resulted in the reduction of response times in some cases by eighty to ninety percent.

Completion of the data base configuration management module marked the final development of the micro-computer knowledge-based interactive configuration management system for the SPLICE project. NAVSUP SPLICE project staff personnel have the system and are currently using the system.

\(^8\)Cohesion is a measure of the relative functional strength possessed by a module (i.e. a cohesive module should only perform one thing or function) [Ref. 2: p. 158]

\(^9\)Coupling is a measure of the relative interdependencies between modules (i.e., the degree to which other modules are dependent upon interfaces and data) [Ref. 2: p. 161]
for initial configurations. Once current sites under configuration are loaded to system data bases, sites previously configured will be loaded. The SPLICE project manager now has the capability to configure sites, perform financial and "what-if" analysis and generate a wide variety of reports to aid in the management of the project. The system report generation facility also enables the project manager to track components by serial number and location. The development of the micro-computer knowledge-based interactive configuration management system has provided the SPLICE project manager with the capability not only to evaluate overall project performance, but also to evaluate the contract vendor's performance with regard to contract requirements.

C. SUMMARY

The development of the micro-computer knowledge-based interactive configuration management system involved several different development methodologies. The success of its development could not have been realized without the inclusion of all methodologies.

Prototyping, while not a solution by itself, identified several problems with the original system design and data entry method. It also highlighted several areas which required modification to achieve the goal of developing a user-friendly system.
The execution of the software engineering methodology described by Pressman [Ref. 2] helped to identify all of the functional tasks for logical incorporation into the system. The use of the various module descriptions identified all of the essential data elements, flows and processes. The use of these descriptions further helped to minimize development time and prevent needless rework. Incremental implementation of completed modules kept the development effort on schedule. The use of commercially proven and tested "off-the-shelf" packages further helped to minimize the development effort.

The SPLICE micro-computer knowledge-based interactive configuration management system is an active system. As with any software system, maintenance must be performed to maintain the system current with its operational environment. The SPLICE configuration management system is no different. Due to a changing environment and requests for further enhancements to the system, a backlog of changes currently exists.

Due to the methodologies used in the design and development of the SPLICE configuration management system, the backlog and future changes should be able to be incorporated into the system with minimal confusion or effort.
IV. SYSTEM EXECUTION DIALOGUE

As discussed in previous chapters, the micro-computer knowledge-based configuration management system is an interactive and user-friendly system. Additionally, the system is an integrated system composed of three functionally separate modules:

1. configuration module - developed using TURBO Pascal
2. financial and "what-if" analysis module - developed using LOTUS 1-2-3
3. configuration management and report generation module - developed using dBASE III

Integration of the system was possible through dBASE III's ability to run other programs during system execution. This feature allowed dBASE III to be used as the shell or driver for the system.

Following discussions describe a typical system execution dialogue. All screen formats mentioned or referenced may be found in Attachment 2 of Appendix A. The system has no on-line help facility other than the on-line User's Manual. Review of the User's Manual may only be accomplished from the system's opening menu (Screen 1). Detailed information regarding system execution is addressed in Appendix A.
A. SYSTEM INITIATION

With initial installation complete and the target system's power on, type the command SPLICE at the DOS command prompt to initiate system execution. The first screen viewed is the Function Selection Menu - Screen 1. From this menu, the user may select any one of six possible options.

B. CONFIGURE A SITE

The first function normally performed would be to configure a site for SPLICE installation. This action is accomplished by selecting menu option 1 from the Function Selection Menu. Selection of this option invokes the Pascal Configuration Module. The user, having accumulated the applicable data for the site to be configured and recorded the information on a copy of Attachment 1 of Appendix A, would commence the configuration process.

The user would first see a module logo and version screen (Screen 2) followed by five data entry screens (Screens 3 through 8) and a final output screen (Screen 9) identifying the output file name to be imported into the financial analysis module. The data field sequence of Attachment 1 to Appendix A is in the sequence of data entries expected for screens 3 through 8.

Screen 3 is a list of designated SPLICE sites. Screens 4 through 8 are the applicable data entry screens. Data
entry is segmented into component and data types (ex: discount and escalation rates, hardware, software, etc.). The output data file name is presented as part of the final display to the configuration module (Screen 8). The output file is formatted for data entry into the financial analysis module. Following completion of the configuration process, the user is returned to the Function Selection Menu.

C. PERFORM FINANCIAL ANALYSIS ON SITE DATA

Financial analysis and delivery order preparation is the next function to be performed. Selection of menu option 2 from the Function Selection Menu invokes the execution of the financial analysis module using the LOTUS 1-2-3 system. The output file previously generated from the configuration module may then be viewed.

Several LOTUS macros, described in detail in Appendix A, enable the configuration module calculations and computations to be verified. "What-if" analysis may also be performed to evaluate the impacts of system costs relative to options selected and/or modify a system configuration to coincide with the current funding environment. Screen 13 is a partial example of how the data is presented in the financial analysis module. Upon completion of the configuration analysis, the data file is formatted for input into dBASE III data base files. Following financial
analysis termination, the user is returned to the Function Selection Menu (Screen 1).

D. INTERACT WITH THE CONFIGURATION MANAGEMENT AND REPORT GENERATION SUB-SYSTEM

Execution and interaction with the configuration management and report generation sub-system is invoked by selecting menu option 3 from the Function Selection Menu (Screen 1). The Process Selection Menu (Screen 14) is displayed and reveals nine additional options from which to choose.

1. Load New Delivery Order Data

The most common option to select will be menu option 1 - load the formatted file from the financial analysis module to the various data bases. The process is menu driven requiring answers to a few questions presented on screens 15 and 16. The data loading process adds new records to three data bases. If the input file is very large, the loading process may be lengthy.

Completion of loading data to the three data bases signals the interim completion of the configuration process for a site. No further data for the site may be loaded to the data bases until the equipment is received at the site. From this point, the user may return to the Process Selection Menu and obtain any of several reports extracted
in a variety of formats or return to the Function Selection Menu and choose another processing option.

2. **Load Serial Number and Manual Data**

   Following the receipt of ordered components at the applicable site, the user may load the serial numbers of the hardware components and the names of the accompanying hardware and software manuals received. This function is a two step process.

   Serial numbers may be loaded to the serial number data base by selecting menu option 6 from the Process Selection Menu, whereby the Serial Number Maintenance Menu (Screen 32) is displayed. Selection of menu option 1 results in the presentation of the Serial Number Update Format screen (Screen 33). To enter the applicable serial numbers, the user must provide the system with three data elements to load the serial number data:

   1. site number
   2. effective date of the applicable delivery order
   3. feature number of the component

   Once all three data elements have been entered, the serial number may then be entered. This process must be iterated for each serial number to be loaded to the data base. Since neither serial number nor manual information is available during the initial data load process, it is necessary to specify all three serial number data elements to ensure data
and file integrity. Following entry of the last serial number, the user terminates the update process by selecting the exit (X) option. This returns the user to the Serial Number Update Format screen (Screen 33). The user may either review the serial numbers just entered or return to the Process Selection Menu to initiate the loading of the applicable manual data.

Following entry of the serial number data, the applicable manual description data may be loaded to the Manual data base. This is accomplished by selecting menu option 5 from the Process Selection Menu, whereby the Manual Maintenance Menu (Screen 27) is displayed. To add manual descriptions to the manual data base, select menu option 1. The Manual Addition Format screen (Screen 28) is displayed. To enter the manual descriptions, first enter the applicable site number followed by the associated feature number for the manual description to be loaded.

Following entry of the last manual description, terminate the addition process by selecting the exit (X) option. This returns the user to the Manual Maintenance Format screen (Screen 27). The user may either review the manual descriptions just entered or return to the Process Selection Menu to initiate another process selection.
3. Generate a Maintenance Delivery Order

At the commencement of each fiscal year, the NAVSUP SPLICE project staff must initiate a delivery order to cover the maintenance and rental services for the current fiscal year for each configured SPLICE site. To accomplish this task, select menu option 8 from the Process Selection Menu (Screen 14). The Maintenance Delivery Order Generation Program screen (Screen 66) is presented and requires five inputs. First, the applicable site number for which the maintenance delivery is to be generated is entered. Then four discount or escalation rates are entered. These rates are based upon pre-determined terms negotiated in the SPLICE contract. These rates are based upon total number of components ordered and the elapsed time relative to the contract award.

A new formatted file (NEWDO.PRN) is generated to be imported into the financial module where computations and calculations are verified in the same manner discussed in section C above. Once the data has been verified financially correct in the financial module, the maintenance delivery order is ready to be printed. Program execution then automatically returns the user back to the Process Selection Menu where another process selection may be made.
4. Generate a Report

A variety of eight different reports are available from the report generation sub-system. Reports are available for:

1. the overall project
2. a particular site
3. a delivery order issued on a particular date

Within these categories, reports may further be broken down by:

a. equipment type
b. serial number

Delivery order equipment type reports may be obtained either with or without unit price data in the report.

The generation of any one of the eight available reports is obtained by initially selecting menu option 7 from the Process Selection Menu, whereby the Report by Type Menu (Screen 36) is displayed. Depending on the type of report desired, further menu options are selected. Screens 36 through 65 are examples of the various menus and report formats that are obtainable from the report generation system but are not discussed in detail.

E. REVIEW THE ON-LINE USER'S MANUAL

The on-line User's Manual may be viewed any time the user is viewing the Function Selection Menu (Screen 1). As stated before, no on-line help facility is available during
functional module execution. The on-line User's Manual uses WORDSTAR as the word processing package to display system execution instructions to the user. As such, the ability to jump to a specific page or process description does not exist. Following termination, the user is returned to the Function Selection Menu (Screen 1).

F. TERMINATE SYSTEM EXECUTION

When all system functions have been performed and the user desires to terminate system execution, two options are available. Menu options 5 and 6 on the Function Selection Menu (Screen 1) allow the user to either terminate system execution and return to the dBASE III environment (dot prompt) for further interactive queries or terminate system execution and return to the DOS operating environment. The most common selection will likely be to terminate system execution and return to the DOS operating environment.
V. COST BENEFIT AND EFFECTIVENESS

Prior to the development of the micro-computer knowledge-based integrated configuration management system for the NAVSUP SPLICE project staff, the first eight of a possible sixty-two initial site configurations were processed in a semi-automated fashion. While LOTUS 1-2-3 was used as the medium to produce the final form delivery order, a considerable amount of the heuristic processing still was manual. The developed system eliminates all such manual processing, except for gathering the initial sizing study input data.

Within the NAVSUP SPLICE project staff, one mid-grade GS-12 government employee is currently responsible for all SPLICE site configuration processing, project configuration management and vendor contract performance monitoring. Average annual salary for this grade level for a step five position is approximately thirty-six thousand dollars.

In the current phase of the project life cycle, sites are being configured for their initial equipment and associated software components. Existing sites with initial configurations require maintenance delivery orders generated to support continuing maintenance services on an annual basis. As mentioned in the introduction, errors discovered in delivery orders submitted to the vendor for processing
are corrected, with an additional charge\(^\text{10}\) levied upon the government for the additional service. Due to the minimum number of sites that have been configured and are in operational status, there currently is little configuration management being performed.

To evaluate the benefit and effectiveness of the developed system, certain (worst case) assumptions are made:

1. based upon previous experience, each delivery order supplied to the vendor will contain errors

2. the government will incur a five thousand dollar additional charge for vendor corrections to initial configuration delivery orders containing errors

3. the government will incur a one thousand dollar additional charge for vendor corrections to maintenance delivery orders containing errors (no experience exists to evaluate the accuracy of this assumption and is therefore an anticipated worst case assumption)

Since only a few of the designated sites are currently operational, the one GS-12 employee has managed to keep pace with the work load. Without the development of the micro-computer knowledge-based integrated configuration management system, this effort would not be possible and

---

\(^{10}\text{Charges of up to five thousand dollars per delivery order to correct existing errors have been experienced.}\)
would most likely require the hiring of another lower grade employee on a full time basis in the future.

During the next two calendar years, the remaining initial site configurations are going to be processed. Figures based on the worst case assumptions stated above, suggest that the developed system has the potential to yield savings of close to two-hundred and fifty thousand dollars for the initial configuration process alone. Since each site must have a maintenance delivery order generated each fiscal year to account for increases or decreases in maintenance rates for services, the potential exists to realize additional savings of approximately sixty thousand dollars for each remaining year of the project life cycle.

The SPLICE contract contains predetermined discount and escalation rates which were negotiated and written into the contract. Certain discounts depend upon the quantity of components previously ordered and are graduated according to predetermined procurement levels. The ability of the GS-12 employee to currently identify these discount levels is

11 Once all SPLICE sites have been configured for initial equipment and component installation, configuration management within the project will come to the forefront. Due to the large number and variety of components that may exist for any site which can have an impact on the discounts that are applicable to component, this phase of contract monitoring and execution becomes critical in terms of cost effectiveness.

12 Approximately twenty sites are scheduled for configuration during CY 1986 and approximately thirty sites are scheduled for configuration during CY 1987.
accomplished solely through a manual process. Each delivery order previously issued has to be manually totaled to arrive at each component's project procurement total. Through the developed system's report generation facility, potential discounts can be identified in a matter of seconds. The potential savings that may be realized in this manner are difficult to quantify. I feel that it is safe to say that over the life cycle of the project, substantial savings as a result of this new capability can result.

The developed system provides the NAVSUP SPLICE project staff with the ability to monitor the vendor's performance relative to contract specifications and perform configuration management for the overall project. While the contract provided a configuration management package line item for these services, development of the system precludes the need to procure the option priced at roughly one-hundred thousand dollars.

The developed system provides the project staff with extensive capabilities needed to properly execute their functions as overseers of the contract and does so in an automated and efficient manner. These capabilities are believed to be developed to a level that will allow the existing project staff employee to perform these functions in roughly half the time experienced prior to system implementation. This increased efficiency should realize a
minimum savings of approximately eighteen thousand dollars each year for the project staff budget.

As seen from the above analysis, the development and implementation of the micro-computer knowledge-based integrated configuration management system for use by the NAVSUP SPLICE project staff provides a more efficient method with increased capability to effectively execute project manager responsibilities and monitor vendor performance. Potential savings realized through the use of this system will be at least eighteen thousand dollars annually for the next few years with the potential to save two-hundred and fifty thousand in the initial configuration process and sixty thousand dollars in annual maintenance modifications.


APPENDIX A

THE NAVAL SUPPLY SYSTEMS COMMAND
STOCK POINT LOGISTICS INTEGRATED COMMUNICATIONS ENVIRONMENT
(SPLICE)
SYSTEM CONFIGURER AND CONFIGURATION MANAGEMENT SYSTEM
USER'S MANUAL

Document No. BBC - 01
1 January 1986
APPENDIX A: USER'S MANUAL

Record of Changes

Original 1 January 1986
APPENDIX A: USER's MANUAL

List of Effective Pages

Page 1 through 44
Page A1-45 through A1-49
Page A2-50 through A2-83
Page A3-84
Acknowledgements

1-2-3 is a registered trademark of Lotus Development Corporation.

dBASE II and dBASE III are registered trademarks of Ashton-Tate.

Flash Code and Screen Sculptor are registered trademarks of The Software Bottling Company of New York.

Turbo Pascal is a registered trademark of Borland International

Wordstar is a registered trademark of MicroPro International Corporation.

Zerodisk is a registered trademark of Quaid Software Limited.
Recognition

The development of the integrated SPLICE Configuration Management System involved several people. The effort devoted to the finished product was spread over a nine month period. The system was also used to satisfy project assignments in several core courses leading to the receipt of the Master of Science degree. Recognition is acknowledged for the persons listed below for their participation in the completion of the SPLICE Configuration Management System.


Lieutenant Commander Winston H. Buckley, Supply Corps, U. S. Navy - programming team librarian, documentation generation.


Lieutenant Commander Gary R. Harmeyer, Nurse Corps, U. S. Navy - dBASE III programming assistant, document generation.

Table of Contents

Record of Changes ----------------------------------------------- 2
List of Effective Pages ----------------------------------------- 3
Acknowledgements ----------------------------------------------- 4
Recognition ----------------------------------------------------- 5
Table of Contents ----------------------------------------------- 6
Introduction ----------------------------------------------------- 9
Background -------------------------------------------------------- 9
Why The System Configurer and Configuration Management System ----------------------------------------------- 9
Input Data --------------------------------------------------------- 11
SPLICE System Configurer and Configuration Management System Files ----------------------------------------------- 11
System Preparations --------------------------------------------- 14
System Execution ----------------------------------------------- 14
FUNCTION 1: Execute the Pascal Configurer --------------------- 15
FUNCTION 2: Perform LOTUS 1-2-3 Financial or "What-If" Analysis ----------------------------------------------- 18
FUNCTION 3: Execute the dBASE III Configuration Management System ----------------------------------------------- 23
  1. Load a new Deliver Order into the Configuration Management System ----------------------------------------------- 24
  2. Perform maintenance on the Equipment File ----------------------------------------------- 25
     a. Modify an Equipment File record ----------------------------------------------- 25
     b. Review an Equipment File record ----------------------------------------------- 26
  3. Perform maintenance on the Equipment Description File ----------------------------------------------- 27
APPENDIX A: USER's MANUAL

4. Perform maintenance on the Site Name File
   a. Modify a Site Name File record
   b. Review a Site Name File record

5. Perform maintenance on the Manual File
   a. Add a new manual description record
   b. Update a manual description record
   c. Delete a manual description record
   d. Review a manual description record

6. Perform maintenance on the Serial Number File
   a. Modify a Serial Number File record
   b. Review a Serial Number File record

7. Generate reports for the Project, a specific site or
   a particular date
   a. Overall Project reports
   (1) Reports by Equipment type
   (2) Reports by Serial Number
   b. Reports for a Particular SPLICE Site
   (1) Reports by Equipment type
   (2) Reports for Manuals
   (3) Reports by Serial Number
   c. Report for a Delivery Order issued on a
      particular date
      (1) Report by Equipment type with unit prices
APPENDIX A: USER's MANUAL

(2) Report by Equipment type without unit prices -- 40
(3) Report by Serial Number ---------------------- 41

8. Generate a Maintenance Delivery Order for a specific SPLICE site -------------------- 41
9. Generate Mailing Labels for all SPLICE sites ------- 42

FUNCTION 4: View the on-line User's Manual --------- 43

System Output ---------------------------------- 43
Exception Reports ------------------------------- 43
Limitations ------------------------------------- 43
Command Sequence -------------------------------- 44
Who To Call ------------------------------------- 44

Attachments:
1. Selection Criteria for SPLICE Configurations ---- A1-45
2. Screens Formats ------------------------------- A2-50
3. Installation Procedures ---------------------- A3-84
1.0 Introduction.

This manual is designed to provide information and guidance to the SPLICE integrated system user. The integrated system components include: 1 - the SPLICE System Configurer, 2 - the LOTUS 1-2-3 financial and "what-if" analysis system, 3 - the dBASE III Configuration Management System, and 4 - the Wordstar on-line User's Manual.

1.1 Background.

The Naval Supply Systems Command (NAVSUP) conceived and developed the Stock Point Logistics Integrated Communications Environment (SPLICE) project. The SPLICE project purpose is to:

a. Provide state-of-the-art local and long haul telecommunications capabilities to 62 NAVSUP Stock Points.

b. Provide interactive and distributed ADP processing capabilities to SPLICE sites.

c. Provide capacity relief to aging Burroughs hosts at the Stock Points.

d. Standardize and upgrade, via mass replacement, the myriad of minicomputers existing at Stock Points.

NAVSUP initiated a competitive solicitation for "fault-tolerant" hardware and software to achieve these goals. The solicitation was completed in November 1983. The winning vendor, Federal Data Corporation (FDC), proposed TANDEM hardware and software to meet most of the solicitation processing and local communications requirements. FDC proposed Network System Corporation hardware and software to meet the local inter-host communications requirements.

1.2 Why The System Configurer and Configuration Management System.

Shortly after the SPLICE contract award, hardware and software components had to be ordered. NAVSUP faced a dilemma. Only a few SPLICE personnel had worked closely with the SPLICE acquisition benchmark and negotiations. These few people were the only personnel that had sufficient
knowledge of the systems to configure and generate delivery orders. These personnel developed initial orders by hand to meet the immediate need. Many minor errors were encountered with these initial orders. FDC corrected and returned the orders to the government and received additional compensation for their efforts.

This manual configuration process was later automated using a software product called SUPERCALC2. It has subsequently transitioned to LOTUS 1-2-3. The basic method of developing these orders remained virtually manual. These few SPLICE personnel, with FDC assistance, developed a series of "rules of thumb" used to configure individual site systems. Many of the original SPLICE group have moved on, taking their knowledge of the systems with them.

This SPLICE Configurer and Configuration Management System software is a knowledge based system designed to codify these "rules of thumb". This integrated system will enable NAVSUP to develop and maintain SPLICE configurations and delivery orders and perform configuration management on the project. Three software products were created in this phase of development to:

a. Configure initial SPLICE site systems by answering a series of questions. SPLICE.COM (written in TURBO Pascal) produces structured delivery orders that must be imported into LOTUS 1-2-3. LOTUS 1-2-3 performs financial review and analysis before loading the dBASE III Configuration Management data bases.

b. Restructure the SPLICE.COM output file into LOTUS 1-2-3 format. A series of macros assist in the regeneration of the delivery order into LOTUS standard formula format. Following the conversion, three options exist: 1 - print the delivery orders, 2 - prepare archival files, or 3 - prepare the output file needed for the dBASE III Configuration Management system.

c. Restructure the LOTUS 1-2-3 output file into dBASE III format. dBASE III command language modules import and convert the LOTUS output file into dBASE III format. They also either generate or update the three dBASE III Configuration Management data bases. This allows the user to generate selected configuration management reports from the three data bases. MAINTDO.PRG, a dBASE III module, generates maintenance delivery orders from the configuration management data bases. These maintenance delivery orders
must be imported into LOTUS 1-2-3 for final financial review and analysis.

2.0 Input Data.

The following paragraphs describe the integrated system data input requirements. The following discussion describes the files required to execute the system and the associated screen formats.

2.1 SPLICE System Configurer and Configuration Management System Files.

The SPLICE System Configurer and Configuration Management System can only be run on a hard disk system, with the following minimum files (refer to Attachment 3 for system installation procedures):

GROUP 1 FILE-IDs (SPLICE Configurer)

a. COSTS.IN  
b. CONFIG.SIT

c. SPLICE.COM  
u. SPLICE.SCR

GROUP 2 FILE-IDs (LOTUS 1-2-3 Financial Analysis)

e. 123.EXE  
   (Associated files for LOTUS version 1A not shown but are also required.)

f. SKELETON.WKS  
g. MAINTORD.WKS

GROUP 3 FILE-IDs (dBASE III Configuration Management System)

h. DBASE.COM  
   (Associated files for dBASE III version 1.1 not shown but are also required.)

i. CONFIG.DBF  
j. CONFIG.NDX  
k. CONFMOD.PRG

l. CONREV.PRG  
m. CONFUPD.PRG  
n. DATERPTS.PRG

o. DELAY.PRG  
p. DESCRIP.DBF  
q. DESCRIP.DBT
Several of the dBASE III command language modules require considerable time to execute. An IBM-PC/XT operating with a clock speed of 6 MHz or greater or IBM-PC/AT provides better performance.
Three additional TURBO Pascal source code files are provided since the Configurer system was developed in Borland International's TURBO Pascal and Software Bottling Company's SCREEN SCULPTOR:

GROUP 1 FILE-IDs

a. SPLICE.PAS  
b. SPLICE1.PAS  
c. SPLICE2.PAS

GROUP 1 files must reside on a subdirectory named \TURBO. GROUP 2 files must reside on a subdirectory named \LOTUS. Group 3 files must reside on a subdirectory named \DBASEIII. The file USERS.MAN must be present on a subdirectory named \WORDSTAR if the User's Manual is viewed on-line (Function Selection Menu option 4). A version of WORDSTAR must also exist on the subdirectory.

Software Bottling Company product FLASH CODE must be purchased to run the dBASE III Configuration Management System. All command language modules in the dBASE III Configuration Management System use a memory resident program FLASHUP.COM. FLASHUP gives dBASE III the extra capabilities of instantly flashing up screens and instantly popping up windows. Load this command module into the computer memory before running dBASE. The SPLICE.BAT

1 SCREEN SCULPTOR is a software product available from The Software Bottling Company of New York, 6600 Long Island Expressway, Maspeth, NY 11378 (718) 458-3700. SCREEN SCULPTOR is a programming productivity tool that enables programmers to design and create input screens in minutes in either BASIC, IBM Pascal or TURBO Pascal.

2 FLASH CODE is a software product available from The Software Bottling Company of New York, 6600 Long Island Expressway, Maspeth, NY 11378 (718) 458-3700. FLASH CODE is a programming productivity tool that provides dBASE II or dBASE III programmers the capability to use either screens or pop-up windows/help menus that instantaneously flash up on the screen.

3 FLASHUP is a memory resident program supplied with FLASH CODE that enables dBASE II or dBASE III programmers to use screens and pop-up windows/help screens which instantly flash up on the screen rather than the dBASE painting method.
command batch file automatically accomplishes this process. FLASHUP is licensed to individuals for use along with either dBASE II or dBASE III programs and may be moved from one computer to another. Any number of people may use FLASHUP, providing there is no possibility of using it concurrently in two or more locations.

Both Software Bottling Company products, SCREEN SCULPTOR and FLASH CODE must be purchased to perform system maintenance on system screens and windows.

2.2 System Preparations.

Fill out a copy of Attachment 1 before executing the SPLICE Pascal Configurer and Configuration Management System modules. Having this information before beginning a session will greatly facilitate system use.

Turn on the IBM-PC AT target system and the 132 column printer's power. Ensure that the minimum required software listed above is loaded on the active hard disk subdirectories specified. Make subdirectory \DBASEIII the default directory.

2.3 System Execution.

Execute the SPLICE Pascal Configurer and Configuration Management System by entering the command SPLICE at the system prompt (ex: C\>SPLICE).

Several copyright notices will appear on the screen after a few seconds delay for system startup. The processes described below are then available: (See Attachment 2 for screen formats).

Screen 1: The Function Selection Menu is the opening screen for the integrated system. Six options exist from which to choose. Option 1 permits the configuration of a SPLICE site. Option 2 uses LOTUS 1-2-3 to perform financial or "what-if" analysis. Option 3 opens the dBASE III SPLICE Configuration Management System. Option 4 reviews the User's Manual on-line. Option 5 returns the system to the dBASE III system prompt. Option 6 returns the system to the DOS prompt. The following discussion is limited to options 1 through 4. Only entries in the range 1 - 6 are valid. The default value is 1.
2.3.1 FUNCTION 1: Execute the Pascal Configurer

Select option 1 (from the Function Selection Menu - Screen 1) to configure a SPLICE site. If the Function Selection Menu is not displayed, select the "Return to " option of the current menu until the Function Selection Menu appears. If a process is active, select the option that terminates the process. Once a menu appears, select the "Return to " option of the current menu until the Function Selection Menu appears. Select option 1 when the Function Selection Menu appears. The first screen of the SPLICE Pascal Configurer (Screen 2) appears.

Screen 2: The opening screen of the Pascal configurer module requires no input.

Screen 3: A list of sites which may be configured appears. Insert an integer value between 01 and 58 to select a currently designated site. Site numbers 59 through 62 are reserved for future designation. Site Number 23 (NAS Oceana) is deactivated and no longer is a designated SPLICE site.

Screen 4: Enter the discount and escalation rates, output file name, number of months of maintenance, and effective delivery order date. Data input ranges apply as described below:

a. FDC SNA Interface Discount Rate: 0.00 - 9.99
b. Non-LCN Purchase Discount Rate: 0.00 - 9.99
c. LCN Purchase Discount Rate: 0.00 - 9.99
d. SPLICENet Software Maintenance Discount Rate: 0.00 - 9.99
e. SPLICENet Software Purchase Discount Rate: 0.00 - 9.99
f. Emergency Maintenance Escalation Rate: 0.0 - 9.9

g. LCN Hardware Maintenance Escalation Rate: 0.000 - 9.999
h. LCN Software Maintenance Escalation Rate: 0.000 - 9.999
APPENDIX A: USER's MANUAL

i. Installation Escalation Rate: 0.000 - 9.999
j. Training Escalation Rate: 0.00 - 9.99
k. Documentation Escalation Rate: 0.00 - (-9.99)
l. Maintenance Escalation Rate: 0.000 - 9.999
m. Output file name: any 8 alphanumeric characters
n. Hardware Maintenance Months: 0 - 12
o. Effective Date: 01/01/84 - 12/31/99

On entry of the effective date, confirm the input values by entering a "Y" to the prompt "Do you accept the input values thus far? Yes or No". The Default value is "N".

Screen 5: Enter the hardware quantities suggested by the Navy Fleet Material Support Office Sizing Study, as transcribed to Attachment 1. The following data input ranges apply:

a. Processors: 0 - 256
b. Centronics Printers: 0 - 12
c. TANDEM CRTs: 0 - 999
d. 128 MB Disks: 0 - 128, in EVEN quantities
e. 240 MB Disks: 0 - 128, in EVEN quantities
f. 540 MB Disks: 0 - 128, in EVEN quantities
g. Non-6100 ASYNC Controllers: 0 - 64. There should be at least two in the initial order for each OSP; subsequent quantities are at the user's discretion.
h. Non-6100 ASYNC Extension Boards: 0 - 2
i. Bit SYNC Lines: 0 - 128
j. Byte SYNC Lines: 0 - 128
k. Tri-Density Tape Drives: 0 - 128
APPENDIX A: USER's MANUAL

1. Reader/Punches: 0 - 12
m. Card Readers: 0 - 12
n. 1000 LPM Printers: 0 - 16
o. 600 LPM Printers: 0 - 16
p. LCN Coaxial Cables (Trunks): 0 - 2. Input is only allowed for sites designated in file CONFIG.SIT as Stock Points (S).
q. 6100 Line Interface Units (LIUs): 0 - 256
r. LCN Interface Adapters (multiple entries): 0 - 256. Input is only allowed for sites designated in file CONFIG.SIT as Stock Points (S).
s. Cabinets: 0 - 16 for computed; 0 - 8 for extra. The system computes the required numbers for the 4 types of cabinets and presents this in the COMP field. Additional quantities may be entered in the XTRA field within the allowed ranges specified above as desired.
t. Max Distance Between Computers: A - F. Input is only allowed for sites designated in file CONFIG.SIT as Stock Points (S).

On completion of the Max Distance input value, confirm the input values by entering a "Y" to the prompt "Do you accept the input values thus far? Yes or No " The default value is "N".

Screen 6: Select various software packages and the number of both NETEX and SPLICENet software maintenance months desired. The system only accepts "Y" or "N" entries for software packages. The system only accepts integers in the range 0 - 12 for software maintenance months entries. Network Maintenance Facility (NMF) software is divided into either a group package or individual packages. If the user selects the group package, none of the individual packages can be selected. The cursor moves directly to the NETEX Maintenance Months field. If the NMF group package field response is "N", the user may select each individual package if desired. On completion of the entry for the number of months of SPLICENet software maintenance desired, confirm the input values by entering a "Y" to the prompt " Do you
accept the input values thus far? Yes or No ». The default value is "N".

Screen 7: Enter the quantities for system documentation, training group and courses, and months of Emergency Per-Call Maintenance. Indicate whether to include Site Preparation charges.

The allowable range for documentation and training courses is 0 - 20. The allowable range for Training Groups is 1 - 5. The allowable range for months of Emergency Maintenance is 0 - 12. The allowable inputs to Site Prep charges are "Y" or "N". On completion of the Site Prep charges, confirm the input values by entering a "Y" to the prompt "Do you accept the input values thus far? Yes or No ». The default value is "N".

Screen 8: The configurer software module sign-off screen requires no input. The system displays the output file name used for this configuration run in the sign-off message.

The system returns to the Function Selection Menu (Screen 1) to await the next selection.

2.3.2 FUNCTION 2: Perform LOTUS 1-2-3 Financial or "What-If" Analysis

Discussion of the following actions is predicated on the user having a well developed understanding of the LOTUS 1-2-3 system. Terminate the system and review any of several available books detailing the system's capabilities and operations before continuing if you are not familiar with that software product.

Select option 2 to begin LOTUS 1-2-3 financial or "what-if" analysis processing. Insert a LOTUS system disk in drive A (or have a product such as ZERODISK installed) to start the LOTUS system. If the Function Selection Menu is not displayed, select the "Return to " option of the current menu until the Function Selection Menu appears. If

4 ZERODISK is a software product available from Quaid Software Limited, 45 Charles Street East, Third Floor, Toronto, Ontario M4Y 1S2 (416) 961-8243. It is a product that enables users to run software applications without the need to place master disks in the "A" drive required by some programs such as dBASE III, LOTUS 1-2-3, etc.
a process is active, select the option that terminates the process. Once a menu appears, select the "Return to " option of the current menu until the Function Selection Menu appears. Insert a LOTUS system disk in drive A and then select option 2 when the Function Selection Menu appears.

Following a message concerning changing the LOTUS active file directory, the first screen of the LOTUS 1-2-3 system (Screen 9 - See Attachment 2 for screen formats) appears. The system experiences a few seconds delay for system startup.

NOTE: a backslash (\) followed by a single letter indicates a LOTUS macro. Execute a macro by simultaneously depressing the ALT and letter keys. A slash (/) followed by a letter indicates a LOTUS command. [CR] denotes the striking of the RETURN or ENTER key.

Screen 9: The opening menu of the LOTUS 1-2-3 system requires no input. Processing continues with the depression of any key.

Screen 10: The empty LOTUS 1-2-3 spreadsheet screen appears. Change the default subdirectory in LOTUS if it is not subdirectory C:\DBASEIII. Enter LOTUS command /WGDC:\DBASEIII[CR]Q to change the default subdirectory. Enter LOTUS command /FR to retrieve a file. Screen 11 appears. Use the arrow keys to point to SKELETON or MAINTORD or type either SKELETON or MAINTORD. SKELETON.WKS is the formatting file for outputs from the Pascal Configurer module. This file includes the macros developed for recalculation analysis beginning in cell A200. MAINTORD is the formatting file for outputs from the Maintenance Delivery Order Generation module executed from within the dBASE Configuration Management System. This file includes macros similar to those beginning in cell A200 of file SKELETON.WKS. If the user selects the SKELETON worksheet, Screen 12 - the formatted spreadsheet, appears.

Screen 13: Enter the LOTUS command /FIN{file name} or the macro \F{file name} to begin the importation process. Enter an output file name generated by the Pascal Configurer module. It may either be typed in without the "PRN" extension or selected by pointing to the file name with the arrow keys.

No further screens for the LOTUS processes are shown here. All screens appear the same, showing different views of the memory resident spreadsheet.
The following LOTUS macros in file SKELETON.WKS have been provided for easier processing:

a. \C - Changes column numeric entries to currency. Execute the macro anywhere in the worksheet.

b. \D - Deletes indicated rows. Place the cursor at the first row to delete before entering \D. Point to the last row to delete using the arrow keys.

c. \E - Deletes all ".PRN" files. Execute the macro anywhere in the worksheet.

d. \F - Imports a ".PRN" file at the cursor position. Execute the macro anywhere in the worksheet.

e. \I - Recalculates the Total Component Installation Price for a row. Place the cursor in the top row cell of the newly created temporary column (e.g., hardware, software, etc.). Copy subsequent entries using /C versus using \I.

f. \M - Recalculates the Total Component Purchase Price for a row. Place the cursor in the top row cell of the newly created temporary column (e.g., hardware, software, etc.). Copy subsequent entries using /C versus using \M.

g. \N - Recalculates the Total Hardware Component Maintenance Price for a row. Place the cursor in the top row cell of the newly created temporary column (e.g., hardware, software, etc.). Copy subsequent entries using /C versus using \N.

h. \O - Recalculates the Total Software Component Maintenance Price for a row. Place the cursor in the top row cell in the newly created temporary column (e.g., hardware, software, etc.). Copy subsequent entries using /C versus using \O.

i. \P - Prepares the worksheet for output to the dBASE process. Execute the macro anywhere in the worksheet.
APPENDIX A: USER's MANUAL

j. \R - Names a macro. Execute the macro in the cell of the new macro identifier.

k. \S - Sum indicated columns. Execute the macro from the cell where the total figure is desired. Use arrow keys, followed by the RETURN or ENTER key, to indicate the beginning and end of the summary area.

l. \T - Recalculates the Component Downtime hourly rate. Place the cursor in the top row cell in the newly created temporary column (e.g., hardware, software, etc.). Copy subsequent entries using /C versus using \T.

m. \U - Recalculates the Component System Downtime hourly rate. Place the cursor in the top row cell in the newly created temporary column (e.g., hardware, software, etc.). Copy subsequent entries using /C versus using \U.

Perform formula recalculation one column at a time starting from the left. Insert a new column to the left of the Total Purchase Price, Total Component Maintenance, Total Installation Price, and two Downtime Credit columns. Execute the \M, \N, \O, \I, \T, and \U macros described above in the first entry of each applicable column. Copy the resulting formula down the remainder of the column. Sum the column using the \S macro. When results are satisfactory, move (/M) the new column over the old column and delete (/WDC) the now blank column. Re-sum (no macro provided) the summary financial data at the bottom of the spreadsheet.

Perform "what-if" analysis, using the macros provided, following formula recalculation. **Exercise extreme care when changing component quantities!** If component quantity changes are made, print and review the proposed changes. After reviewing the changes, reverify the accuracy of the changes using the Configurer system. Use the Configurer to ensure that all configuration rules are properly followed.

Save an archival copy of the worksheet with the /FS{file name} command. Print a delivery order with the /PP command. Strip off the worksheet headers, non-hardware and software line items, section cost totals, summary notes and cost information with the \D macro. Print the remaining contents of the spreadsheet (less macros) with the /PF{file name} command or \P macro.
Terminate 1-2-3 by entering the LOTUS command `/QY[CR]`. The system returns to the Function Selection Menu (Screen 1) to await the next selection.

The following processing is accomplished if the file MAINTORD is selected. The system automatically loads the NEWDO.PRN file created from the dBASE III Maintenance Delivery Order Generation module. The cursor moves to the appropriate field to accept entry of the effective date. Use the macros stored at location A200 to verify and complete the maintenance delivery order following entry of the effective date.

The following LOTUS macros on MAINTORD.WKS have been provided for easier processing:

a. \C - Copies header information.

b. \D - Deletes the first column.

c. \O - Automatically imports the maintenance delivery order called NEWDO.PRN.

d. \I - Adds rows for software headers.

e. \N - Recalculates the Total Hardware Component Maintenance Price for a row. Place the cursor in the top row cell in the newly created temporary column (i.e., hardware and software). Copy subsequent entries using /C versus using \N.

f. \O - Recalculates the Total Software Component Maintenance Price for a row. Place the cursor in the top row cell in the newly created temporary column (i.e., hardware and software, etc.). Copy subsequent entries using /C versus using \O.

g. \R - Names a macro. Execute the macro in the cell of the new macro identifier.

h. \S - Sum indicated columns. Execute the macro from the cell where the total figure is desired. Use arrow keys, followed by the RETURN or ENTER key, to indicate the beginning and end of the summary area.
APPENDIX A: USER's MANUAL

Locate the first software item in the body of the spreadsheet (feature number between 510101 and 660101, 860101 or 860201). Move the cursor to the corresponding location in column "A". Execute the \I macro to insert blank rows at the location. When complete, move the cursor down 4 rows and execute the /C LOTUS command to copy headers to the beginning of the next section.

Verify the calculated Component Factored Maintenance cell for each data entry. Move the cursor to the first entry in the hardware section of the Component Factored Maint column and execute the command /WIC[CR]. This will add an additional column to the spreadsheet. Execute macro \N to automatically recalculate the maintenance amount at the first hardware component cell. Execute the LOTUS command /C[CR]{DOWN}.{DOWN to the end of the hardware column}{CR}. This copies the formula in the first cell to all following cells. Use the \S macro to sum the column and copy the same formula to the next cell to the right with the /C LOTUS command.

Comparison of these two sums may show minor rounding differences. Use the /M command to move the desired cells one column to the right to retain the LOTUS figure. Use the same procedure in the software section, substituting the \O macro for the \N macro. Delete the unnecessary column with the /WDC command following the movement of the data to the newly created column.

When validation of all entries is complete, manually enter financial appropriation data and end of delivery order comments. Manually recalculate a new System Downtime Credit Factor value using data supplied on the spreadsheet plus the installation cost. Save or print the new delivery order, as desired.

Terminate LOTUS 1-2-3 by executing the LOTUS command /QY[CR]. The system returns to the Function Selection Menu (Screen 1) to await the next selection.

2.3.3 FUNCTION 3: Execute the dBASE III Configuration Management System

Select menu option 3 (from the Function Selection Menu - Screen 1) to invoke the dBASE III Configuration Management System. If the Function Selection Menu is not displayed, select the "Return to " option of the current menu until the Function Selection Menu appears. If a
process is active, select the option that terminates the process. Once a menu appears, select the "Return to" option of the current menu until the Function Selection Menu appears. Next select menu option 3. The first screen of the dBASE III Configuration Management System (the Process Selection Menu - Screen 14) appears.

Screen 14: Ten menu options (0 - 9) exist. Processing continues based on the selection entered. Option 0 returns the system to the Function Selection Menu (Screen 1). The remaining options are discussed in order.

2.3.3.1 Load a new Delivery Order into the Configuration Management System.

Select menu option 1 (from the Function Selection Menu - Screen 1) to load a new delivery order generated by the SPLICE Configurer. The Delivery Order Load Menu (Screen 15) appears. Next select menu option 1 to commence the loading process for the new delivery order.

Screen 15: Select one of two options: 1 - load a new delivery order or 2 - return to the Process Selection Menu (Screen 14).

Screen 16: Enter the LOTUS output file name. A file name may be from one to eight alphanumeric characters long. The default file name supplied by the system is "SPLICE.PRN". The system automatically provides the extension. If the file name entered cannot be found on the default subdirectory, re-enter a valid name. An error message appears on the status line if the file name entered cannot be found. After three invalid entries, either exit the program or supply another file name. When a valid file name is supplied, enter the effective date for the delivery order.

Valid dates range from 840101 to 991231 (the system currently will not accept leap year dates - 29 February). The actual site number from the input delivery order appears following the entry of a valid date. The user may change the site number to any site number within the range 01 - 58 or accept the site number displayed. Following the entry of a valid site number, accept all data entries before the load process begins. If the response is "N", all data entries are erased and the input process is repeated. If the response is "Y", indicate input file disposition: 1 - retain or 2 - erase.
APPENDIX A: USER's MANUAL

The update process commences following this response. The load process may take up to 10 minutes. This is primarily due to the building of serial number records for each individual component on the delivery order. **BE PATIENT.** During the load process, status messages appear to keep the user apprised of the transactions as they occur. When the load process finishes, indicate whether to load another delivery order. If the response is "Y", the process starts with a new Screen 16. If the response is "N", the system returns to the Delivery Order Load Menu (Screen 15). Select menu option 2 to return to the Process Selection Menu (Screen 14) to await the next selection.

2.3.3.2 Perform maintenance on the Equipment File.

Select menu option 2 (from the Process Selection Menu - Screen 14) to either modify or review records in the Equipment File. Following the selection of option 2, the Equipment Maintenance Selection Menu (Screen 17) appears.

Screen 17: The Equipment Maintenance Selection Menu enables the user to review or modify selected entries in the Equipment File. Select one of three options: 1 - update price information; 2 - review equipment file entries; or 3 - return to the Process Selection Menu (Screen 14).

2.3.3.2.1 Modify an Equipment File Record.

Select menu option 1 (from the Equipment Maintenance Selection Menu - Screen 17) to modify an Equipment File record. Following the selection of option 1, the Equipment Update Format screen (Screen 18) appears.

Screen 18: Enter the site number to update, an integer from 01 to 58.

Select one of three options: 1 - update a specific site's records; or 2 - start at the beginning of the Equipment File; or 3 - start at the end of the Equipment File viewing records until the desired record appears. Enter "00" to start viewing records at the beginning of the file. Enter "99" to start viewing records at the end of the file. Enter the site number to modify records for a specific site.
Following entry of the specified site number, two options exist: 1 - select a specific six digit feature number or 2 - start with the first feature number by entering "00  " (two zeroes followed by four spaces). Status messages appear at the bottom of the screen when reaching the first and last records. This action only occurs if a specific site is selected (a site selection other than "00" or "99").

The only authorized changes in this screen are the three price fields. An introductory window, explaining how to terminate the modification of a record field, appears following the entry of a feature number. Terminate the introductory information window by striking the RETURN or ENTER key. Changes to fields are possible one field at a time. If changes are made to any field, either accept or reject the changes. Choose either: 1 - access the next record; 2 - access the previous record; or 3 - exit the update process. On exiting, the system returns to the Equipment Maintenance Selection Menu (Screen 17).

2.3.3.2.2 Review an Equipment File Record.

Select menu option 2 (from the Equipment Maintenance Selection Menu - Screen 17) to review an Equipment File record. Following the entry of option 2, the Equipment Review Format screen (Screen 19) appears.

Screen 19: Enter the site number to review, an integer from 01 to 58.

Select one of three options: 1 - review a specific site's records; or 2 - start at the beginning of the Equipment File; or 3 - start at the end of the Equipment File viewing records until the desired record appears. Enter "00" to start viewing records at the beginning of the file. Enter "99" to start viewing records at the end of the file. Enter the site number to review records for a specific site.

Following the entry of specified site number, two options exist: 1 - select a specific six digit feature number or 2 - start with the first feature number by entering "00  " (two zeroes followed by four spaces). Status messages appear at the bottom of the screen when reaching the first and last records. This action only occurs if a specific site is selected (a site selection other than "00" or "99").
2.3.3.3 Perform Maintenance on the Equipment Description File.

Select menu option 3 (from the Process Selection Menu - Screen 14) to either modify or review records in the Equipment Description File. Following the selection of option 3, the Equipment Description Maintenance Menu (Screen 20) appears.

Screen 20: The Equipment Description Maintenance Menu enables the user to review or modify selected entries in the Equipment Description File. Select one of three options: 1 - modify Equipment Description File entries; 2 - review Equipment Description File entries; or 3 - return to the Process Selection Menu (Screen 14).

2.3.3.3.1 Modify an Equipment Description File Record.

Select menu option 1 (from the Equipment Maintenance Selection Menu - Screen 17) to modify an Equipment Description File record. After the selection of option 1, the Description Update Format screen (Screen 21) appears.

Screen 21: Enter: 1 - "00" (two zeroes followed by four spaces) to start the update process at the top of the file; 2 - "99" (two nines followed by four spaces) to start at the update process the end of the file; or 3 - a six digit feature number. Valid feature numbers range from 000101 to 994001.

An introductory window, explaining how to terminate the modification of a record field, appears following the entry of a feature number. Terminate the introductory information window by striking the RETURN or ENTER key. Changes to fields are possible one field at a time.

All data entries in this screen may be modified. Once the Base Maintenance Price field is either modified or passed, the user may update the memo field. If the response is "Y", a window of instructions (Screen 22) appears. The
instructions describe how to make changes to the memo field. If the response is "N", processing continues.

Accept or reject changes made to any field. Choose either: 1 - access the next record; 2 - access the previous record; or 3 - exit the update process. On exiting, the system returns to the Equipment Maintenance Selection Menu (Screen 17).

2.3.3.3.2 **Review an Equipment Description File Record.**

Select menu option 2 (from the Equipment Maintenance Selection Menu - Screen 17) to review an Equipment Description File record. After the selection of option 2, the Description Review Format screen (Screen 23) appears.

**Screen 23:** Enter either: 1 - "00" *(two zeroes followed by four spaces)* to start the update process at the top of the file; 2 - "99" *(two nines followed by four spaces)* to start at the update process the end of the file; or 3 - a six digit feature number. Valid feature numbers range from 000101 to 994001.

No changes are allowed in this screen. Choose either: 1 - access the next record; 2 - access the previous record; or 3 - exit the review process. On exiting, the system returns to the Equipment Maintenance Selection Menu (Screen 17).

2.3.3.4 **Perform Maintenance on the Site Name File.**

Select menu option 4 (from the Process Selection Menu - Screen 14) to either modify or review records in the Site Name File. Following the selection of option 4, the Site Name Maintenance Menu (Screen 24) appears.

**Screen 24:** The Site Name Maintenance Menu enables the user to review or modify selected entries in the Site Name File. Select one of three options: 1 - modify Site Name File entries; 2 - review Site Name File entries; or 3 - return to the Process Selection Menu (Screen 14).
2.3.3.4.1 **Modify a Site Name File Record.**

Select menu option 1 (from the Site Name Maintenance Menu - Screen 24) to modify a Site Name File record. After the selection of option 1, the Site Address Data Update Format screen (Screen 25) appears.

**Screen 25:** Enter the site number to update, an integer from 01 to 58.

Select one of three options: 1 - update a specific site's records; or 2 - start at the beginning of the Site Name File; or 3 - start at the end of the Site Name File viewing records until the desired record appears. Enter "00" to start viewing records at the beginning of the file. Enter "99" to start viewing records at the end of the file. Enter the site number to modify records for a specific site.

All data entries, except site number and type activity, may be changed. An introductory window, explaining how to terminate the modification of a record field, appears following the entry of a feature number. Terminate the introductory information window by striking the RETURN or ENTER key. Changes to fields are possible one field at a time. Accept or reject changes made to any field. Choose either: 1 - access the next record; 2 - access the previous record; or 3 - exit the update process. On exiting, the system returns to the Site Name Maintenance Menu (Screen 24).

2.3.3.4.2 **Review a Site Name File Record.**

Select menu option 2 (from the Site Name Maintenance Menu - Screen 24) to review a Site Name File record. Following the selection of option 2, the Site Address Data Review Format screen (Screen 26) appears.

**Screen 26:** Enter the site number to review, an integer from 01 to 58.

Select one of three options: 1 - review a specific site's records; or 2 - start at the beginning of the Site Name File; or 3 - start at the end of the Site Name File viewing records until the desired record appears. Enter "00" to start viewing records at the beginning of the file.
Enter "99" to start viewing records at the end of the file. Enter the site number to review records for a specific site.

No data entries may be changed in this screen. Choose either: 1 - access the next record; 2 - access the previous record; or 3 - exit the update process. On exiting, the system returns to the Site Name Maintenance Menu (Screen 24).

2.3.3.5 Perform Maintenance on the Manual File.

Select menu option 5 (from the Process Selection Menu - Screen 14) to either modify or review records in the Manual File. Following the selection of option 5, the Manual Maintenance Menu (Screen 27) appears.

Screen 27: The Manual Maintenance Menu enables the user to either access, modify, add or delete selected entries in the Manual File. Select one of five options: 1 - add a new Manual Description entry; 2 - update Manual Description entries; 3 - delete a Manual Description entry; 4 - review Manual Description entries; or 5 - return to the Process Selection Menu (Screen 14).

2.3.3.5.1 Add a new Manual Description entry.

Manual description entries may only be added for the site selected. The site number and feature number must be known to successfully execute this process. This restriction applies even if a manual description already exists for a site and feature number. Be sure you want to add a new manual and not just update an existing one! Delete an old manual if it is no longer applicable.

Screen 28: Enter a valid site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site number selected.

Enter the feature number for the manual description to add. Valid feature numbers range from 000101 to 994001. The system validates the feature number to ensure that the feature number exists on the file. Once a valid feature number is entered, the CLIN and description data appear. The cursor moves to the Manual Description field where the new manual description is entered. Indicate whether the new description is acceptable. If the response is "N", either choose to continue or exit. If the response is "Y", the new
description entered is accepted. Choose either to continue or exit. On exiting, the system returns to the Manual Maintenance Menu (Screen 27).

2.3.3.5.2 Update a Manual Description entry.

Select menu option 2 (from the Manual Maintenance Menu - Screen 27) to modify a Manual File record. After the selection of option 2, the Manual Update Format screen (Screen 29) appears.

Screen 29: Enter a valid site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Select one of three options: 1 - update a specific site's records; or 2 - start at the beginning of the Manual File; or 3 - start at the end of the Manual File viewing records until the desired record appears. Enter "00" to start viewing records at the beginning of the file. Enter "99" to start viewing records at the end of the file. Enter the site number to modify records for a specific site.

Following entry of the specified site number, two options exist: 1 - select a specific six digit feature number or 2 - start with the first feature number by entering a feature number of "00 " (two zeroes followed by four spaces). Status messages appear at the bottom of the screen when reaching the first and last records. This action only occurs if a specific site is selected (a site selection other than "00" or "99").

The only field allowed to be modified during this process is the Manual Description field. An introductory window, explaining how to terminate the modification of a record field, appears following the entry of a feature number. Terminate the introductory information window by striking the RETURN or ENTER key.

Changes to fields are possible one field at a time. Accept or reject changes made to any field. Choose either: 1 - access the next record; 2 - access the previous record; or 3 - exit the update process. On exiting, the system returns to the Manual Maintenance Menu (Screen 27).
2.3.3.5.3 **Delete a Manual Description entry.**

Select menu option 3 (from the Manual Maintenance Menu - Screen 27) to delete a Manual Description entry. After the selection of option 3, the Manual Deletion Format screen (Screen 30) appears.

**Screen 30:** Enter a valid site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Enter the feature number for the description to delete. Valid feature numbers range from 000101 to 994001. When the description appears, verify the deletion decision. If the response is "N", the Manual Description is left intact. If the response is "Y", the Manual Description is deleted. Choose either to continue or exit. On exiting, the system returns to the Manual Maintenance Menu (Screen 27).

2.3.3.5.4 **Review a Manual Description entry.**

Select menu option 4 (from the Manual Maintenance Menu - Screen 27) to review a Manual Description entry. After the selection of option 4, the Manual Review Format screen (Screen 31) appears.

**Screen 31:** Enter a site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Select one of three options: 1 - review a specific site's records; or 2 - start at the beginning of the Manual File; or 3 - start at the end of the Manual File viewing records until the desired record appears. Enter "00" to start viewing records at the beginning of the file. Enter "99" to start viewing records at the end of the file. Enter the site number to review records for a specific site.

No data entries may be changed in this screen. Choose either: 1 - access the next record; 2 - access the previous record; or 3 - exit the update process. On exiting, the system returns to the Site Name Maintenance Menu (Screen 24).
2.3.3.6 **Perform Maintenance on the Serial Number File.**

Select menu option 6 (from the Process Selection Menu - Screen 14) to either modify or review records in the Serial Number File. Following the selection of option 6, the Serial Number Maintenance Menu (Screen 32) appears.

Three data elements must be known to perform an update on a Serial Number File record. The three data elements are: 1 - site number, 2 - effective delivery order date and 3 - feature number of the serial number to be modified. If all three or any of these data elements are not known, run a date level report to obtain the three elements (refer to the section Generate REPORTS for the Project, a Site or Equipment for specific procedures).

2.3.3.6.1 **Modify a Serial Number File record.**

Select menu option 1 (from the Serial Number Maintenance Selection Menu - Screen 32) to modify a Serial Number File record. After the selection of option 1, the Serial Number Update Format screen (Screen 33) appears.

**Screen 33:** Enter a valid site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Following the site number entry, enter an effective delivery order date. Three attempts are allowed to specify an effective delivery order date. Screen 34 appears if on the third attempt a valid effective delivery order date is not entered. Select one of two choices: 1 - continue with the update process or 2 - exit the update process and obtain the three elements (refer to the section Generate REPORTS for the Project, a Site or Equipment for specific procedures).

Once a delivery order date is entered, enter a valid feature number. Valid feature numbers range from 000101 to 994001. Screen 34 appears if all three data elements do not match any record data fields for the site selected. The same two choices described in the paragraph above may be chosen. When a valid feature number is entered and all three data elements match, a short introductory window explaining how to terminate the modification of a record.
field appears. Terminate the introductory information window by striking the RETURN or ENTER key.

Following termination of the introductory information screen, the Serial Number File record selected appears. The only field that may be modified is the serial number field. Accept or reject changes made to the serial number field. If the response is "y", the change is made to the database. If the response is "N", the change is not accepted. Choose either: 1 - access the next record; 2 - access the previous record; or 3 - exit the update process. On exiting, the system returns to the Serial Number Maintenance Menu (Screen 32).

2.3.3.6.2 Review a Serial Number File record.

Select menu option 2 (from the Serial Number Maintenance Selection Menu - Screen 32) to review a Serial Number File record. After the selection of option 2, the Serial Number Update Format screen (Screen 35) appears.

Screen 35: Enter a valid site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Select one of three options: 1 - review a specific site's records; or 2 - start at the beginning of the Serial Number File; or 3 - start at the end of the Serial Number File viewing records until the desired record appears. Enter "00" to start viewing records at the beginning of the file. Enter "99" to start viewing records at the end of the file. Enter the site number to review records for a specific site.

Following entry of the specified site number, two options exist: 1 - select a specific six digit feature number or 2 - start with the first feature number by entering "00 " (two zeroes followed by four spaces). Valid feature numbers range from 000101 to 994001. Status messages appear at the bottom of the screen when reaching the first and last records. This action only occurs if a specific site is selected (a site selection other than "00" or "99").

No data fields are allowed to be modified during the review process. Choose either: 1 - access the next record; 2 - access the previous record; or 3 - exit the review.
process. On exiting, the system returns to the Serial Number Maintenance Menu (Screen 32).

2.3.3.7 Generate REPORTS for the Project, a Specific Site or a Particular Date.

Select menu option 7 (from the Process Selection Menu - Screen 14) to obtain an overall project report, a report for a particular site or a report for a delivery order issued on a particular site. Following the selection of option 7, the Report by Type Menu (Screen 36) appears.

Screen 36: Various levels of reports which may be selected appear. Select one of three options: 1 - obtain a project level report; 2 - obtain a site specific report; 3 - obtain a delivery order specific report; or 4 - return to the Process Selection Menu (Screen 14).

Screen 37: When obtaining any of the various types of reports, two options exist: 1 - obtain a printed report or 2 - view the data on screen. Screen 37 always appears if a printed report is selected. Ensure: 1 - the power to the printer is on; 2 - sufficient paper is loaded in the printer and 3 - the leading edge of the paper is positioned with the printer's typing line alignment mark. After all three conditions are satisfied, commence printing by the striking the RETURN or ENTER key. Once printing commences, the appropriate screen appears and status messages detailing the progress of the report are displayed.

2.3.3.7.1 Obtain an Overall Project Level Report.

Select menu option 7 (from the Process Selection Menu - Screen 14) to obtain an overall project level report for a site. The Report by Type Menu (Screen 36) appears. From the Report by Type Menu, select option 1. After the selection of option 1, the Project Level Reports Menu (Screen 38) appears.

Screen 38: Select one of three options: 1 - obtain a report by equipment type; 2 - obtain a report by serial numbers; or 3 - return to the Report by Type Menu (Screen 36).
2.3.3.7.1.1 Obtain an Overall Project Report by Equipment Type.

Select menu option 1 (from the Project Level Reports Menu - Screen 38) to obtain an overall project report broken down by type of equipment. After the selection of option 1, the Equipment Project Level Report screen (Screen 39) appears.

Screen 39: Indicate whether a printed report is desired. Responses are either "Y" or "N". If the response is "Y", Screen 37 appears before printing commences (See the discussion of Screen 37 above). If the response is "N", the data and associated headings appear. Screen 40 is a sample report format.

Screen 40: All equipment is totaled by feature number and presented. The quantity for each feature number displayed represents the total quantity ordered for all sites in the Equipment database. After each screen, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Project Level Reports Menu (Screen 38).

2.3.3.7.1.2 Obtain an Overall Project Report by Serial Number.

Select menu option 2 (from the Project Level Reports Menu - Screen 38) to obtain an overall project report broken down by serial number. After the selection of option 2, the Equipment Serial Number Project Level Report screen (Screen 41) appears.

Screen 41: Indicate whether a printed report is desired. Responses are either "Y" or "N". If the response is "Y", Screen 37 appears before printing commences (See the discussion of Screen 37 above). If the response is "N", the data and associated headings appear. Screen 42 is a sample report format.

Screen 42: All serial numbers for each component at all sites are presented. This will probably be a LARGE report! Entries include: Site Number, CLIN, Feature Number, Description, Effective Delivery Order Date, total component quantity on the delivery order, specific component number (e.g. 1 of 9), and the applicable serial number. After each
screen, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Project Level Reports Menu (Screen 38).

2.3.3.7.2 Obtain a Report for a Particular Site.

Select menu option 7 (from the Process Selection Menu - Screen 14) to obtain a report for a particular site. The Report by Type Menu (Screen 36) appears. Select menu option 2 from the Report by Type Menu. After the selection of option 2, the Site Level Reports Menu (Screen 43) appears.

Screen 43: Select one of four options: 1 - obtain a report by equipment type; 2 - obtain a report of site manuals; 3 - obtain a report by serial number; or 4 - return to the Site Level Reports Menu (Screen 43).

2.3.3.7.2.1 Obtain a Site Specific Report by Equipment Type.

Select menu option 1 (from the Site Level Reports Menu - Screen 43) to obtain a site specific report broken down by equipment type. After the selection of option 1, the Equipment Site Level Report screen (Screen 44) appears.

Screen 44: Enter the site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Screen 45: Indicate whether a printed report is desired. Responses are either "Y" or "N". If the response is "Y", Screen 37 appears before printing commences (See the discussion of Screen 37 above). If the response is "N", the data and associated headings appear. Screen 46 is a sample report format.

Screen 46: All records for a specific site are selected from the Equipment database and their quantities are totaled. The Site Number, CLIN, Feature Number, Equipment Description, and total site quantity are presented. After each screen, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Site Level Reports Menu (Screen 42).
2.3.3.7.2.2 Obtain a Site Specific Report of Manuals.

Select menu option 2 (from the Site Level Reports Menu - Screen 43) to obtain a site specific manual report. After the selection of option 2, the Site Level Manual Report screen (Screen 47) appears.

Screen 47: Enter the site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Screen 48: Indicate whether a printed report is desired. Responses are either "Y" or "N". If the response is "Y", Screen 37 appears before printing commences (See the discussion of Screen 37 above). If the response is "N", the data and associated headings appear. Screen 49 is a sample report format.

Screen 49: The Manual File is accessed and each feature number within the selected site appears. Report items include Site Number, CLIN, Feature Number, Description, and Manual Description. After each screen, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Site Level Reports Menu (Screen 42).

2.3.3.7.2.3 Obtain a Site Specific Report by Serial Number.

Select menu option 3 (from the Site Level Reports Menu - Screen 43) to obtain a site specific report of serial numbers. After the selection of option 3, the Site Serial Number Report screen (Screen 50) appears.

Screen 50: Enter the site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Screen 51: Indicate whether a printed report is desired. Responses are either "Y" or "N". If the response is "Y", Screen 37 appears before printing commences (See the discussion of Screen 37 above). If the response is "N", the data and associated headings appear. Screen 52 is a sample report format.
APPENDIX A: USER's MANUAL

Screen 52: All serial numbers for each component at a site appear. Entries include: Site Number, CLIN, Feature Number, Description, Effective Delivery Order Date, total component quantity on the delivery order, specific component number (e.g. 1 of 9), and the applicable serial number. After each screen, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Site Level Reports Menu (Screen 42).

2.3.3.7.3 Obtain a Report for a Delivery Order Issued on a Particular Date.

Select menu option 7 (from the Process Selection Menu - Screen 14) to obtain a report for a delivery order issued on a particular date. The Report by Type Menu (Screen 36) appears. From the Report by Type Menu, select option 3. After the selection of option 3, the Delivery Order Date Level Reports Menu (Screen 53) appears.

Screen 53: Select one of four options: 1 - obtain an equipment report with unit costs; 2 - obtain an equipment report without costs; 3 - obtain a report by serial number; or 4 - return to the Delivery Order Date Level Reports Menu (Screen 53).

2.3.3.7.3.1 Obtain a Report by Equipment Type with Unit Prices.

Select menu option 1 (from the Delivery Order Date Level Reports Menu - Screen 53) to obtain a date level report broken down by equipment type with unit prices. After the selection of option 1, the Delivery Order Level Report screen (Screen 54) appears.

Screen 54: Enter the site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Screen 55: All delivery orders existing in the Equipment database appear. After the last delivery order effective date appears, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Delivery Order Date Level Reports Menu (Screen 53).

Screen 56: Indicate whether a printed report is desired. Responses are either "Y" or "N". If the response...
is "Y", Screen 37 appears before printing commences (See the discussion of Screen 37 above). If the response is "N", the data and associated headings appear. Screen 57 is a sample report format.

Screen 57: Items appearing on the report are Delivery Order Date, Site Number, CLIN, Feature Number, Description, Delivery Order Quantity, and Component Unit Purchase Price. After each screen, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Delivery Order Date Level Reports Menu (Screen 53).

2.3.3.7.3.2 Obtain a Report by Equipment Type without Unit Prices.

Select menu option 2 (from the Delivery Order Date Level Reports Menu - Screen 53) to obtain a date level report broken down by equipment type without unit prices. After the selection of option 2, the Delivery Order Level Report screen (Screen 58) appears.

Screen 58: Enter the site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

Screen 59: All delivery orders existing in the Equipment database appear. After the last delivery order effective date appears, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Delivery Order Date Level Reports Menu (Screen 53).

Screen 60: Indicate whether a printed report is desired. Responses are either "Y" or "N". If the response is "Y", Screen 37 appears before printing commences (See the discussion of Screen 37 above). If the response is "N", the data and associated headings appear. Screen 61 is a sample report format.

Screen 61: Items appearing on the report are Delivery Order Date, Site Number, CLIN, Feature Number, Description, Delivery Order Quantity, and FDC Model Number. After each screen, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Delivery Order Date Level Reports Menu (Screen 53).
2.3.3.7.3.3 *Obtain a Date Level Report by Serial Number.*

Select menu option 3 (from the Delivery Order Date Level Reports Menu - (Screen 53) to obtain a date level report broken down by serial number. After the selection of option 3, the Site Serial Number Report screen (Screen 62) appears.

**Screen 62:** Enter the site number, an integer from 01 to 58. The site number entered is validated to ensure that records exist for the site selected.

**Screen 63:** All delivery orders existing in the Equipment database appear. After the last delivery order effective date appears, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Delivery Order Date Level Reports Menu (Screen 53).

**Screen 64:** Indicate whether a printed report is desired. Responses are either "Y" or "N". If the response is "Y", Screen 37 appears before printing commences (See the discussion of Screen 37 above). If the response is "N", the data and associated headings appear. Screen 65 is a sample report format.

**Screen 65:** Items appearing on the report are Delivery Order Date, Site Number, CLIN, Feature Number, Description, Delivery Order Effective Date, Total Quantity by Component ordered on the delivery order, specific component quantity (e.g. 1 of 9), and Item Serial Number. After each screen, choose either: 1 - continue the report or 2 - exit the report process. On exiting, the system returns to the Delivery Order Date Level Reports Menu (Screen 53).

2.3.3.8 *Generate a Maintenance Delivery Order for a SPLICE Site.*

Select menu option 8 (from the Process Selection Menu - Screen 14) to generate a maintenance delivery order for a SPLICE site. Following the selection of option 8, the Maintenance Delivery Order Generation Program screen (Screen 66) appears.

**Screen 66:** Enter the following data: 1 - Site Number; 2 - LCN Hardware Maintenance Escalation Rate; 3 - LCN
Software Escalation Maintenance Rate; 4 - SPLICENet Maintenance Discount Rate; and 5 - Site Maintenance Escalation Rate. After these data elements are entered, choose either: 1 - continue or 2 - exit the process. If the response is "Y", the maintenance delivery order generation process is initiated and takes approximately 10 minutes to complete. The output file generated is always "NEWDO.PRN". On completion of the generation process, the system returns to the Process Selection Menu (Screen 14) to await the next selection.

On returning to the Process Selection Menu, select menu option 0 to return to the Function Selection Menu (Screen 1). From the Function Selection Menu, select menu option 2. After selecting option 2, the system transfers to the LOTUS 1-2-3 environment. Refer to section 2.3.2 on page 19 to obtain the specific details for step-by-step procedures. Since this is a maintenance delivery order rather than an initial delivery order, follow the procedures which address MAINTDO worksheet execution versus SKELETON worksheet execution.

2.3.3.9 Generate Mailing Labels for all SPLICE Sites.

Select menu option 9 (from the Process Selection Menu - Screen 14) to generate mailing labels for all SPLICE sites. Following the selection of option 9, the Mailing Label Generation Program screen (Screen 67) appears.

Screen 67: The mailing label generation program simply produces mailing labels for all the SPLICE sites. Delivery order changes, contract amendments, or other SPLICE related correspondence may be mailed to all SPLICE sites without having to manually create labels. The only input required for the process is the number of copies of mailing labels desired during the run. Valid input values are from 1 to 10 copies of mailing labels. When processing is complete, control returns to the Process Selection Menu (Screen 14) to await the next selection.

This completes the discussion of the process functions of the SPLICE Configurer and dBASE III Configuration Management System. Exit the integrated system by either of two options: 1 - select Function Selection Menu option 5 to return to the dBASE III system prompt or 2 - select Function Selection Menu option 6 to return to the DOS operating system prompt.
DEVELOPMENT OF AN AUTOMATED MICROCOMPUTER KNOWLEDGE-BASED INTEGRATED CON. (U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA R L BEARD MAR 86
2.3.4 FUNCTION 4: View the on-line User's Manual

Select Function Selection Menu option 4 to view the on-line User's Manual. The system temporarily transfers control to Wordstar where a copy of the file "USERS.MAN" is viewed. Any changes made to this file during the viewing process are not retained. The file copy is destroyed on termination from Wordstar. Terminate User's Manual viewing by typing either "^KD" or "^KQ" (see note below). Either command returns the system to the Wordstar opening menu. Typing the letter "X" returns the system to the Configuration Management System.

NOTE: The commands "^KD" and "^KQ" are executed by simultaneously holding down the "CTRL" key (represented by the character ^) on the left side of the keyboard and typing the letter "K" followed by either letters "D" or "Q".

3.0 System Output.

The output from the SPLICE Pascal configurer is a formatted disk file. The file is input data for LOTUS 1-2-3, which has 3 outputs: 1 - an archival LOTUS ".WKS" disk file; 2 - a dBASE ".PRN" input disk file; and 3 - a delivery order.

The dBASE process has numerous outputs. Refer to Section 2 (Screens 36 through 66) for further information.

4.0 Exception Reports.

This integrated system is interactive, therefore, no hard copy exception reports are produced. Errorously entered data is presented to the user for immediate action or correction.

5.0 Limitations.

The SPLICE System Configurer was designed on an IBM-PC, but is intended to be run on an IBM-PC AT. The designers recommend that the target IBM-PC AT have the maximum user memory allowed (640KB). To run the dBASE Configuration Management System, a hard disk is mandatory. The system requires a 132 column printer to print delivery orders.
APPENDIX A: USER'S MANUAL

generated from both LOTUS 1-2-3 and dBASE processes and mailing labels.

If a system other than an IBM-PC/AT is used, the system will respond slowly. Further performance degradation will occur while importing the ".PRN" file into LOTUS. Performance degradation will also occur during the Serial Number building process in the file load and in the maintenance delivery order generation process.

256KB of memory is required if dBASE III version 1.0 is used. 384KB RAM is required if dBASE version 1.1 is used.

The SPLICE Pascal Configurer system is limited by the number of components it can configure (200) and the number of sites it can configure (58).

The LOTUS 1-2-3 and dBASE III modules exhibit only those limitations which exist for those "off-the-shelf" packages.

6.0 Command Sequence.

Issue the command SPLICE (ex: C>SPLICE) from the DOS command prompt to invoke the SPLICE integrated configuration system (Pascal Configurer and dBASE Configuration Management System). This directs DOS to process a command batch file named SPLICE. The command batch file issues all required commands and causes the integrated system to load the memory resident module FLASHUP and commence integrated system execution (See Section 2 for more detailed entries).

NOTE: Prior to issuing the command SPLICE, deactivate any resident color enhancement programs (ex: KOLOR.COM). Such programs interfere with the screen colors generated by the system and data entry color attributes.

7.0 Who to Call.

If program malfunctions occur or questions related to the system arise, contact LCDR E. J. Case, SC, USN, phone number (408) 384-8204 or LCDR R. L. Beard III, SC, USN, phone number (408) 646-1982.
APPENDIX A: USER'S MANUAL

SELECTION CRITERIA FOR A SPLICE CONFIGURATION

SITE NAME: ________________________________

SITE NUMBER: ____________________________

DISCOUNT/ESCALATION RATES:

FDC SNA Interface discount rate: ____________

NON-LCN PURCHASE discount rate: ____________

LCN PURCHASE discount rate: ________________

SPLICENet Software Maintenance discount rate: ____________

SPLICENet Software Purchase discount rate: ____________

EMERGENCY MAINTENANCE escalation rate: ____________

LCN HARDWARE MAINTENANCE escalation rate: ____________

LCN SOFTWARE MAINTENANCE escalation rate: ____________

INSTALLATION escalation rate: ________________

TRAINING escalation rate: _________________

DOCUMENTATION escalation rate: _____________

MAINTENANCE escalation rate from SPLICE contract: ____________

Output File Name: ____________________________ PRN

Number of MAINTENANCE MONTHS for this order: ____________

Effective Delivery Order Date: ____________________________

(MM / DD / YY)
APPENDIX A: USER's MANUAL

HARDWARE SELECTIONS:

PROCESSORS recommended by FMSO Sizing Study: ______________________

CENTRONICS PRINTERS to be ordered: ______________________

TANDEM CRTS to be ordered: ______________________

128MB DISCs FMSO Sizing Study recommended, EVEN No.: __________

240MB DISCs FMSO Sizing Study recommended, EVEN No.: __________

540MB DISCs FMSO Sizing Study recommended, EVEN No.: __________

Non-6100 ASYNC Controllers to be installed: ______________________

Non-6100 ASYNC EXTENSION BOARDS to be installed per controller (0/1/2): ______________________

BIT SYNC LINES to be supported: ______________________

BYTE SYNC LINES to be supported: ______________________

TRI-DENSITY TAPE DRIVES to be installed: ______________________

(Ensure fixed disk archival back-up drives are included)

READER/PUNCHES to be installed: ______________________

CARD READERS to be installed: ______________________

1000 LPM PRINTERS to be installed: ______________________

600 LPM PRINTERS to be installed: ______________________

LCN TRUNKS required for the network: ______________________

6100 LINE INTERFACE UNITS: ______________________

PERKIN-ELMER Local Computer Network interfaces: ______________________

Burroughs B4800 Local Computer Network interfaces: ______________________

Burroughs B4900 Local Computer Network interfaces: ______________________

IBM System Local Computer Network interfaces: ______________________
APPENDIX A: USER'S MANUAL

UNIVAC System Local Computer Network interfaces: __________

FIPS Standard Local Computer Network interfaces: __________

TANDEM HYPERchannels to be installed: __________

PATCH PANEL CABINETS:
   (additional for reserve and expansion) __________

SYSTEM CABINETS:
   (additional for reserve and expansion) __________

EXPANSION CABINET(S):
   (additional for reserve and expansion) __________

HYPERchannel Adapter Cabinet(s) required: __________

Estimate the distance between the two most distant Computers on the Local Computer Network, Range - (1 to 5000 feet): __________
SOFTWARE SELECTIONS:

File Security System Software (Yes/No)?

LCN File Utility Package Software (Yes/No)?

ATP 6100 Software (Yes/No)?

BSC 6100 Software (Yes/No)?

ADCCP 6100 Software (Yes/No)?

BURROUGHS POLL/SELECT 6100 Software (Yes/No)?

SNAX and SNAX/HLS 6100 Software (Yes/No)?

TINET 6100 Software (Yes/No)?

TR 3271 Software (Yes/No)?

AM 6520 Software (Yes/No)?

T-TEXT Software (Yes/No)?

FDC SNA Interface Software (Yes/No)?

FDC DLANet Interface Software (Yes/No)?

DDN Interface Software (Yes/No)?

NETWORK MAINTENANCE FACILITY (NMF):

NMF Group Package Software (Yes/No)?

NMF Base Facility Software (Yes/No)?

NMF Performance Monitoring Software (Yes/No)?

NMF Diagnostic Monitoring Software (Yes/No)?

NMF Accounting Application Software (Yes/No)?

NETEX MAINTENANCE MONTHS for this order:

SPICENet MAINTENANCE MONTHS for this order:
APPENDIX A: USER'S MANUAL

DOCUMENTATION SELECTIONS:

COMPUTER OPERATIONS MANUAL sets required: 

SYSTEMS PROGRAMMER MANUAL sets required: 

HARDWARE MANUAL sets required: 

PROGRAMMER REFERENCE MANUAL sets required: 

TRAINING SELECTIONS:

Select Training Group to be ordered
(Group I-IV / None):

OPERATOR TRAINING COURSES required:

HARDWARE OVERVIEW COURSES required:

SYSTEMS RESOURCE MANAGEMENT COURSES required:

SYSTEMS TUNING AND XRAY COURSES required:

DATA COMMUNICATIONS COURSES required:

TAL COURSES required:

SPLICENet Migration Workshop COURSES required:

MAINTENANCE AND SITE PREP SELECTIONS:

EMERGENCY PER-CALL MAINTENANCE months required:

Should we include SITE PREPS in this run? (Yes/No): 

102
FUNCTION SELECTION MENU

Stock Point Logistics Integrated Communications Environment

SPLICE

1 - Configure a site using the SPLICE Configurer
2 - Perform financial analysis of a site using LOTUS 1-2-3
3 - Interact with the Configuration Management System
4 - Review the USER'S MANUAL
5 - Return to the DOBASE prompt
6 - Return to the DOS Operating System prompt

Please enter your choice:  

SCREEN 1

NAVAL SUPPLY SYSTEMS COMMAND

STOCK POINT LOGISTICS INTEGRATED COMMUNICATIONS ENVIRONMENT

S P L I C E

SYSTEM CONFIGURER

Version 1.1
December 1990

SCREEN 2

103
## APPENDIX A: USER's MANUAL

### Listing of Splice Sites

<table>
<thead>
<tr>
<th>Site Name</th>
<th>Site Name</th>
<th>Site Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 ASO Philadelphia</td>
<td>02 FMSO Mechanicsburg</td>
<td>03 FMSO Mechanicsburg</td>
</tr>
<tr>
<td>04 MCAS Cherry Point</td>
<td>05 MCAS El Toro</td>
<td>06 MCAF Quantico</td>
</tr>
<tr>
<td>07 MCAS Yuma</td>
<td>08 NAC Indianapolis</td>
<td>09 NARDAC Jacksonville</td>
</tr>
<tr>
<td>10 NARDAC New Orleans</td>
<td>11 NARDAC Norfolk</td>
<td>12 NARDAC Pensacola</td>
</tr>
<tr>
<td>13 NARDAC San Diego</td>
<td>14 NARDAC San Francisco</td>
<td>15 NARDAC Washington</td>
</tr>
<tr>
<td>16 NAS Barbers Point</td>
<td>17 NAS Brunswick</td>
<td>18 NAS Cecil Field</td>
</tr>
<tr>
<td>19 NAS Key West</td>
<td>20 NAEC Lake Hurst</td>
<td>21 NAS Memphis</td>
</tr>
<tr>
<td>22 NAS Miramar</td>
<td>23 NAS Oceana (Inactive)</td>
<td>24 NAS Pensacola</td>
</tr>
<tr>
<td>25 NAS Whidbey Island</td>
<td>26 NATC Patuxent River</td>
<td>27 PMTC Point Mugu</td>
</tr>
<tr>
<td>28 NAVDAF Corpus Christi</td>
<td>29 NAVDAF Great Lakes</td>
<td>30 NAVDAF Lemoore</td>
</tr>
<tr>
<td>31 NAVDAF Moffett Field</td>
<td>32 NAVDAF Orlando</td>
<td>33 NRCC Long Beach</td>
</tr>
<tr>
<td>34 NRCC Newport</td>
<td>35 NRCC Philadelphia</td>
<td>36 NRCC Washington</td>
</tr>
<tr>
<td>37 NUWES Keyport</td>
<td>38 NAVSIA Mayport</td>
<td>39 NSC Charleston</td>
</tr>
<tr>
<td>40 NSC Norfolk</td>
<td>41 NSC Oakland</td>
<td>42 NSC Pearl Harbor</td>
</tr>
<tr>
<td>43 NSC Puget Sound</td>
<td>44 NSC San Diego</td>
<td>45 NSD Guam</td>
</tr>
<tr>
<td>46 NSD Subic Bay</td>
<td>47 NSD Yokosuka</td>
<td>48 NSY Philadelphia</td>
</tr>
<tr>
<td>49 NSY Portsmouth</td>
<td>50 NTX San Diego</td>
<td>51 SPC Mechanicsburg</td>
</tr>
<tr>
<td>52 SUBASE Kings Bay</td>
<td>53 SUBASE New London</td>
<td>54 SUBASE Pearl Harbor</td>
</tr>
<tr>
<td>55 SWFPAC Bremerton</td>
<td>56 TRF Bangor</td>
<td>57 SWFLANT Kings Bay</td>
</tr>
<tr>
<td>58 TRF Kings Bay</td>
<td>59 To Be Determined</td>
<td>60 To Be Determined</td>
</tr>
<tr>
<td>61 To Be Determined</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Please select the site you desire to configure: **

### Screen 3

### Output Media and Discount/escalation Rates

<table>
<thead>
<tr>
<th>Discount &amp; Escalation Rates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Values input are added to one (1) to generate the correct discount or escalation rate</td>
</tr>
</tbody>
</table>

**FDC SNA Interface Discount Rate:**

**Non-LCN Purchase Discount Rate:**

**LCN Purchase Discount Rate:**

**SPLICEnet Software Maintenance Discount Rate:**

**SPLICEnet Software Purchase Discount Rate:**

**Emergency Maintenance Escalation Rate:**

**LCN Hardware Maintenance Escalation Rate:**

**LCN Software Maintenance Escalation Rate:**

**Installation Escalation Rate:**

**Training Escalation Rate:**

**Documentation Escalation Rate:**

**Maintenance Escalation Rate:**

### Output "PRN" File Name

**SPICE Input and Lotus 1-2-3 Input Filename:**

### Maintenance Months

**Hardware Maintenance Months:**

**Delivery Order, Effective Date:**

**Effective Date:**

### Screen 4
**APPENDIX A: USER'S MANUAL**

<table>
<thead>
<tr>
<th>SYSTEM COMPONENTS</th>
<th>HARDWARE</th>
<th>LCN INTERFACE ADAPTERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processors</td>
<td></td>
<td>PERKIN-ELMER</td>
</tr>
<tr>
<td>Centronics Printers</td>
<td></td>
<td>BURROUGHS 4800</td>
</tr>
<tr>
<td>Tandem CRIs</td>
<td></td>
<td>BURROUGHS 4900</td>
</tr>
<tr>
<td>128 MB Disks</td>
<td></td>
<td>IBM System</td>
</tr>
<tr>
<td>240 MB Disks</td>
<td></td>
<td>UNIVAC System</td>
</tr>
<tr>
<td>540 MB Disks</td>
<td></td>
<td>FIPS Standard</td>
</tr>
<tr>
<td>Non-6100 ASYNC Controllers</td>
<td></td>
<td>Tandem HYPERchannel</td>
</tr>
<tr>
<td>Non-6100 ASYNC Extension Boards</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit SYNC Lines</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Byte SYNC Lines</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tri-Density Tape Drives</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reader/Punches</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Card Readers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1000 LPM Printers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>600 LPM Printers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LCN Coaxial Cables (Trunks)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6100 Line Interface Units (LIUs)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**CABINETS**

- MAX DISTANCE BETWEEN COMPUTERS
  - (A) 1 - 500 FT
  - (B) 501 - 1000 FT
  - (C) 1001 - 1500 FT
  - (D) 1501 - 2500 FT
  - (E) 2501 - 4000 FT
  - (F) 4001 - 5000 FT

**SCREEN 5**

<table>
<thead>
<tr>
<th>SYSTEM COMPONENTS</th>
<th>HARDWARE</th>
<th>LCN INTERFACE ADAPTERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processors</td>
<td></td>
<td>PERKIN-ELMER</td>
</tr>
<tr>
<td>Centronics Printers</td>
<td></td>
<td>BURROUGHS 4800</td>
</tr>
<tr>
<td>Tandem CRIs</td>
<td></td>
<td>BURROUGHS 4900</td>
</tr>
<tr>
<td>128 MB Disks</td>
<td></td>
<td>IBM System</td>
</tr>
<tr>
<td>240 MB Disks</td>
<td></td>
<td>UNIVAC System</td>
</tr>
<tr>
<td>540 MB Disks</td>
<td></td>
<td>FIPS Standard</td>
</tr>
<tr>
<td>Non-6100 ASYNC Controllers</td>
<td></td>
<td>Tandem HYPERchannel</td>
</tr>
<tr>
<td>Non-6100 ASYNC Extension Boards</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit SYNC Lines</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Byte SYNC Lines</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tri-Density Tape Drives</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reader/Punches</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Card Readers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1000 LPM Printers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>600 LPM Printers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LCN Coaxial Cables (Trunks)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6100 Line Interface Units (LIUs)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SCREEN 6**

105
**APPENDIX A: USER's MANUAL**

### SOFTWARE RELATED INPUTS

<table>
<thead>
<tr>
<th>SOFTWARE PACKAGES</th>
<th>SOFTWARE RELATED INPUTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Security</td>
<td>FDC SNA Interface</td>
</tr>
<tr>
<td>LCN File Utility Package</td>
<td>FDC ULANET Interface</td>
</tr>
<tr>
<td>6100 Packages</td>
<td>UDN Interface</td>
</tr>
<tr>
<td>ATP</td>
<td>NMF Group</td>
</tr>
<tr>
<td>BSC</td>
<td>NMF Packages</td>
</tr>
<tr>
<td>ADCCP</td>
<td>Base Facility</td>
</tr>
<tr>
<td>Burroughs Poll Select</td>
<td>Performance Monitoring</td>
</tr>
<tr>
<td>SNAX and SNAX/MLS</td>
<td>Diagnostic Monitoring</td>
</tr>
<tr>
<td>TINET</td>
<td>Accounting Application</td>
</tr>
</tbody>
</table>

### MOUNTS OF SOFTWARE MAINTENANCE

<table>
<thead>
<tr>
<th>NETEX Maintenance Months</th>
<th>SPLICE/Net Maintenance Months</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### SCREEN 7

#### DOCUMENTATION, TRAINING & MAINTENANCE INPUTS

<table>
<thead>
<tr>
<th>DOCUMENTATION MANUALS</th>
<th>TRAINING GROUPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Operations</td>
<td>(1) Group I</td>
</tr>
<tr>
<td>Programmer Reference</td>
<td>(2) Group II</td>
</tr>
<tr>
<td>Hardware</td>
<td>(3) Group III</td>
</tr>
<tr>
<td>Systems Programmer</td>
<td>(4) Group IV</td>
</tr>
<tr>
<td></td>
<td>(5) None</td>
</tr>
</tbody>
</table>

#### TRAINING COURSES

- Operator Training
- Hardware Overview
- Systems Resource Management
- Systems Tuning and XRAY
- Data Communications
- TANDEM Application Language
- SPLICE/Net Migration Workshop

### EMERGENCY MAINTENANCE & SITE PREPARATIONS

<table>
<thead>
<tr>
<th>Months of EMERGENCY PER-CALL</th>
<th>Include Changes? (Yes or No)</th>
</tr>
</thead>
</table>

### SCREEN 8

106
SCREEN 9

A1: 

A B C D E F G

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

SCREEN 10

107
APPENDIX A: USER's MANUAL

MENU

Enter name of file to retrieve:
SKELETON MAINTORD

A     B     C     D     E     F     G

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

SCREEN 11

READY

A     B     C     D     E

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

SCREEN 12

108
APPENDIX A: USER's MANUAL

Site: 44 NSC SAN DIEGO, CA

<table>
<thead>
<tr>
<th>Contract Feature</th>
<th>Description</th>
<th>Qty</th>
<th>Unit Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>440101</td>
<td>NS-TXP, 2 MEG</td>
<td>8</td>
<td>86760.00</td>
</tr>
<tr>
<td>440102</td>
<td>2 MEG MEMORY</td>
<td>8</td>
<td>19800.00</td>
</tr>
<tr>
<td>440104</td>
<td>OSP WITH 6530</td>
<td>1</td>
<td>13387.50</td>
</tr>
<tr>
<td>440105</td>
<td>CENTRONIX PRINTER</td>
<td>2</td>
<td>1615.50</td>
</tr>
<tr>
<td>440106</td>
<td>6530 CRT</td>
<td>17</td>
<td>2317.50</td>
</tr>
<tr>
<td>440107</td>
<td>PRINTER INTERFACE</td>
<td>1</td>
<td>409.50</td>
</tr>
<tr>
<td>440108</td>
<td>PATCH PANEL CABINET</td>
<td>2</td>
<td>2250.00</td>
</tr>
</tbody>
</table>

SCREEN 13

PROCESS SELECTION MENU

Stock Point Logistics Integrated Communications Environment

SPLICE

1 - Load a new DELIVERY ORDER into the database
2 - Perform maintenance on the EQUIPMENT File
3 - Perform maintenance on the EQUIPMENT DESCRIPTION File
4 - Perform maintenance on the SITE NAME File
5 - Perform maintenance on the MANUAL File
6 - Perform maintenance on the SERIAL NUMBER File
7 - Generate REPORTS for the Project, a Site or Equipment
8 - Generate a MAINTENANCE DELIVERY ORDER for a SPLICE Site
9 - Generate MAILING LABELS for all SPLICE Sites
0 - RETURN to the Function Selection Menu

Please enter your choice: •

SCREEN 14

109
APPENDIX A: USER's MANUAL

DELIVERY ORDER LOAD MENU

[1] Load a new delivery order
[2] Return to the Main Menu

SCREEN 15

DELIVERY ORDER LOADING SELECTION MENU

LOTUS 1-2-3 output file name to load:  ********.PRN
Effective Date of the Delivery Order:  ******
YYMMDD
Site Number on the Delivery Order:  

Enter the Site Number to be loaded:  

Do you want to enter another Delivery Order? (Yes or No):  

SCREEN 16
APPENDIX A: USER's MANUAL

EQUIPMENT MAINTENANCE SELECTION MENU

[1] Modify Database Entries
[3] Return to the Main Menu

SCREEN 17

EQUIPMENT UPDATE FORMAT

Current Record #: *******

Site Number:  | Effective Date of Delivery Order:  | YYMMDD
Feature Number: | *******
CLIN Nomenclature/Description:  | ******************************************************
Quantity Ordered:  | *******

Basic Unit Cost:  | *******
Monthly Maintenance Cost:  | *******
Unit Installation Cost:  | *******

Enter N - next record, P - previous record or X - exit:  

SCREEN 18
EQUIPMENT REVIEW FORMAT

Current Record #:

Site Number:   Effective Date of Delivery Order: YYMMDD

Contract Line Item Number (CLIN):
Feature Number:
CLIN Nomenclature/Description:
Quantity Ordered:

Basic Unit Cost:
Monthly Maintenance Cost:
Unit Installation Cost:

Enter N - next record, P - previous record or X - exit:

SCREEN 19

EQUIPMENT DESCRIPTION MAINTENANCE MENU

[1] Modify Database Entries
[3] Return to the Main Menu

SCREEN 20
APPENDIX A: USER's MANUAL

DESCRIPTION UPDATE FORMAT

Current Record #: ******

Feature Number: ******
Contract Line Item Number (CLIN): ******
CLIN Nomenclature / Description: ...........................
TANDEM Model Number: ******
FDC Model Number: ******
Type of Component: ******
Base Maintenance Price: ******
Notes: Enter N - next record, P - previous record or X - exit:

SCREEN 21

EQUIPMENT DESCRIPTION EDITING/TERMINATION INFORMATION

1. To edit the NOTES field, ensure the cursor is on the word "memo" and press the <CTRL> and "PgDn" keys together.

2. To EXIT the internal editor and SAVE the changes made to the NOTES field, press the <CTRL> and "W" keys together.

3. To EXIT the internal editor WITHOUT SAVING the changes made to the NOTES field, press the <ESC> key. This will return you to the full screen mode for the record being changed.

4. To SAVE the changes made by the internal editor and return to the configuration program, press the <CTRL> and "W" keys together.

5. To return to the configuration program WITHOUT SAVING the changes made by the internal editor, press the <ESC> key.

Press ENTER to continue

SCREEN 22
APPENDIX A: USER's MANUAL

DESCRIPTION REVIEW FORMAT

Current Record #:  

Feature Number:  

Contract Line Item Number (CLIN):  

CLIN Nomenclature / Description:  

TANDEM Model Number:  

FDC Model Number:  

Type of Component:  

Base Maintenance Price:  

Notes:  

Enter N - next record, P - previous record or X - exit:

SCREEN 23

SITE NAME MAINTENANCE MENU

[1] Modify DataBase Entries
[3] Return to the Main Selection Menu

SCREEN 24
### SITE ADDRESS DATA UPDATE FORMAT

Current Record #: ****

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>***</td>
</tr>
<tr>
<td>Abbreviated Name</td>
<td>..................................................................</td>
</tr>
<tr>
<td>Commander's Title</td>
<td>..................................................................</td>
</tr>
<tr>
<td>Full Name</td>
<td>..................................................................</td>
</tr>
<tr>
<td>Address - Line 1</td>
<td>..................................................................</td>
</tr>
<tr>
<td>Address - Line 2</td>
<td>..................................................................</td>
</tr>
<tr>
<td>City</td>
<td>..................................................................</td>
</tr>
<tr>
<td>State</td>
<td>**</td>
</tr>
<tr>
<td>Zip Code</td>
<td>**</td>
</tr>
<tr>
<td>Type Activity</td>
<td>**</td>
</tr>
<tr>
<td>Maintenance Option</td>
<td>**</td>
</tr>
<tr>
<td>Maintenance Response Time</td>
<td>**</td>
</tr>
</tbody>
</table>

Enter N - next record, P - previous record or X - exit:

SCREEN 25

### SITE ADDRESS DATA UPDATE FORMAT

Current Record #: ****

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>***</td>
</tr>
<tr>
<td>Abbreviated Name</td>
<td>..................................................................</td>
</tr>
<tr>
<td>Commander's Title</td>
<td>..................................................................</td>
</tr>
<tr>
<td>Full Name</td>
<td>..................................................................</td>
</tr>
<tr>
<td>Address - Line 1</td>
<td>..................................................................</td>
</tr>
<tr>
<td>Address - Line 2</td>
<td>..................................................................</td>
</tr>
<tr>
<td>City</td>
<td>..................................................................</td>
</tr>
<tr>
<td>State</td>
<td>**</td>
</tr>
<tr>
<td>Zip Code</td>
<td>**</td>
</tr>
<tr>
<td>Type Activity</td>
<td>**</td>
</tr>
<tr>
<td>Maintenance Option</td>
<td>**</td>
</tr>
<tr>
<td>Maintenance Response Time</td>
<td>**</td>
</tr>
</tbody>
</table>

Enter N - next record, P - previous record or X - exit:

SCREEN 26
MANUAL MAINTENANCE MENU

[1] Add a new manual description
[2] Update existing description(s)
[3] Delete existing description(s)
[4] Review existing description(s)
[5] Return to the Main Menu

SCREEN 27

MANUAL ADDITION FORMAT

Current Record #: ******

Site Number:  **

Feature Number:  ******
Contract Line Item Number (CLIN):  ****
CLIN Nomenclature/Description:  ****************************
Manual Description:  ****************************

Enter N - next record, P - previous record or X - exit:  

SCREEN 28
APPENDIX A: USER's MANUAL

MANUAL UPDATE FORMAT

Current Record #: ******

Site Number: ******

Feature Number: ******
Contract Line Item Number (CLIN): ******
CLIN Nomenclature/Description: ******

Manual Description: ******

Enter N - next record, P - previous record or X - exit: ******

SCREEN 29

MANUAL DELETION FORMAT

Current Record #: ******

Site Number: ******

Feature Number: ******
Contract Line Item Number (CLIN): ******
CLIN Nomenclature/Description: ******

Manual Description: ******

Enter N - next record, P - previous record or X - exit: ******

SCREEN 30

117
MANUAL REVIEW FORMAT

Current Record #: ******

Site Number:  

Feature Number: ******
Contract Line Item Number (CLIN): ******
CLIN Nomenclature/Description:  

Manual Description:  

Enter N - next record, P - previous record or X - exit:  

SCREEN 31

SERIAL NUMBER MAINTENANCE MENU

[1] CHANGE an existing Serial Number
[2] REVIEW existing Serial Numbers
[3] Return to the Main Menu

SCREEN 32
In order for the user to be able to perform a SERIAL NUMBER update, three (3) data elements must be known:
1: The SITE NUMBER
2: The EFFECTIVE DATE of the delivery order
3: The FEATURE NUMBER of the serial number to be modified

If all three of these elements are not known, the user should terminate the update process and request a DATE LEVEL REPORT for the site number in question (Option "7" on the PROCESS SELECTION MENU followed by option "3" on the REPORT BY TYPE MENU). Any one of the three options will enable the user to view all three of the data elements needed for the Serial Number Update process. Once all three data elements have been obtained, the user can then select the Serial Number Update option.

Please select the option desired below:

[1] Continue with the Serial Number Update process.
[2] Exit the Serial Number Update process to obtain a DATE LEVEL Report and the three required data elements.
SERIAL NUMBER REVIEW FORMAT

Current Record #: ******

Site Number: ****** Effective Date of Delivery Order: ****** YYMMDD

Contract Line Item Number (CLIN): ******
Feature Number: ******
CLIN Nomenclature/Description: ****************************
Quantity Ordered: ******

Serial Number sub-record: ****** of ******

Serial Number: ******

Enter N - next record, P - previous record or X - exit: ******

SCREEN 35

REPORT BY TYPE MENU

[1] Overall PROJECT Report
[2] Report for a particular SITE
[3] Report for a DELIVERY ORDER issued on a particular date

SCREEN 36

120
ATTENTION!
1. Turn on your printer.
2. Insert paper.
3. Position to top edge.
Press ENTER to continue

SCREEN 37

PROJECT LEVEL REPORTS

[1] Report by EQUIPMENT Type
[3] Return to the Reports Level Menu

SCREEN 38
APPENDIX A: USER'S MANUAL

EQUIPMENT PROJECT LEVEL REPORT

Do you want a printed report? (Yes or No): N

Enter C to continue or X to exit:

SCREEN 39

<table>
<thead>
<tr>
<th>CLIN</th>
<th>FEATURE#</th>
<th>DESCRIPTION</th>
<th>UTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>000101</td>
<td>SITE POWER PREPARATIONS</td>
<td>2</td>
</tr>
<tr>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>22</td>
</tr>
<tr>
<td>0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td>22</td>
</tr>
<tr>
<td>0104</td>
<td>013001</td>
<td>CPU WITH 6530</td>
<td></td>
</tr>
<tr>
<td>0105</td>
<td>013101</td>
<td>CENTRONIX PRINTER</td>
<td></td>
</tr>
<tr>
<td>0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>50</td>
</tr>
<tr>
<td>0107</td>
<td>013202</td>
<td>PRINTER INTERFACE</td>
<td>2</td>
</tr>
<tr>
<td>0108</td>
<td>015001</td>
<td>PATCH PANEL CABINET</td>
<td></td>
</tr>
<tr>
<td>0109</td>
<td>015101</td>
<td>SYSTEMS CABINET</td>
<td></td>
</tr>
<tr>
<td>0110</td>
<td>015201</td>
<td>I/O POWER MODULE</td>
<td>1</td>
</tr>
<tr>
<td>0109</td>
<td>015301</td>
<td>EXPANSION CABINET</td>
<td></td>
</tr>
<tr>
<td>0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td></td>
</tr>
<tr>
<td>0113</td>
<td>016101</td>
<td>THL PATCH PANEL</td>
<td></td>
</tr>
<tr>
<td>0114</td>
<td>016201</td>
<td>ASYNC PATCH PANEL</td>
<td></td>
</tr>
<tr>
<td>0115</td>
<td>016301</td>
<td>SYNC PATCH PANEL</td>
<td></td>
</tr>
</tbody>
</table>

SCREEN 40
EQUIPMENT SERIAL NUMBER PROJECT LEVEL REPORT

Do you want a printed report? (Yes or No): N

Enter C to continue or X to exit:

SCREEN 41

<table>
<thead>
<tr>
<th>SITE CLIN</th>
<th>FEATURE#</th>
<th>DESCRIPTION</th>
<th>EFFECT DATE</th>
<th>QTY</th>
<th>QTY</th>
<th>SERIAL NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td>841127</td>
<td>9</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>01 0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td>841127</td>
<td>9</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>01 0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td>841127</td>
<td>9</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>01 0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td>841127</td>
<td>9</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>01 0104</td>
<td>013001</td>
<td>OSP WITH 6530</td>
<td>841127</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>01 0105</td>
<td>013101</td>
<td>CENTRONIX PRINTER</td>
<td>841127</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>01 0105</td>
<td>013101</td>
<td>CENTRONIX PRINTER</td>
<td>841127</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>01 0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>841127</td>
<td>25</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>01 0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>841127</td>
<td>25</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>01 0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>841127</td>
<td>25</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>01 0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>841127</td>
<td>25</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>01 0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>841127</td>
<td>25</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>01 0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>841127</td>
<td>25</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>01 0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>841127</td>
<td>25</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>01 0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>841127</td>
<td>25</td>
<td>18</td>
<td></td>
</tr>
</tbody>
</table>

Enter C to continue or X to exit: C

SCREEN 42

123
APPENDIX A: USER's MANUAL

SITE LEVEL REPORTS

[1] Report by EQUIPMENT type
[4] Return to the Reports Level Menu

SCREEN 43

EQUIPMENT SITE LEVEL REPORT

Enter site number for which the report is desired: 01

Enter C to continue or X to exit:

SCREEN 44

124
APPENDIX A: USER's MANUAL

EQUIPMENT SITE LEVEL REPORT

Do you want a printed report? (Yes or No): N

Enter C to continue or X to exit:

SCREEN 45

EQUIPMENT SITE LEVEL REPORT

<table>
<thead>
<tr>
<th>SITE</th>
<th>CLIN</th>
<th>FEATURE#</th>
<th>DESCRIPTION</th>
<th>QTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0001</td>
<td>000101</td>
<td>SITE POWER PREPARATIONS</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>13</td>
</tr>
<tr>
<td>40</td>
<td>0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td>13</td>
</tr>
<tr>
<td>40</td>
<td>0104</td>
<td>013001</td>
<td>OSP WITH 6530</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>0105</td>
<td>013101</td>
<td>CENTRONIX PRINTER</td>
<td>3</td>
</tr>
<tr>
<td>40</td>
<td>0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>25</td>
</tr>
<tr>
<td>40</td>
<td>0107</td>
<td>013202</td>
<td>PRINTER INTERFACE</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>0108</td>
<td>015001</td>
<td>PATCH PANEL CABINET</td>
<td>3</td>
</tr>
<tr>
<td>40</td>
<td>0109</td>
<td>015101</td>
<td>SYSTEMS CABINET</td>
<td>4</td>
</tr>
<tr>
<td>40</td>
<td>0110</td>
<td>015201</td>
<td>1/O POWER MODULE</td>
<td>12</td>
</tr>
<tr>
<td>40</td>
<td>0109</td>
<td>015301</td>
<td>EXPANSION CABINET</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td>5</td>
</tr>
<tr>
<td>40</td>
<td>0113</td>
<td>016101</td>
<td>THL PATCH PANEL</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>0114</td>
<td>016201</td>
<td>ASYNC PATCH PANEL</td>
<td>2</td>
</tr>
<tr>
<td>40</td>
<td>0115</td>
<td>016301</td>
<td>SYNC PATCH PANEL</td>
<td>2</td>
</tr>
</tbody>
</table>

Enter C to continue or X to exit: C

SCREEN 46
SITE LEVEL MANUAL REPORT

Enter site number for which the report is desired: 01

Enter C to continue or X to exit:

SCREEN 47

SITE LEVEL MANUAL REPORT

Do you want a printed report? (Yes or No): N

Enter C to continue or X to exit:

SCREEN 48
### SITE LEVEL MANUAL REPORT

<table>
<thead>
<tr>
<th>SITE</th>
<th>CLIN</th>
<th>FEATURE#</th>
<th>DESCRIPTION</th>
<th>MANUAL DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0104</td>
<td>013001</td>
<td>OSP WITH 6530</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0105</td>
<td>013101</td>
<td>CENTRONIX PRINTER</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0107</td>
<td>013202</td>
<td>PRINTER INTERFACE</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0108</td>
<td>015001</td>
<td>PATCH PANEL CABINET</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0109</td>
<td>015101</td>
<td>SYSTEMS CABINET</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0110</td>
<td>015201</td>
<td>I/O POWER MODULE</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0109</td>
<td>015301</td>
<td>EXPANSION CABINET</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0113</td>
<td>016101</td>
<td>THL PATCH PANEL</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0114</td>
<td>016201</td>
<td>ASYNC PATCH PANEL</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0115</td>
<td>016301</td>
<td>SYNC PATCH PANEL</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>1101</td>
<td>110101</td>
<td>DISC CONTROLLER</td>
<td></td>
</tr>
</tbody>
</table>

Enter C to continue or X to exit: C

### SCREEN 49

### SITE SERIAL NUMBER REPORT

Enter site number for which the report is desired: 01

Enter C to continue or X to exit:
SITE SERIAL NUMBER REPORT

Do you want a printed report? (Yes or No): N

Enter C to continue or X to exit:

SCREEN 51

SITE SERIAL NUMBER REPORT

<table>
<thead>
<tr>
<th>SITE CLIN</th>
<th>FEATURE#</th>
<th>DESCRIPTION</th>
<th>EFFECT DATE</th>
<th>TOT QTY</th>
<th>COMP QTY</th>
<th>SERIAL NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>40 0109</td>
<td>015301</td>
<td>EXPANSION CABINET</td>
<td>851207</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>40 0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td>851207</td>
<td>5</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>40 0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td>851207</td>
<td>5</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>40 0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td>851207</td>
<td>5</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>40 0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td>851207</td>
<td>5</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>40 0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td>851207</td>
<td>5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>40 0113</td>
<td>016101</td>
<td>THL PATCH PANEL</td>
<td>851207</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>40 0114</td>
<td>016201</td>
<td>ASYNC PATCH PANEL</td>
<td>851207</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>40 0114</td>
<td>016201</td>
<td>ASYNC PATCH PANEL</td>
<td>851207</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>40 0115</td>
<td>016301</td>
<td>SYNC PATCH PANEL</td>
<td>851207</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>40 0115</td>
<td>016301</td>
<td>SYNC PATCH PANEL</td>
<td>851207</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>40 1101</td>
<td>110101</td>
<td>DISC CONTROLLER</td>
<td>851207</td>
<td>18</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>40 1101</td>
<td>110101</td>
<td>DISC CONTROLLER</td>
<td>851207</td>
<td>18</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>40 1101</td>
<td>110101</td>
<td>DISC CONTROLLER</td>
<td>851207</td>
<td>18</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>40 1101</td>
<td>110101</td>
<td>DISC CONTROLLER</td>
<td>851207</td>
<td>18</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

Enter C to continue or X to exit: C

SCREEN 52

128
## DELIVERY ORDER DATE LEVEL REPORT

<table>
<thead>
<tr>
<th></th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EQUIPMENT with unit costs</td>
</tr>
<tr>
<td>2</td>
<td>EQUIPMENT without unit costs</td>
</tr>
<tr>
<td>3</td>
<td>SERIAL NUMBERS</td>
</tr>
<tr>
<td>4</td>
<td>RETURN to the Reports Level Menu</td>
</tr>
</tbody>
</table>

SCREEN 53

---

Enter site number for which the report is desired: 01

SCREEN 54
DELIVERY ORDER LEVEL REPORT
EFFECTIVE DATE: 851207

The following Delivery Order Effective Dates exist for Site 40:

- 851207
- 851020
- 841127
- 850404
- 851110

Enter C to continue or X to exit:

SCREEN 55

DELIVERY ORDER LEVEL REPORT
EFFECTIVE DATE: 851207

Do you want a printed report? (Yes or No): N

Enter C to continue or X to exit:

SCREEN 56
### DELIVERY ORDER LEVEL REPORT
**EFFECTIVE DATE: 851207**

<table>
<thead>
<tr>
<th>SITE</th>
<th>CLIN</th>
<th>FEATURE#</th>
<th>DESCRIPTION</th>
<th>QTY</th>
<th>UNIT PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0001</td>
<td>000101</td>
<td>SITE POWER PREPARATIONS</td>
<td>1</td>
<td>101886.00</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>13</td>
<td>86750.00</td>
</tr>
<tr>
<td>40</td>
<td>0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td>13</td>
<td>19800.00</td>
</tr>
<tr>
<td>40</td>
<td>0104</td>
<td>013001</td>
<td>OSP WITH 6530</td>
<td>1</td>
<td>13387.50</td>
</tr>
<tr>
<td>40</td>
<td>0105</td>
<td>013101</td>
<td>CENTRONIX PRINTER</td>
<td>3</td>
<td>1615.50</td>
</tr>
<tr>
<td>40</td>
<td>0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>25</td>
<td>2317.50</td>
</tr>
<tr>
<td>40</td>
<td>0107</td>
<td>013202</td>
<td>PRINTER INTERFACE</td>
<td>1</td>
<td>409.50</td>
</tr>
<tr>
<td>40</td>
<td>0108</td>
<td>015001</td>
<td>PATCH PANEL CABINET</td>
<td>3</td>
<td>2250.00</td>
</tr>
<tr>
<td>40</td>
<td>0109</td>
<td>015101</td>
<td>SYSTEMS CABINET</td>
<td>4</td>
<td>14220.00</td>
</tr>
<tr>
<td>40</td>
<td>0110</td>
<td>015201</td>
<td>I/O POWER MODULE</td>
<td>12</td>
<td>3150.00</td>
</tr>
<tr>
<td>40</td>
<td>0109</td>
<td>015301</td>
<td>EXPANSION CABINET</td>
<td>1</td>
<td>2250.00</td>
</tr>
<tr>
<td>40</td>
<td>0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td>5</td>
<td>697.50</td>
</tr>
<tr>
<td>40</td>
<td>0113</td>
<td>016101</td>
<td>THL PATCH PANEL</td>
<td>1</td>
<td>315.00</td>
</tr>
<tr>
<td>40</td>
<td>0114</td>
<td>016201</td>
<td>ASYNC PATCH PANEL</td>
<td>2</td>
<td>697.50</td>
</tr>
<tr>
<td>40</td>
<td>0115</td>
<td>016301</td>
<td>SYNC PATCH PANEL</td>
<td>2</td>
<td>697.50</td>
</tr>
</tbody>
</table>

Enter C to continue or X to exit: C
DELIVERY ORDER LEVEL REPORT
EFFECTIVE DATE: 851207

The following Delivery Order Effective Dates exist for Site 40:

<table>
<thead>
<tr>
<th>Date</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>851207</td>
<td>841127</td>
</tr>
<tr>
<td>851020</td>
<td>850404</td>
</tr>
<tr>
<td>851110</td>
<td></td>
</tr>
</tbody>
</table>

Enter C to continue or X to exit:

SCREEN 59

DELIVERY ORDER LEVEL REPORT
EFFECTIVE DATE: 851207

Do you want a printed report? (Yes or No): N

Enter C to continue or X to exit:

SCREEN 60

132
# APPENDIX A: USER’s MANUAL

## DELIVERY ORDER LEVEL REPORT

**EFFECTIVE DATE:** 851207

<table>
<thead>
<tr>
<th>SITE</th>
<th>CLIN</th>
<th>FEATURE#</th>
<th>DESCRIPTION</th>
<th>QTY</th>
<th>MODEL NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0001</td>
<td>000101</td>
<td>SITE POWER PREPARATIONS</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>13</td>
<td>1432</td>
</tr>
<tr>
<td>40</td>
<td>0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td>13</td>
<td>2432</td>
</tr>
<tr>
<td>40</td>
<td>0104</td>
<td>013001</td>
<td>OSP WITH 6530</td>
<td>1</td>
<td>3910</td>
</tr>
<tr>
<td>40</td>
<td>0105</td>
<td>013101</td>
<td>CENTRONIX PRINTER</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0106</td>
<td>013201</td>
<td>6530 CRT</td>
<td>25</td>
<td>6530</td>
</tr>
<tr>
<td>40</td>
<td>0107</td>
<td>013202</td>
<td>PRINTER INTERFACE</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0108</td>
<td>015001</td>
<td>PATCH PANEL CABINET</td>
<td>3</td>
<td>7105</td>
</tr>
<tr>
<td>40</td>
<td>0109</td>
<td>015101</td>
<td>SYSTEMS CABINET</td>
<td>4</td>
<td>7104</td>
</tr>
<tr>
<td>40</td>
<td>0110</td>
<td>015201</td>
<td>I/O POWER MODULE</td>
<td>12</td>
<td>7301</td>
</tr>
<tr>
<td>40</td>
<td>0109</td>
<td>015301</td>
<td>EXPANSION CABINET</td>
<td>1</td>
<td>7107</td>
</tr>
<tr>
<td>40</td>
<td>0112</td>
<td>016001</td>
<td>DISC PATCH PANEL</td>
<td>5</td>
<td>7504</td>
</tr>
<tr>
<td>40</td>
<td>0113</td>
<td>016101</td>
<td>THL PATCH PANEL</td>
<td>1</td>
<td>7506</td>
</tr>
<tr>
<td>40</td>
<td>0114</td>
<td>016201</td>
<td>ASYNC PATCH PANEL</td>
<td>2</td>
<td>7501</td>
</tr>
<tr>
<td>40</td>
<td>0115</td>
<td>016301</td>
<td>SYNC PATCH PANEL</td>
<td>2</td>
<td>7502</td>
</tr>
</tbody>
</table>

Enter C to continue or X to exit: C

## SCREEN 61

### SITE SERIAL NUMBER REPORT

Enter site number for which the report is desired: 01

Enter C to continue or X to exit: 

## SCREEN 62

133
### SITE SERIAL NUMBER REPORT
EFFECTIVE DATE: 851207

The following Delivery Order Effective Dates exist for Site 40:

<table>
<thead>
<tr>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>851207</td>
</tr>
<tr>
<td>851207</td>
</tr>
<tr>
<td>841127</td>
</tr>
<tr>
<td>850404</td>
</tr>
<tr>
<td>851020</td>
</tr>
<tr>
<td>851110</td>
</tr>
</tbody>
</table>

Enter C to continue or X to exit:

**SCREEN 63**

---

Do you want a printed report? (Yes or No): **N**

Enter C to continue or X to exit:

**SCREEN 64**
## SITE SERIAL NUMBER REPORT

**Effective Date:** 08/12/07

<table>
<thead>
<tr>
<th>SITE</th>
<th>CLIN</th>
<th>FEATURE#</th>
<th>DESCRIPTION</th>
<th>EFFECT DATE</th>
<th>QTY</th>
<th>QTY NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0001</td>
<td>000101</td>
<td>SITE POWER PREPARATIONS</td>
<td>08/12/07</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>12</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>11</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>9</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>8</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>7</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>6</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>5</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>4</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>3</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>2</td>
</tr>
<tr>
<td>40</td>
<td>0101</td>
<td>010201</td>
<td>NS-TXP, 2 MEG</td>
<td>08/12/07</td>
<td>13</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>0102</td>
<td>010301</td>
<td>2 MEG MEMORY</td>
<td>08/12/07</td>
<td>13</td>
<td>13</td>
</tr>
</tbody>
</table>

Enter C to continue or X to exit: C

### SCREEN 65

#### MAINTENANCE DELIVERY ORDER GENERATION PROGRAM

Generate maintenance Delivery Order for Site Number: 40

**DISCOUNT and ESCALATION RATES**

Values input are added to one (1) to generate the correct discount or escalation rates

- LCN Hardware Maintenance Escalation Rate: 0.100
- LCN Software Maintenance Escalation Rate: 0.100
- SPLICEnet Maintenance Discount Rate: 0.000
- Site Maintenance Escalation Rate: 0.822

File name to be imported into Lotus 1-2-3: NEWDD.PRN

### SCREEN 66
MAILING LABEL GENERATION PROGRAM

Number of copies for each label: 01

Site Number: 01

SPLICED SITE MAILING LABELS

SCREEN 67
Before using the integrated Configuration Management System, make a backup copy of all five disks. Work with the backup copy and store the original disks safely away. This will allow the initial files to be restored if files are erased, damaged or an accident occurs.

Before the integrated Configuration Management System may be used, it must be installed on a micro-computer. Installation is easily accomplished using a DOS command batch file supplied on the Initial Startup Disk. Programs were developed on an IBM-PC environment and tested on an IBM-PC AT. System performance on other than a 100% IBM compatible configuration is unknown and without guarantee.

The integrated Configuration Management System must be run on a hard disk system configuration. This is a limitation caused by the size and number of dBASE files which are part of the system.

The integrated Configuration Management System consists of five disks, one installation disk and four system disks. Each disk is labelled to reflect the portion of the system residing on each disk. The label identifies the directory where the system files must be loaded. Three directories will be created during the installation process if they do not already exist. These directory names may not be modified. System execution is dependent on files existing in predefined locations.

Ensure the system default drive is the hard disk where the integrated system is to be loaded (ex: C\ or D\, etc.). Start system installation by placing the disk labelled Initial Startup Disk in drive A. Type the command STARTUP and follow the instructions and messages displayed on the screen.
Record of Changes

Original 1 February 1986
APPENDIX B: MAINTENANCE MANUAL  Page 3

List of Effective Pages

Page 1 through 285  Original
Table of Contents

Record of Changes --------------------------------------------- 2
List of Effective Pages ------------------------------------------ 3
Table of Contents ----------------------------------------------- 4
Overall System Data Flow Diagram ---------------------------------- 7
Configurer Data Flow Diagram (Level 1) -------------------------- 8
File Description Example ----------------------------------------- 9
Data Flow Description Example ------------------------------------ 11
Process Description Example ------------------------------------- 12
Configurer Structure Chart (Overview - Level 0) ------------------ 14
Configurer Structure Chart (Level 1) ----------------------------- 15
Configurer Structure Chart (Level 2) ----------------------------- 16
Module Description Example -------------------------------------- 17
Pascal Configurer Record Descriptions ---------------------------- 20
Configuration Input Data File (CONFIG.SIT) ----------------------- 22
Input Data File (COSTS.IN) --------------------------------------- 24
Configuration Management System Bachmann Diagram --------------- 28
dBASE III Data Base Structure Descriptions ---------------------- 29
dBASE III Configuration Management System Indices Composition -------------- 32
dBASE III Configuration Management System Program Invocation Sequences ------------------------------- 33
Pascal SPLICE Configurer Program Source Code ------------------ 34
dBASE III Configuration Management System Program Module Source Code:

141
<table>
<thead>
<tr>
<th>Program Name</th>
<th>Page Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>NEWDOADD.PRG</td>
<td>221</td>
</tr>
<tr>
<td>NEWDCMD.PRG</td>
<td>224</td>
</tr>
<tr>
<td>NEWDOCVT.PRG</td>
<td>226</td>
</tr>
<tr>
<td>PROJRPTS.PRG</td>
<td>235</td>
</tr>
<tr>
<td>REPORTCMD.PRG</td>
<td>237</td>
</tr>
<tr>
<td>SELECTOR.PRG</td>
<td>239</td>
</tr>
<tr>
<td>SERNGBLD.PRG</td>
<td>242</td>
</tr>
<tr>
<td>SERNOCMD.PRG</td>
<td>244</td>
</tr>
<tr>
<td>SERNOREV.PRG</td>
<td>246</td>
</tr>
<tr>
<td>SERNOUPD.PRG</td>
<td>252</td>
</tr>
<tr>
<td>SITERPTS.PRG</td>
<td>263</td>
</tr>
<tr>
<td>SNODTRPT.PRG</td>
<td>265</td>
</tr>
<tr>
<td>SNOPJRT.PRG</td>
<td>275</td>
</tr>
<tr>
<td>SNOSTRPT.PRG</td>
<td>280</td>
</tr>
</tbody>
</table>
FILE DESCRIPTION

PROJECT: SPLICE CONFIGURER
DATE: 3 September 1985

FILE OR DATABASE NAME: CONFIG.SIT

ALIASES: None

COMPOSITION: The CONFIG.SIT file contains the site specific data associated with all the designated Stock Point Logistics Integrated Communications Environment (SPLICE) sites.

ORGANIZATION: Sequential. The structure of the file is as follows:

<table>
<thead>
<tr>
<th>DATA ELEMENT</th>
<th>TYPE VARIABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site Number</td>
<td>Integer</td>
</tr>
<tr>
<td>Site Name</td>
<td>String</td>
</tr>
<tr>
<td>Documentation Site Group</td>
<td>Integer</td>
</tr>
<tr>
<td>Training Site Group</td>
<td>Integer</td>
</tr>
<tr>
<td>Maintenance Option</td>
<td>String</td>
</tr>
<tr>
<td>Maintenance Responsibility</td>
<td>String</td>
</tr>
<tr>
<td>Site Type</td>
<td>String</td>
</tr>
<tr>
<td>Installation Cost</td>
<td>Real</td>
</tr>
</tbody>
</table>

* NOTES: These data elements are not currently designated for implementation, but are specified for use in later revisions.

1. Site Number range can be from one (1) to sixty-two (62). Current only fifty-six (56) sites are designated SPLICE sites and is the upper range limit.

2. Documentation Site Group is used to restrict the maximum number of documentation sets that each site is allowed to receive.

3. Training Site Group is used to restrict the maximum number of training courses that each site is allowed to receive.
4. Maintenance Option and Maintenance Responsibility are used together to establish the maintenance repair and response times desired by each site.

5. Site Type restricts various hardware options to certain designated sites. The value is either "S" or "M". "S" designates a site as a Stock Point which can receive all hardware/software options. "M" designates a site as a Multiple Activity Processing System (MAPS) site which is not permitted to receive Local Computer Network (e.g. HYPERchannel) components.

6. Installation Cost is a one time cost that is paid to the vendor for his initial site survey and installation preparations.
DATA FLOW DESCRIPTION

PROJECT: SPLICE CONFIGURER
DATE: 3 September 1985

DATAFLOW NAME: Config_Data

ALIASES: None

COMPOSITION: The data represented in this flow is the data coming from the input file "CONFIG.SIT". The site number selected for configuration is located within the CONFIG.SIT file and site unique information is extracted. This unique site configuration data is then used to create the site information record. This record is used to determine the maximum limits applicable to sites under configuration, as specified in the notes to the CONFIG.SIT file description. The site information record also is used to determine which repair and maintenance options are to be selected and serves to restrict certain types of options from being selected, depending upon the sites type designation. The Site Preparations Charge is taken from the CONFIG.SIT file and input as the first entry in the COSTTABLE array.

NOTES: The user was previously prompted for the number of the site to be configured.
PROCESS DESCRIPTION

PROJECT: SPLICE CONFIGURER
DATE: 3 September 1985

PROCESS NAME: Build Cost Array

PROCESS NUMBER: 1.2

PROCESS DESCRIPTION:

1. Take a feature number for each element resident in the input cost file and place it in a feature number field in the cost array.

2. Take a contract line item number for each element resident in the input cost file and place it in a contract line item number field in the cost array.

3. Take a nomenclature description for each element resident in the input cost file and place it in an item description field in the cost array.

4. Take the unit maintenance costs from the input cost file and place it in the fourth element of the cost array.

5. Take the unit purchase price from the input cost file and apply a discount rate specified by the user. Place the result in the fifth element of the cost array.

6. Take the unit installation cost from the input cost file and apply an escalation rate specified by the user. Place the result in the sixth element of the cost array.

7. Take the basic monthly maintenance rate from the input cost file and apply an escalation rate specified by the user. Place the result in the seventh element of the cost array.
NOTES: The cost array mentioned on the previous page is a two dimensional memory array. The array contains an entry for every line item identified on the Automated Data Processing Selection Office (ADPSO) SPLICE contract. The maximum number of entries expected is two hundred. This estimate is based upon the maximum number of possible line items that may exist for available selections. The site cost array structure is planned as follows:

<table>
<thead>
<tr>
<th>FEATURE NUMBER</th>
<th>VARIABLE TYPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contract Line Item Number (CLIN)</td>
<td>String</td>
</tr>
<tr>
<td>CLIN Description</td>
<td>String</td>
</tr>
<tr>
<td>Monthly Maintenance</td>
<td>Real</td>
</tr>
<tr>
<td>CLIN Unit Price</td>
<td>Real</td>
</tr>
<tr>
<td>Installation Cost</td>
<td>Real</td>
</tr>
<tr>
<td>Basic Monthly Maintenance Cost</td>
<td>Real</td>
</tr>
</tbody>
</table>
CONFIGURER STRUCTURE CHART

OVERVIEW - LEVEL 0

SUMMARY

CONFIGURE SITE

CONFIGURE COMPONENTS

INITIALIZE

APPENDIX B: MAINTENANCE MANUAL  Page 14
CONFIGURER STRUCTURE CHART - LEVEL 1

APPENDIX B: MAINTENANCE MANUAL

152
MODULE DESCRIPTION

SPLICE DESIGN
DATE: 3 September 1985

MODULE NAME: Print_Software

MODULE PURPOSE: This routine writes software related contract line item numbers (CLINs) to the output disk file. It invokes global procedure LINE_SETUP to generate the CLIN and accumulate section and appropriation totals. The global variable Quantity is used to compute the CLIN extended price and installation costs. The procedure is not used in maintenance computations.

INPUT: None.

OUTPUT: CLIN related data elements written to output disk file are:

- Line_Number String-7
- CostTable[I].featureno String-8
- CostTable[I].descript String-28
- Quantity Integer-3
- CostTable[I].purcprice Real-13, 2 decimals
- Extended_Price Real-12, 2 decimals
- CostTable[I].basemaint Real-9, 2 decimals
- Maint_Factor Real-8, 3 decimals
- Maint_Months Integer-5
- CostTable[I].basemaint * Maint_Factor * Maint_Months Real-9, 2 decimals
- CostTable[I].instcost Real-8, 2 decimals
- CostTable[I].instcost * Quantity Real-9, 2 decimals
- Downtime_Credit Real-9, 2 decimals
- CostTable[I].basemaint * Quantity * Maint_Factor Real-9, 2 decimals
PROCEDURAL DESCRIPTION:

Begin [Print Software]
CASE Type_Software of
  1: Begin { Per Processor Software }
      Maint_Factor = Momaint_Esc_Cost
      Extended_Price = Quantity * CostTable
      End
  2: Begin { Per Site Software }
      Maint_Factor = Momaint_Esc_Cost
      Extended_Price = CostTable[I].purchprice
      End
  3: Begin { NETEX Software }
      Maint_Factor = 1
      Extended_Price = CostTable[I].purchprice
      * Quantity
      End
END [End of CASE Statement]
CALL LINE_SETUP
  Compute_System_Downtime_Component * See Notes
  Compute_Downtime_Credit * See Notes
  Write_CLIN_Data.Elements_to_Output_Disk_File
End [Print_Software]
MODULE DESCRIPTION (Continued)

VARIABLES:

PROGRAM GLOBALS: See CONFIGURE_SITE module description

MODULE LOCALS: None.

PROCEDURE LOCALS:

1: Type_Software - Integer, parameter list variable, Range: 1-3, code controlling which values are assigned to the variables Maint_Factor and Extended_Price.

NOTES:

1. Computation for System_Downtime_Component:
   System_Downtime_Component = (Maint_Factor * Quantity * CostTable[I].basemaint) + (Maint_Factor * Quantity * CostTable[I].basemaint)

2. Computation for Downtime_Credit:
   (((CostTable[I].purchprice + CostTable[I].instcost) / 48) + (CostTable[I].basemaint * Maint_Factor)) * 0.005
APPENDIX B: MAINTENANCE MANUAL  Page 20

PASCAL CONFIGURER RECORD DESCRIPTIONS

1. COSTS.IN - file contains the individual contract line items which appear as line items on the generated delivery order.

<table>
<thead>
<tr>
<th>COLUMN POSITION</th>
<th>FIELD LENGTH</th>
<th>DATA ELEMENT DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>01-04</td>
<td>4</td>
<td>Contract Line Item Number (CLIN)</td>
</tr>
<tr>
<td>05</td>
<td>1</td>
<td>Blank (Filler)</td>
</tr>
<tr>
<td>06-11</td>
<td>6</td>
<td>Contract Feature Number</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>Blank (Filler)</td>
</tr>
<tr>
<td>13-39</td>
<td>27</td>
<td>Component Description</td>
</tr>
<tr>
<td>40-48</td>
<td>9</td>
<td>Basic Contract Maintenance Rate</td>
</tr>
<tr>
<td>49</td>
<td>1</td>
<td>Blank (Filler)</td>
</tr>
<tr>
<td>50-60</td>
<td>11</td>
<td>Basic Contract Purchase Price</td>
</tr>
<tr>
<td>61</td>
<td>1</td>
<td>Blank (Filler)</td>
</tr>
<tr>
<td>62-69</td>
<td>8</td>
<td>Basic Contract Installation Rate</td>
</tr>
<tr>
<td>70-80</td>
<td>11</td>
<td>Blank (Filler)</td>
</tr>
</tbody>
</table>

NOTE: All data elements are left justified. This file is read into a memory array (COSTTABLE). The data elements are modified by the discount and escalation rates entered by the user. The file is maintained in Contract Feature Number sequence, with two exceptions. T-Text and TRANSFER line items are not in Contract Feature Number sequence. Use extreme care when adding components and corresponding line items in the source code. Line items are identified in the source code by using comments. An example of a comment is

{ I=6 Serial Printers }
PASCAL CONFIGURER RECORD DESCRIPTIONS (Continued)

2. CONFIG.SIT - file contains site specific information used to determine several factors required in the configuration process.

<table>
<thead>
<tr>
<th>COLUMN POSITION</th>
<th>FIELD LENGTH</th>
<th>DATA ELEMENT DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>01-02</td>
<td>2</td>
<td>Site Number</td>
</tr>
<tr>
<td>03-30</td>
<td>28</td>
<td>Site Name</td>
</tr>
<tr>
<td>31</td>
<td>1</td>
<td>Documentation Site Group</td>
</tr>
<tr>
<td>32</td>
<td>1</td>
<td>Blank (Filler)</td>
</tr>
<tr>
<td>33</td>
<td>1</td>
<td>Training Site Group</td>
</tr>
<tr>
<td>34</td>
<td>1</td>
<td>Blank (Filler)</td>
</tr>
<tr>
<td>35-38</td>
<td>4</td>
<td>Maintenance Option</td>
</tr>
<tr>
<td>39</td>
<td>1</td>
<td>Blank (Filler)</td>
</tr>
<tr>
<td>40</td>
<td>1</td>
<td>Maintenance Responsibility</td>
</tr>
<tr>
<td>41</td>
<td>1</td>
<td>Blank (Filler)</td>
</tr>
<tr>
<td>42</td>
<td>1</td>
<td>Site Type (Stock Point or MAP Site)</td>
</tr>
<tr>
<td>43</td>
<td>1</td>
<td>Blank (Filler)</td>
</tr>
<tr>
<td>44-49</td>
<td>6</td>
<td>Installation Cost</td>
</tr>
<tr>
<td>50-80</td>
<td>31</td>
<td>Blank (Filler)</td>
</tr>
</tbody>
</table>

NOTE: All data elements are left justified. Site specific information is read into a memory array (SITEINFO). The file is maintained in site number sequence. Site installation costs were obtained from NAVSUP SPLICE personnel. Installation costs reflect costs originally specified in the SPLICE contract. If these costs are not correct or are revised, update the site preparation charges in CONFIG.SIT prior to running the configurer.
## APPENDIX B: MAINTENANCE MANUAL

### CONFIG.SIT Program Listing

| 1 | 01 | ASO PHILADELPHIA, PA | 2 | 2 | X | A | S | 81735.0 |
| 2 | 02 | FMSO MECHANICSBURG, PA | 1 | 2 | III | B | S | 93939.0 |
| 3 | 03 | FMSO MECHANICSBURG, PA | 4 | 4 | I | F | M | 56721.0 |
| 4 | 04 | MCAS CHERRY POINT, NC | 3 | 3 | VIII | C | M | 70860.0 |
| 5 | 05 | MCAS EL TORO, CA | 3 | 3 | II | D | M | 76473.0 |
| 6 | 06 | MCAF QUANTICO, VA | 4 | 4 | P | F | M | 59748.0 |
| 7 | 07 | MCAS YUMA, AZ | 4 | 4 | I | F | M | 59748.0 |
| 8 | 08 | NAC INDIANAPOLIS, IN | 4 | 4 | P | A | M | 59748.0 |
| 9 | 09 | NARDAC JACKSONVILLE, FL | 2 | 2 | VIII | A | S | 188471.0 |
| 10 | 10 | NARDAC NEW ORLEANS, LA | 2 | 2 | P | A | S | 73918.0 |
| 11 | 11 | NARDAC NORFOLK, VA | 2 | 2 | VIII | A | S | 74913.0 |
| 12 | 12 | NARDAC PENSACOLA, FL | 2 | 2 | VIII | A | S | 76523.0 |
| 13 | 13 | NARDAC SAN DIEGO, CA | 2 | 2 | VIII | A | S | 74829.0 |
| 14 | 14 | NARDAC SAN FRANCISCO, CA | 2 | 2 | VIII | A | S | 75967.0 |
| 15 | 15 | NARDAC WASHINGTON, DC | 4 | 4 | P | F | S | 59748.0 |
| 16 | 16 | NAS BARBERS POINT, HI | 4 | 4 | I | F | M | 59748.0 |
| 17 | 17 | NAS BRUNSWICK, ME | 4 | 4 | I | F | M | 59748.0 |
| 18 | 18 | NAS CECIL FIELD, FL | 4 | 4 | I | F | M | 59748.0 |
| 19 | 19 | NAS KEY WEST, FL | 4 | 4 | I | F | M | 59748.0 |
| 20 | 20 | NAEC LAKE HURST, NJ | 4 | 4 | I | F | M | 59748.0 |
| 21 | 21 | NAS MEMPHIS, TN | 4 | 4 | I | F | M | 59748.0 |
| 22 | 22 | NAS MIRAMAR, CA | 4 | 4 | I | F | M | 59748.0 |
| 23 | 23 | NAS OCEANA, VA | 4 | 4 | I | F | M | 59748.0 |
| 24 | 24 | NAS PENSACOLA, FL | 4 | 4 | I | F | M | 59748.0 |
| 25 | 25 | NAS WHIDBEY ISLAND, WA | 3 | 3 | P | D | M | 68448.0 |
| 26 | 26 | NATC PATUXENT RIVER, MD | 3 | 3 | II | D | M | 63841.0 |
| 27 | 27 | PMTC POINT MUGU, CA | 4 | 4 | I | F | M | 59748.0 |
| 28 | 28 | NAVDAF Corpus Christi, TX | 4 | 4 | I | F | M | 59748.0 |
| 29 | 29 | NAVDAF GREAT LAKES, IL | 4 | 4 | I | F | M | 59748.0 |
| 30 | 30 | NAVDAF LEMOORE, CA | 4 | 4 | I | F | M | 59748.0 |
| 31 | 31 | NAVDAF MOFFETT FIELD, CA | 4 | 4 | I | F | M | 59748.0 |
| 32 | 32 | NAVDAF ORLANDO, FL | 4 | 4 | I | F | M | 59748.0 |
| 33 | 33 | NRCC LONG BEACH, CA | 4 | 4 | I | F | S | 57816.0 |
| 34 | 34 | NRCC NEWPORT, RI | 4 | 4 | I | F | S | 57816.0 |
| 35 | 35 | NRCC PHILADELPHIA, PA | 4 | 4 | I | F | S | 57816.0 |
| 36 | 36 | NRCC WASHINGTON, D.C. | 4 | 4 | I | F | S | 57816.0 |
| 37 | 37 | NUWES KEYPORT, WA | 4 | 4 | I | F | M | 59748.0 |
| 38 | 38 | NAVSTA Mayport, FL | 4 | 4 | I | F | M | 59748.0 |
| 39 | 39 | NSC CHARLESTON, SC | 2 | 2 | VIII | A | S | 78279.0 |
| 40 | 40 | NSC NORFOLK, VA | 2 | 2 | X | A | S | 101886.0 |
| 41 | 41 | NSC OAKLAND, CA | 2 | 2 | V | A | S | 94646.0 |
| 42 | 42 | NSC PEARL HARBOR, HI | 2 | 2 | VIII | A | S | 89493.0 |
| 43 | 43 | NSC PUGET SOUND, WA | 2 | 2 | VIII | A | S | 89493.0 |
| 44 | 44 | NSC SAN DIEGO, CA | 2 | 2 | V | A | S | 95620.0 |
| 45 | 45 | NSD GUAM | 3 | 3 | VII | E | S | 101824.0 |
| 46 | 46 | NSD SUBIC BAY, PHI | 3 | 3 | P | E | S | 102835.0 |
| 47 | 47 | NSD YOKOSUKA, JAPAN | 3 | 3 | VII | E | S | 100835.0 |
| 48 | 48 | NSY PHILADELPHIA, PA | 4 | 4 | I | E | M | 59748.0 |
| 49 | 49 | NSY PORTSMOUTH, NH | 4 | 4 | I | E | M | 59748.0 |
| 50 | 50 | NSY SAN DIEGO, CA | 4 | 4 | I | E | M | 59748.0 |
### CONFIG.SIT Program Listing

<p>| | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>51</td>
<td>51 SPCC MECHANICSBURG, PA</td>
<td>2</td>
<td>2</td>
<td>X</td>
<td>A</td>
<td>S</td>
<td>95520.0</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>52 SUBASE KINGS BAY, GA</td>
<td>4</td>
<td>4</td>
<td>VIII</td>
<td>D</td>
<td>M</td>
<td>59748.0</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>53 SUBASE NEW LONDON, CN</td>
<td>4</td>
<td>4</td>
<td>I</td>
<td>F</td>
<td>M</td>
<td>59748.0</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>54 SUBASE PEARL HARBOR, HI</td>
<td>4</td>
<td>4</td>
<td>I</td>
<td>F</td>
<td>M</td>
<td>59748.0</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>55 SWFPAC BREMERTON, WA</td>
<td>3</td>
<td>3</td>
<td>VI</td>
<td>E</td>
<td>M</td>
<td>88507.0</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>56 TRF BANGOR, WA</td>
<td>4</td>
<td>4</td>
<td>P</td>
<td>E</td>
<td>S</td>
<td>88507.0</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>57 SWFPAC KINGS BAY, GA</td>
<td>3</td>
<td>3</td>
<td>VI</td>
<td>E</td>
<td>M</td>
<td>88507.0</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>58 TRF KINGS BAY, GA</td>
<td>4</td>
<td>4</td>
<td>P</td>
<td>E</td>
<td>S</td>
<td>88507.0</td>
<td></td>
</tr>
</tbody>
</table>
### COSTS.IN Program Listing

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Cost1</th>
<th>Cost2</th>
<th>Cost3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>000101 SITE POWER PREPARATIONS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>0101</td>
<td>010201 NS-TXP, 2 NEG</td>
<td>439.20</td>
<td>96400.00</td>
<td>700.00</td>
</tr>
<tr>
<td>0102</td>
<td>010301 2 MEG MEMORY</td>
<td>89.06</td>
<td>22000.00</td>
<td>113.00</td>
</tr>
<tr>
<td>0103</td>
<td>012401 FLTG PT ARITH</td>
<td>2000.00</td>
<td>700.00</td>
<td>100.00</td>
</tr>
<tr>
<td>0104</td>
<td>013001 OSP WITH 6530</td>
<td>198.86</td>
<td>14875.00</td>
<td>0.00</td>
</tr>
<tr>
<td>0105</td>
<td>013101 CENTRONIX PRINTER</td>
<td>30.00</td>
<td>1795.00</td>
<td>0.00</td>
</tr>
<tr>
<td>0106</td>
<td>013201 6530 CRT</td>
<td>35.38</td>
<td>2575.00</td>
<td>0.00</td>
</tr>
<tr>
<td>0107</td>
<td>013202 PRINTER INTERFACE</td>
<td>2.44</td>
<td>455.00</td>
<td>0.00</td>
</tr>
<tr>
<td>0108</td>
<td>015001 OSP WITH 6530</td>
<td>198.86</td>
<td>14875.00</td>
<td>0.00</td>
</tr>
<tr>
<td>0109</td>
<td>015101 SYSTEMS CABINET</td>
<td>202.52</td>
<td>15800.00</td>
<td>600.00</td>
</tr>
<tr>
<td>0110</td>
<td>015201 I/O POWER MODULE</td>
<td>48.80</td>
<td>3500.00</td>
<td>600.00</td>
</tr>
<tr>
<td>0112</td>
<td>016001 DISC PATCH PANEL</td>
<td>0.00</td>
<td>350.00</td>
<td>0.00</td>
</tr>
<tr>
<td>0114</td>
<td>016201 ASYNC PATCH PANEL</td>
<td>7.32</td>
<td>775.00</td>
<td>75.00</td>
</tr>
<tr>
<td>0115</td>
<td>016301 SYNC PATCH PANEL</td>
<td>7.32</td>
<td>775.00</td>
<td>75.00</td>
</tr>
<tr>
<td>1101</td>
<td>110101 DISC CONTROLLER</td>
<td>70.76</td>
<td>10500.00</td>
<td>200.00</td>
</tr>
<tr>
<td>1201</td>
<td>120201 DISC, WINCHESTER, 128MB</td>
<td>123.22</td>
<td>19500.00</td>
<td>325.00</td>
</tr>
<tr>
<td>1202</td>
<td>120301 DRAWER, WINCHESTER, 128MB</td>
<td>123.22</td>
<td>16500.00</td>
<td>325.00</td>
</tr>
<tr>
<td>1301</td>
<td>130201 DISC, MOVING HEAD, 240MB</td>
<td>253.76</td>
<td>26500.00</td>
<td>450.00</td>
</tr>
<tr>
<td>1401</td>
<td>140201 DISC, WINCHESTER, 540MB</td>
<td>395.28</td>
<td>39500.00</td>
<td>625.00</td>
</tr>
<tr>
<td>2101</td>
<td>210101 TAPE CONTROLLER</td>
<td>41.48</td>
<td>6100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>2102</td>
<td>210201 TAPE DRIVE/FORMATTER</td>
<td>469.70</td>
<td>47500.00</td>
<td>475.00</td>
</tr>
<tr>
<td>24</td>
<td>240101 CARD RDR/PNCH</td>
<td>191.00</td>
<td>20442.00</td>
<td>75.00</td>
</tr>
<tr>
<td>24</td>
<td>240201 CARD READER</td>
<td>56.12</td>
<td>5600.00</td>
<td>175.00</td>
</tr>
<tr>
<td>27</td>
<td>270101 LP/CR CONTROLLER</td>
<td>24.40</td>
<td>2800.00</td>
<td>188.00</td>
</tr>
<tr>
<td>27</td>
<td>270201 1000 LPM PRINTER</td>
<td>202.52</td>
<td>20000.00</td>
<td>100.00</td>
</tr>
<tr>
<td>28</td>
<td>270301 600 LPM PRINTER</td>
<td>202.52</td>
<td>14000.00</td>
<td>100.00</td>
</tr>
<tr>
<td>31</td>
<td>310101 INTRPROC BUS(INCL.W/010101)</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31</td>
<td>310201 FIBER OPTIC LINK CNTRL</td>
<td>610.00</td>
<td>35000.00</td>
<td>450.00</td>
</tr>
<tr>
<td>32</td>
<td>320101 320101 TANDEM/P-E HC ADAP</td>
<td>215.00</td>
<td>38940.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32</td>
<td>320201 320201 HC ADAPTER 2ND TRUNK</td>
<td>38.00</td>
<td>4705.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32</td>
<td>320301 320301 HC CABINET (3 ADAP)</td>
<td>14.00</td>
<td>3760.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32</td>
<td>320401 320301 TCL CONTROLLER</td>
<td>194.22</td>
<td>14000.00</td>
<td>300.00</td>
</tr>
<tr>
<td>32</td>
<td>320701 320400 HC TRUNK, 500 FT.</td>
<td>0.00</td>
<td>400.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32</td>
<td>320701 320400 HC TRUNK, 1000 FT.</td>
<td>0.00</td>
<td>800.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32</td>
<td>320701 320402 HC TRUNK, 1500 FT.</td>
<td>0.00</td>
<td>3075.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32</td>
<td>320701 320403 HC TRUNK, 2500 FT.</td>
<td>0.00</td>
<td>6250.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32</td>
<td>320701 320404 HC TRUNK, 4000 FT.</td>
<td>0.00</td>
<td>12600.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32</td>
<td>320701 320405 HC TRUNK, 5000 FT.</td>
<td>0.00</td>
<td>22750.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32</td>
<td>320701 320406 HC TRUNK, 7500 FT.</td>
<td>0.00</td>
<td>38400.00</td>
<td>0.00</td>
</tr>
<tr>
<td>33</td>
<td>330101 330101 Burroughs DCC HC</td>
<td>215.00</td>
<td>38410.00</td>
<td>0.00</td>
</tr>
<tr>
<td>33</td>
<td>330201 330201 Burroughs DLP HC</td>
<td>215.00</td>
<td>38410.00</td>
<td>0.00</td>
</tr>
<tr>
<td>33</td>
<td>330301 330301 ECBIC/ASCII RAM</td>
<td>16.00</td>
<td>3225.00</td>
<td>0.00</td>
</tr>
<tr>
<td>34</td>
<td>340300 340300 HC PROC 1/F(P.E./INTERDATA)</td>
<td>19.00</td>
<td>4060.00</td>
<td>0.00</td>
</tr>
<tr>
<td>36</td>
<td>360101 360101 HC ADAPTER (IBM 360/370)</td>
<td>215.00</td>
<td>39515.00</td>
<td>0.00</td>
</tr>
<tr>
<td>37</td>
<td>370101 370101 HC ADAPTER (UNIVAC 1100,1400)</td>
<td>215.00</td>
<td>38410.00</td>
<td>0.00</td>
</tr>
<tr>
<td>41</td>
<td>410101 410101 HC ADAPTER (IBM PS)</td>
<td>215.00</td>
<td>39515.00</td>
<td>0.00</td>
</tr>
<tr>
<td>42</td>
<td>420301 420301 HC PROC 1/F (MINI-COMPUTER)</td>
<td>19.00</td>
<td>4060.00</td>
<td>0.00</td>
</tr>
<tr>
<td>45</td>
<td>450101 450101 ASYNCH CNTR</td>
<td>21.96</td>
<td>3600.00</td>
<td>125.00</td>
</tr>
<tr>
<td>Page 2 COSTS.IN Program Listing</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>APPENDIX B: MAINTENANCE MANUAL</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Page 25</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>4502 450102 ASYNCH EXTENSION BOARD</td>
<td>26.84</td>
<td>4300.0</td>
<td>188.0</td>
</tr>
<tr>
<td>52</td>
<td>4503 450103 AUTOMATIC CALLING UNIT</td>
<td>8.19</td>
<td>1540.0</td>
<td>0.0</td>
</tr>
<tr>
<td>53</td>
<td>4504 450301 COMM SUBSYSTEM BASE</td>
<td>160.0</td>
<td>23673.0</td>
<td>630.0</td>
</tr>
<tr>
<td>54</td>
<td>4505 450302 BASE ADD-ON</td>
<td>124.0</td>
<td>19374.0</td>
<td>500.0</td>
</tr>
<tr>
<td>55</td>
<td>4506 450303 RS-232 LIU/CABLE</td>
<td>12.0</td>
<td>1869.0</td>
<td>157.0</td>
</tr>
<tr>
<td>56</td>
<td>4507 450304 6100 CABLE/30M</td>
<td>0.0</td>
<td>145.0</td>
<td>0.0</td>
</tr>
<tr>
<td>57</td>
<td>4508 450305 6100 CABLE/45M</td>
<td>0.0</td>
<td>160.0</td>
<td>0.0</td>
</tr>
<tr>
<td>58</td>
<td>4509 450306 6100 CABLE/60M</td>
<td>0.0</td>
<td>175.0</td>
<td>0.0</td>
</tr>
<tr>
<td>59</td>
<td>4601 460101 BIT SYNCH CNTR</td>
<td>50.02</td>
<td>6059.0</td>
<td>125.0</td>
</tr>
<tr>
<td>60</td>
<td>4602 460201 BYTE SYNCH CNTR</td>
<td>35.38</td>
<td>5800.0</td>
<td>100.0</td>
</tr>
<tr>
<td>61</td>
<td>4701 470101 COMM.PATCH PANEL/LINE MON</td>
<td>140.4</td>
<td>6653.0</td>
<td>100.0</td>
</tr>
<tr>
<td>62</td>
<td>4702 470201 ARCLI</td>
<td>7.02</td>
<td>5145.0</td>
<td>5.0</td>
</tr>
<tr>
<td>63</td>
<td>5101 510101 GUARDIAN OS</td>
<td>158.6</td>
<td>3500.0</td>
<td>125.0</td>
</tr>
<tr>
<td>64</td>
<td>5102 510201 BATCH SUBSYSTEM</td>
<td>61.0</td>
<td>4444.0</td>
<td>0.0</td>
</tr>
<tr>
<td>65</td>
<td>5103 510301 FDC SYS UTILITIES</td>
<td>125.0</td>
<td>1000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>66</td>
<td>5201 520101 ENCOMPASS</td>
<td>323.3</td>
<td>8000.0</td>
<td>150.0</td>
</tr>
<tr>
<td>67</td>
<td>5202 520102 ENABLE (P/O 520101)</td>
<td>67.1</td>
<td>1500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>68</td>
<td>5203 520103 ENFORM (P/O 520101)</td>
<td>85.4</td>
<td>2000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>69</td>
<td>5204 520104 PATHWAY (P/O 520101)</td>
<td>103.7</td>
<td>2500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>70</td>
<td>5205 520105 TMF (P/O 520101)</td>
<td>122.0</td>
<td>2500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>71</td>
<td>5206 520106 DDL (P/O 520101)</td>
<td>36.6</td>
<td>500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>72</td>
<td>5207 520107 FDC TPS SAS</td>
<td>240.0</td>
<td>3500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>73</td>
<td>5308 530102 ENSCRIBE (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>74</td>
<td>5309 530103 SORT/MERGE (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>75</td>
<td>5310 530104 FUP(FILE UTIL PRG,P/0510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>76</td>
<td>5311 530105 FUP(PERIP UTL PRG,P/0510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>77</td>
<td>5312 530106 BACKUP/RESTORE(P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>78</td>
<td>5313 530107 FILE SYSTEM SECURITY</td>
<td>600.0</td>
<td>5000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>79</td>
<td>5314 530108 SYSTEM CARD READER SUPPORT</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>80</td>
<td>5401 540201 SPOOLER (P/O 510101)</td>
<td>24.4</td>
<td>500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>81</td>
<td>5502 550102 ENVOY (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>82</td>
<td>5502 550103 CUP(COM UTL PRG,P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>83</td>
<td>5503 550201 EXPAND</td>
<td>122.0</td>
<td>2000.0</td>
<td>50.0</td>
</tr>
<tr>
<td>84</td>
<td>5504 550301 EXCHANGE RJE HASP</td>
<td>24.4</td>
<td>500.0</td>
<td>50.0</td>
</tr>
<tr>
<td>85</td>
<td>5505 550401 AM3270 ACCESS METHOD</td>
<td>24.4</td>
<td>500.0</td>
<td>50.0</td>
</tr>
<tr>
<td>86</td>
<td>5506 550501 X25 ACCESS METHOD</td>
<td>24.4</td>
<td>500.0</td>
<td>50.0</td>
</tr>
<tr>
<td>87</td>
<td>5507 550601 HYPER LINK ACC MD(P/O 550101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>88</td>
<td>5508 550602 LCN FUP SUPPORT</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>89</td>
<td>5509 550701 DELETED FDC CRT SUPPORT</td>
<td>350.0</td>
<td>13000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>90</td>
<td>5510 550702 6100 ATP</td>
<td>27.0</td>
<td>430.0</td>
<td>50.0</td>
</tr>
<tr>
<td>91</td>
<td>55 550703 6100 BSC</td>
<td>27.0</td>
<td>430.0</td>
<td>50.0</td>
</tr>
<tr>
<td>92</td>
<td>55 550704 6100 ADCCP</td>
<td>27.0</td>
<td>430.0</td>
<td>50.0</td>
</tr>
<tr>
<td>93</td>
<td>55 550705 6100 TINET</td>
<td>27.0</td>
<td>430.0</td>
<td>50.0</td>
</tr>
<tr>
<td>94</td>
<td>5528 550706 BURR POLL/SELECT</td>
<td>27.0</td>
<td>430.0</td>
<td>50.0</td>
</tr>
<tr>
<td>95</td>
<td>5530 550707 SNAX AND SNAX/IILS</td>
<td>27.0</td>
<td>430.0</td>
<td>50.0</td>
</tr>
<tr>
<td>96</td>
<td>5528 550708 TR 3271</td>
<td>54.0</td>
<td>860.0</td>
<td>78.0</td>
</tr>
<tr>
<td>97</td>
<td>5530 550709 AM 6520</td>
<td>27.0</td>
<td>430.0</td>
<td>50.0</td>
</tr>
<tr>
<td>98</td>
<td>55 550710 FDC SNA INTERFACE PACKAGE</td>
<td>350.0</td>
<td>84000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>99</td>
<td>55 550711 FDC DLANET INTERFACE PRG</td>
<td>400.0</td>
<td>25000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>100</td>
<td>5511 550801 BURROUGHS HTC NETEX</td>
<td>156.4</td>
<td>391.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
### APPENDIX B: MAINTENANCE MANUAL

#### Page 3

**COSTS.IN Program Listing**

<table>
<thead>
<tr>
<th>Page</th>
<th>55</th>
<th>550802</th>
<th>DELETED HTC PRESENTATION</th>
<th>450.0</th>
<th>0.0</th>
<th>0.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>55</td>
<td>550803</td>
<td>CIP, BURROUGHS HTC</td>
<td>450.0</td>
<td>22500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>102</td>
<td>55</td>
<td>550901</td>
<td>BURROUGHS DLP NETEX</td>
<td>680.0</td>
<td>720.0</td>
<td>0.0</td>
</tr>
<tr>
<td>103</td>
<td>55</td>
<td>550902</td>
<td>DELETED DLP PRESENTATION</td>
<td>450.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>104</td>
<td>55</td>
<td>550903</td>
<td>CIP, BURROUGHS DLP</td>
<td>450.0</td>
<td>22500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>105</td>
<td>55</td>
<td>551001</td>
<td>PE 3200 NETEX</td>
<td>680.0</td>
<td>725.0</td>
<td>0.0</td>
</tr>
<tr>
<td>106</td>
<td>55</td>
<td>551002</td>
<td>DELETED PE3200 PRESENTATION</td>
<td>450.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>107</td>
<td>55</td>
<td>551003</td>
<td>CIP, PERKIN-ELMER</td>
<td>450.0</td>
<td>22500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>108</td>
<td>55</td>
<td>551101</td>
<td>IBM NETEX</td>
<td>800.0</td>
<td>850.0</td>
<td>0.0</td>
</tr>
<tr>
<td>109</td>
<td>55</td>
<td>551102</td>
<td>DELETED IBM PRESENTATION</td>
<td>450.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>110</td>
<td>55</td>
<td>551103</td>
<td>CIP, IBM MVS</td>
<td>450.0</td>
<td>22500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>111</td>
<td>55</td>
<td>551104</td>
<td>UNIVAC 1100 NETEX</td>
<td>800.0</td>
<td>850.0</td>
<td>0.0</td>
</tr>
<tr>
<td>112</td>
<td>55</td>
<td>551105</td>
<td>DELETED IBM PRESENTATION</td>
<td>450.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>113</td>
<td>55</td>
<td>551106</td>
<td>CIP, IBM MVS</td>
<td>450.0</td>
<td>22500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>114</td>
<td>55</td>
<td>551201</td>
<td>TANDEM NETEX</td>
<td>326.4</td>
<td>816.0</td>
<td>0.0</td>
</tr>
<tr>
<td>115</td>
<td>55</td>
<td>551202</td>
<td>DELETED TANDEM PRESENTATION</td>
<td>450.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>116</td>
<td>55</td>
<td>551203</td>
<td>CIP, TANDEM</td>
<td>550.0</td>
<td>22500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>117</td>
<td>55</td>
<td>551301</td>
<td>DELETED DDN INTERFACE</td>
<td>24.4</td>
<td>500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>118</td>
<td>55</td>
<td>551302</td>
<td>DELETED DDN SVC INTERFACE</td>
<td>350.0</td>
<td>13000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>119</td>
<td>55</td>
<td>551303</td>
<td>DDN INTERFACE SUBSYSTEM</td>
<td>750.0</td>
<td>32000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>120</td>
<td>55</td>
<td>551500</td>
<td>NETWORK MGMT FACILITY GRP</td>
<td>324.0</td>
<td>13200.0</td>
<td>0.0</td>
</tr>
<tr>
<td>121</td>
<td>55</td>
<td>551501</td>
<td>NMF BASE FACILITY</td>
<td>180.0</td>
<td>6000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>122</td>
<td>55</td>
<td>551502</td>
<td>NMF PERFORMANCE MONITORING</td>
<td>75.0</td>
<td>3500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>123</td>
<td>55</td>
<td>551503</td>
<td>NMF DIAGNOSTIC MONITORING</td>
<td>75.0</td>
<td>3500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>124</td>
<td>55</td>
<td>551504</td>
<td>NMF ACCOUNTING APPLICATION</td>
<td>75.0</td>
<td>3500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>125</td>
<td>61</td>
<td>610101</td>
<td>EDIT (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>126</td>
<td>61</td>
<td>610102</td>
<td>TGA (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>127</td>
<td>61</td>
<td>610103</td>
<td>FILE COMPARISON UTILITY</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>128</td>
<td>61</td>
<td>610201</td>
<td>BLOCK STRUCTURED LANGUAGE</td>
<td>85.4</td>
<td>500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>129</td>
<td>61</td>
<td>610202</td>
<td>BINDER (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>130</td>
<td>61</td>
<td>610301</td>
<td>ENFORP (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>131</td>
<td>61</td>
<td>610401</td>
<td>TANDER (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>132</td>
<td>61</td>
<td>610501</td>
<td>BLOCK STRUCTURED LANGUAGE</td>
<td>61.0</td>
<td>170.0</td>
<td>0.0</td>
</tr>
<tr>
<td>133</td>
<td>61</td>
<td>610502</td>
<td>FORTRAN-ANSI 78</td>
<td>73.2</td>
<td>500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>134</td>
<td>61</td>
<td>610601</td>
<td>BINDER (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>135</td>
<td>61</td>
<td>610701</td>
<td>ENFORP (P/O 510101)</td>
<td>85.4</td>
<td>2000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>136</td>
<td>61</td>
<td>610801</td>
<td>DDL (P/O 510101)</td>
<td>36.6</td>
<td>500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>137</td>
<td>61</td>
<td>610901</td>
<td>BINDER (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>138</td>
<td>61</td>
<td>611001</td>
<td>FUP (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>139</td>
<td>61</td>
<td>611101</td>
<td>TANDER (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>140</td>
<td>61</td>
<td>611201</td>
<td>BLOCK STRUCTURED LANGUAGE</td>
<td>85.4</td>
<td>500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>141</td>
<td>61</td>
<td>611301</td>
<td>BINDER (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>142</td>
<td>61</td>
<td>611401</td>
<td>OSP (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>143</td>
<td>61</td>
<td>611501</td>
<td>ENFORP (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>144</td>
<td>61</td>
<td>611601</td>
<td>LOADFILE (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>145</td>
<td>61</td>
<td>611701</td>
<td>XRAY (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>146</td>
<td>61</td>
<td>611801</td>
<td>ENABLE (P/O 510101)</td>
<td>67.1</td>
<td>1500.0</td>
<td>0.0</td>
</tr>
<tr>
<td>147</td>
<td>65</td>
<td>650001</td>
<td>RUNTIME MSG SYS (P/O 510101)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>148</td>
<td>66</td>
<td>660101</td>
<td>TANDER DIAG SYS/STD/05101010</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>149</td>
<td>66</td>
<td>660201</td>
<td>TRANSFER</td>
<td>1.3</td>
<td>2000.0</td>
<td>0.0</td>
</tr>
<tr>
<td>150</td>
<td>66</td>
<td>660301</td>
<td>T-TEXT</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Line</td>
<td>Cost Center</td>
<td>Description</td>
<td>Cost</td>
<td>Hours</td>
<td>Notes</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
<td>-------------</td>
<td>------</td>
<td>-------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>151</td>
<td>67</td>
<td>CONFIG MGT QRY &amp; RPT 1 T/CH</td>
<td>95000.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>152</td>
<td>68</td>
<td>SFTWE CTL QRY &amp; RPT 1 T/CH</td>
<td>9000.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>153</td>
<td>7101</td>
<td>COMPUTER OPERATIONS MAN SET</td>
<td>427.00</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>154</td>
<td>7201</td>
<td>SYSTEMS PROGRAMMER MAN SET</td>
<td>607.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>155</td>
<td>7301</td>
<td>HARDWARE MANUAL SET</td>
<td>375.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>156</td>
<td>7401</td>
<td>PROGRAMMERS REF MAN SET</td>
<td>437.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>157</td>
<td>XXXX</td>
<td>TRAINING GROUP I</td>
<td>26837.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>158</td>
<td>XXXX</td>
<td>TRAINING GROUP II</td>
<td>164271.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>159</td>
<td>XXXX</td>
<td>TRAINING GROUP III</td>
<td>89655.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>XXXX</td>
<td>TRAINING GROUP IV</td>
<td>21909.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>161</td>
<td>XXXX</td>
<td>OPERATOR TRAINING</td>
<td>14109.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>162</td>
<td>XXXX</td>
<td>HARDWARE OVERVIEW</td>
<td>7000.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>163</td>
<td>XXXX</td>
<td>SYSTEMS RESOURCE MGT</td>
<td>20000.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>164</td>
<td>XXXX</td>
<td>SYSTEMS TUNING AND XRAY</td>
<td>15000.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>165</td>
<td>XXXX</td>
<td>DATA COMMUNICATIONS</td>
<td>10000.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>166</td>
<td>XXXX</td>
<td>TAL</td>
<td>15000.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>167</td>
<td>XXXX</td>
<td>SPLICENET MIGRATION WORKSHOP</td>
<td>8000.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>168</td>
<td>81</td>
<td>PM ON-CALL</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>169</td>
<td>81</td>
<td>PRVT MAINT FOR PER/CALL SIT</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>170</td>
<td>82</td>
<td>ON-CALL MAINTENANCE</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>171</td>
<td>83</td>
<td>PER-CALL MAINTENANCE</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>172</td>
<td>84</td>
<td>EMERGENCY PER-CALL MAINT</td>
<td>160.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>173</td>
<td>85</td>
<td>NETWORK ADMN COMP (P/05502010)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>174</td>
<td>89</td>
<td>TPS SIMULATION (P/O 520101)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>175</td>
<td>90</td>
<td>TPS APPL. INT (P/O 520101)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>176</td>
<td>91</td>
<td>TPS NTWK INTFCOMP (P/05201010)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>177</td>
<td>92</td>
<td>DSTB TPS PROC CMP (P/05201010)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>178</td>
<td>93</td>
<td>INTGRTED DDL CMP (P/O 5201010)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>179</td>
<td>94</td>
<td>TPS RECOVERY CMP (P/O 5201010)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>180</td>
<td>95</td>
<td>ENVISION (P/O 519101)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>181</td>
<td>96</td>
<td>CONTRACTOR PERS SUP (P/PERS6533)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>182</td>
<td>96</td>
<td>SPLICENET MIGRATION SUPPORT</td>
<td>800.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>183</td>
<td>97</td>
<td>CNFG MGT DATA &amp; RP(MTH COST</td>
<td>4500.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>184</td>
<td>98</td>
<td>CONTRACTOR TRAVEL COSTS</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>185</td>
<td>99</td>
<td>PRE-INST TEST FAC. (SEE NOTE)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>186</td>
<td>99</td>
<td>REMOTE BATCH TPLM. (SEE NOTE)</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>187</td>
<td>99</td>
<td>INTERACTIV TERM ACC SEE. NOTE</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>188</td>
<td>99</td>
<td>HAND ON TEST FAC SEE NOTE</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CONFIGURATION MANAGEMENT SYSTEM

BACHMAN DIAGRAM

CONFIGURATION DATA BASE
CONFIG.DBF

EQUIPMENT DATA BASE
EQUIP.DBF

DESCRIPTION DATA BASE
DESCRIP.DBF

SERIAL NUMBER DATA BASE
SERIALNO.DBF

MANUAL DATA BASE
MANUAL.DBF
APPENDIX B: MAINTENANCE MANUAL

# dBASE III DATA BASE STRUCTURE DESCRIPTIONS

**Structure for database: CONFIG.DBF**

**Date of last update: 12/21/85**

<table>
<thead>
<tr>
<th>Field</th>
<th>Field Name</th>
<th>Type</th>
<th>Width</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SITENO</td>
<td>Character</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>SITENAME</td>
<td>Character</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>SITECO</td>
<td>Character</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>SITENAMEFL</td>
<td>Character</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>SITEADD1</td>
<td>Character</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>SITEADD2</td>
<td>Character</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>SITECITY</td>
<td>Character</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>SITESTATE</td>
<td>Character</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>SITEZIP</td>
<td>Character</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>SITETYPE</td>
<td>Character</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>MAINTOPT</td>
<td>Character</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>MAINTRESP</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

**Total Record Width in Characters** 253

**Structure for database: DESCRIP.DBF**

**Date of last update: 12/08/85**

<table>
<thead>
<tr>
<th>Field</th>
<th>Field Name</th>
<th>Type</th>
<th>Width</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FEATURENO</td>
<td>Character</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>CLIN</td>
<td>Character</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>DESCRIPT</td>
<td>Character</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>MODELNO</td>
<td>Character</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>FDCMODEL</td>
<td>Character</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>TYPECOMPON</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>BASEMAINT</td>
<td>Numeric</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>NOTES</td>
<td>Memo</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

**Total Record Width in Characters** 83
### dBASE III DATA BASE STRUCTURE DESCRIPTIONS (Continued)

**Structure for database:** EQUIP.DBF  
**Date of last update:** 01/08/86

<table>
<thead>
<tr>
<th>Field</th>
<th>Field Name</th>
<th>Type</th>
<th>Width</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EFFDATE</td>
<td>Character</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>SITENO</td>
<td>Character</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>FEATURENO</td>
<td>Character</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>UNIT_PRICE</td>
<td>Numeric</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>MO_MAINT</td>
<td>Numeric</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>UNIT_INSTA</td>
<td>Numeric</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>QTY</td>
<td>Numeric</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

**Total Record Width in Characters:** 47

**Structure for database:** MANUAL.DBF  
**Date of last update:** 01/11/86

<table>
<thead>
<tr>
<th>Field</th>
<th>Field Name</th>
<th>Type</th>
<th>Width</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SITENO</td>
<td>Character</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>FEATURENO</td>
<td>Character</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>MANLDESC</td>
<td>Character</td>
<td>24</td>
<td></td>
</tr>
</tbody>
</table>

**Total Record Width in Characters:** 32

**Structure for database:** SERIALNO.DBF  
**Date of last update:** 01/08/86

<table>
<thead>
<tr>
<th>Field</th>
<th>Field Name</th>
<th>Type</th>
<th>Width</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EFFDATE</td>
<td>Character</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>SITENO</td>
<td>Character</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>FEATURENO</td>
<td>Character</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>QTY</td>
<td>Numeric</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>TOTQTY</td>
<td>Numeric</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>SERIALNO</td>
<td>Character</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

**Total Record Width in Characters:** 28
dBASE III DATA BASE STRUCTURE DESCRIPTIONS (Continued)

Structure for database: TED.DBF
Date of last update: 07/18/85

<table>
<thead>
<tr>
<th>Field</th>
<th>Field Name</th>
<th>Type</th>
<th>Width</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FILLER1</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>SITENO</td>
<td>Character</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>CLIN</td>
<td>Character</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>FILLER2</td>
<td>Character</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>FEATURENO</td>
<td>Character</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>FILLER3</td>
<td>Character</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>DESCRIPT</td>
<td>Character</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>FILLER4</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>QTY</td>
<td>Numeric</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>FILLER5</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>UNIT_PRICE</td>
<td>Numeric</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>FILLER6</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>TOT_PRICE</td>
<td>Numeric</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>14</td>
<td>FILLER7</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>MO_MAINT</td>
<td>Numeric</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>FILLER8</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>MAINT_FAC</td>
<td>Numeric</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>18</td>
<td>FILLER9</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>MAINT_MOS</td>
<td>Numeric</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>FILLER10</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>TOT_MAINT</td>
<td>Numeric</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>22</td>
<td>FILLER11</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>UNIT_INSTA</td>
<td>Numeric</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>24</td>
<td>FILLER12</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>TOT_INSTAL</td>
<td>Numeric</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>26</td>
<td>FILLER13</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>COMP_DT_CR</td>
<td>Numeric</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>28</td>
<td>FILLER14</td>
<td>Character</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>SYS_DT_CR</td>
<td>Numeric</td>
<td>11</td>
<td>2</td>
</tr>
</tbody>
</table>

** Total Record Width in Characters ** 156
## dBASE III Configuration Management System

### INDICES COMPOSITION

<table>
<thead>
<tr>
<th>DATA BASE FILE NAME</th>
<th>INDEX NAME</th>
<th>INDEX KEY COMPOSITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONFIG.DBF</td>
<td>CONFIG.NDX</td>
<td>SITENO</td>
</tr>
<tr>
<td>DESCRIPT.DBF</td>
<td>DESCRIPT.NDX</td>
<td>FEATURENO</td>
</tr>
<tr>
<td>EQUIP.DBF</td>
<td>EQUIPSIT.NDX</td>
<td>SITENO</td>
</tr>
<tr>
<td></td>
<td>EFEAT.NDX</td>
<td>FEATURENO</td>
</tr>
<tr>
<td></td>
<td>EQUIPSD.NDX</td>
<td>SITENO + EFFDATE</td>
</tr>
<tr>
<td></td>
<td>EQUIPDAT.NDX</td>
<td>SITENO + FEATURENO</td>
</tr>
<tr>
<td></td>
<td>EQUIPPRJ.NDX</td>
<td>EFFDATE + SITENO + FEATURENO</td>
</tr>
<tr>
<td>MANUAL.DBF</td>
<td>MANULSIT.NDX</td>
<td>SITENO + FEATURENO</td>
</tr>
<tr>
<td>SERIALNO.DBF</td>
<td>SERNOSIT.NDX</td>
<td>SITENO</td>
</tr>
<tr>
<td></td>
<td>SERNODAT.NDX</td>
<td>SITENO + EFFDATE</td>
</tr>
<tr>
<td></td>
<td>SERNOFEA.NDX</td>
<td>SITENO + FEATURENO</td>
</tr>
<tr>
<td></td>
<td>SERNOPRJ.NDX</td>
<td>EFFDATE + SITENO + FEATURENO</td>
</tr>
</tbody>
</table>
### dBASE III Configuration Management System

#### PROGRAM INVOCATION SEQUENCES

<table>
<thead>
<tr>
<th>DATA LOAD</th>
<th>EQUIPMENT FILE MAINTENANCE</th>
<th>DESCRIPTION FILE MAINTENANCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECTOR.PRG</td>
<td>SELECTOR.PRG</td>
<td>SELECTOR.PRG</td>
</tr>
<tr>
<td>MAINMENU.PRG</td>
<td>MAINMENU.PRG</td>
<td>MAINMENU.PRG</td>
</tr>
<tr>
<td>NEWDOCMD.PRG</td>
<td>EQUIPCMD.PRG</td>
<td>DESPMOD.PRG</td>
</tr>
<tr>
<td>NEWDOCVT.PRG</td>
<td>EQUIPUPD.PRG</td>
<td>DESPPUPD.PRG</td>
</tr>
<tr>
<td>NEWDOADD.PRG</td>
<td>EQUIPREV.PRG</td>
<td>DESPPREV.PRG</td>
</tr>
<tr>
<td>SERNOBLD.PRG</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CONFIGURATION FILE MAINTENANCE</th>
<th>MANUAL FILE MAINTENANCE</th>
<th>SERIAL NUMBER FILE MAINTENANCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECTOR.PRG</td>
<td>SELECTOR.PRG</td>
<td>SELECTOR.PRG</td>
</tr>
<tr>
<td>MAINMENU.PRG</td>
<td>MAINMENU.PRG</td>
<td>MAINMENU.PRG</td>
</tr>
<tr>
<td>CONFMOD.PRG</td>
<td>MANULCMD.PRG</td>
<td>SERNOCMD.PRG</td>
</tr>
<tr>
<td>CONFUPD.PRG</td>
<td>MANULADD.PRG</td>
<td>SERNOUPD.PRG</td>
</tr>
<tr>
<td>CONFREV.PRG</td>
<td>MANULUPD.PRG</td>
<td>SERNOREV.PRG</td>
</tr>
<tr>
<td></td>
<td>MANULDEL.PRG</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MANULREV.PRG</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PROJECT LEVEL REPORTS</th>
<th>SITE LEVEL REPORTS</th>
<th>EFFECTIVE DATE LEVEL REPORTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECTOR.PRG</td>
<td>SELECTOR.PRG</td>
<td>SELECTOR.PRG</td>
</tr>
<tr>
<td>MAINMENU.PRG</td>
<td>MAINMENU.PRG</td>
<td>MAINMENU.PRG</td>
</tr>
<tr>
<td>REPORCMD.PRG</td>
<td>REPORCMD.PRG</td>
<td>REPORCMD.PRG</td>
</tr>
<tr>
<td>PROJRPTS.PRG</td>
<td>SITERPTS.PRG</td>
<td>DATERPTS.PRG</td>
</tr>
<tr>
<td>EQPPJRPT.PRG</td>
<td>EQPSTRPT.PRG</td>
<td>EQPDTPRC.PRG</td>
</tr>
<tr>
<td>SNOPJRPT.PRG</td>
<td>MNLSTRPT.PRG</td>
<td>EQPDTNPC.PRG</td>
</tr>
<tr>
<td></td>
<td>SNOSTRPT.PRG</td>
<td>SNODTRPT.PRG</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MAINTENANCE DELIVERY ORDER</th>
<th>LABEL GENERATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECTOR.PRG</td>
<td>SELECTOR.PRG</td>
</tr>
<tr>
<td>MAINMENU.PRG</td>
<td>MAINMENU.PRG</td>
</tr>
<tr>
<td>MAINTDO.PRG</td>
<td>MKLABELS.PRG</td>
</tr>
</tbody>
</table>
Program SPLICE_CONFIGURE (Textin, Input, Output);

Title : SPLICE Configurer

Authors : LCDR Robert L. Beard, III, SC, USN
          LCDR Winston H. Buckley, SC, USN
          LCDR Edward J. Case, SC, USN

Purpose : To be used by Naval Supply Systems Command, SUP 0473, Point Logistic Integrated Communications Environment (SPLICE) sites. In later versions additions will be made to assist in preparing augmentations to existing sites, as well as prepare annual renewal delivery orders for existing sites.

Developed: 04 October 1985
Updated: 07 December 1985

General Comments: This program is being designed as an "expert" system. It will use a series of "rules of thumb" to develop and maintain SPLICE configurations at 62 sites throughout the world. The SPLICE configurations developed to date have been done by hand and have required extensive "hand message" by technical, financial, and contractor personnel to ensure their accuracy. This has proven to be both costly in terms of dollars and manpower. By prompting the user for key information, this "expert system" will develop technically accurate configurations, cost them out, and prepare the final delivery orders.

The following constants, type and variable declarations are used by the Software Bottling Company of New York screen generation program "SCREEN SCULPTOR".

Type

SIR2 = STRING[2]; SIR80 = STRING[80]; SIR79 = STRING[79];
ress = (staySS, prevSS, exitSS, nextSS);

Const 'C')Copyright 1984, The Software Bottling Company of New York';
'DO NOT REMOVE The Above Copyright Notice
This Program may not be used without the above Copyright Notice'

Const

User, Up Arrow Key, Left Arrow Key, Page Up Key 

ศาสตร์='A'; MSS='M'; ISS='K'; XSS='I';
Blank, Down Arrow Key, Right Arrow Key, Page Down Key 

BlankSS=' '; MSS='P'; MSS='M'; XSS='I';
APPENDIX B: MAINTENANCE MANUAL

Page 2

SPLICE.PAS Program Listing

51   | Function keys F1-F10 |
52   | f1SS='1'; f2SS='2'; f3SS='3'; f4SS='4'; f5SS='5';
53   | f6SS='6'; f7SS='7'; f8SS='8'; f9SS='9'; f10SS='0';
54   | retSS : STR2='"';
55
56
57 Var
58   | answerSS : String [1];
59   | rangeSS : STR80;
60   | DeepOnSS, last_fieldSS, retrieveSS : BOOLEAN;
61   | actionSS, last_field_actionSS : rez88;
62   | hiSS, loSS : REAL;
63   | vtypeSS, screenSS, screen_fieldSS, varSS : INTEGER;
64
65 |
66   | The following constants, type and variable declarations are used by the
67   | SPLICE configuror.
68
69 Type
70   | Op_Mode = (Hard, Soft, Document, Train, Maint, Other);
71
72   | Title = String [19];
73   | Names = Array [1..12] of String [9];
74   | CostType = Record
75       | featureno : String [6];
76       | clin : String [6];
77       | descrip : String [27];
78       | maint : Real;
79       | purchase : Real;
80       | instcost : Real;
81       | basement : Real;
82       | End;  |
83
84 SiteType = Record
85       | siteno : Integer;
86       | sitename : String [27];
87       | documentation : Integer;
88       | training : Integer;
89       | maint_options : String [4];
90       | maint_response : String [11];
91       | site_type : String [1];
92       | site_inst_cost : Real;
93       | End;  |
94
95 Cost
96   | File1 = 'Costs.DAT';
97   | File2 = 'Config.DAT';
98   | File3 = 'Spire.DAT';
Month_Name : Names = ('January', 'February', 'March', 'April',
'May', 'June', 'July', 'August',
'September', 'October', 'November', 'December');

Var

Month : Op_Mode; (Subscript for Totals)
SiteInfo : SiteType; (Record containing site specific info)
Subtotals : Array [0..5] of Array [1..2] of Real;
(Totals for each section)
Totals : Array [0..5] of Array [1..2] of Real;
(OPN & ONN Totals for each section)
CostTable : Array [1..200] of CostType;
(CardRdr, LRU, Processors, THYPERchannels : integer;
Maint_Months, NETIX_Months, CDN_SW_Months : integer;
M140, M150, A22, A400, A510, AXXX, I, Quantity : integer;
System_Downtime_Component, Downtime_Credit, Maint_Factor : Real;
Emerg_Maint_Rate, Extended_Price, Maint_Esc_Rate : Real;
Stock_Point : Char; (Variables for character responses)
Screenfile : File; (File of Screen Images)
Site_Preps : String [1]; (Yes or No user response variable)
Day : String [2]; (Effective Day of Delivery Order)
Year : String [4]; (Effective Year of Delivery Order)
Line_Number : String [6]; (Contract Line Item Number)
Month : String [9]; (Effective Month of Delivery Order)
PRN_File_Name : String [12]; (Output LOTUS .PRN file)
Diskfile : Text; (Output Delivery Order File)

{$V-,C-,R-} (Pascal Directives used by SCREEN SCULPTOR, See Compiler Manual)
{$I SPLICE.PAS Include Procedures In This File by SCREEN SCULPTOR.)
(C) COPYRIGHT, THE SOFTWARE HINLING COMPANY OF NEW YORK, 1984, 1985
** Turbo Pascal Version, Trade Mark Of Borland International)

TYPE

REPACKSS = record

AX, BX, CX, DX, BP, SI, DI, DS, ES, FL: word: INIT13H;
end;

VAR

REPACKSS : REPACKSS;

TYPE

video_pointerSS = array[1..3840] of CHAR;

VAR

video_pointerSS ;

videoSS, volfSS, conSS: byte;

videoSS: INTEGER;
PROCEDURE BEEP(BeepOn: BOOLEAN);
BEGIN
  if BeepOn then write(chr(7));
END;

PROCEDURE COLOR(foregr, backgr: BYTE);
VAR
  TextColor: procedure;
BEGIN
  if backgr > 7 then foregr := foregr + 16;
  TextColor(foregr);
  TextBackground(backgr);
END;

FUNCTION SET_MONITOR_TYPE: INTEGER;
VAR
  j: INTEGER;
BEGIN
  VAR
    v1, v2, v3: INTEGER;
  BEGIN
    if j = 2 then
      BEGIN
        v1 := $44;
        v2 := $45;
        v3 := $46;
      END;
  END;
END;
2011 v2:=$3b5;
2021 v3:=$3b9
2031 end;
2041 if (j=2) or (j=3) then
2051 begin
2061 port[v1]:=$0A; port[v2]:=0;  { Set High Cursor Scan Line }
2071 port[v1]:=$0B; port[v2]:=7;  { Set Low Cursor Scan Line }
2081 port[v3]:=1;  { Set Border Color toブルー }
2091 end;
2101 END;
2111
2121 BEGIN
2131 j:=mem[$40:$10];  { Figure out the monitor type }
2141 j:=j and $0030) DIV 16;
2151 CASE j OF
2161 0: begin writeln('Ill egal Monitor Mode'); halt end;
2171 1: begin  { Set 40 column color to 80 column color }
2181 writeln('Use MODE command to set to 80. (MODE ("40")'); halt
2191 end;
2201 2: videoSS:=ptr($4800,0);  { Graphics 80 }
2211 3: videoSS:=ptr($4000,0);  { Monochrome }
2221 END;
2231 voffSS:=$1; vonSS:=$29; vdispSS:=$3B8;  { Video Off, On, Location }
2241 CURSOR_SET;  { Set To A Large Cursor }
2251 COLOR(4,1);  { Set Default Color }
2261 SET_MONITOR_TYPE:=j;
2271 END;  { SET_MONITOR_TYPE }
2281
2291 2301 PROCEDURE DISPLAY_SCREEN (var screenfile: FILE);
2311  { Load Screen From Disk, Display To Monitor }
2321 VAR blood: array[1..3968] of CHAR;
2331 exist : Boolean;
2341
2351 PROCEDURE VIDE0_OFF;  { Turn Video Off }
2361 BEGIN port[vdispSS]:=voffSS; END;
2371
2381 PROCEDURE VIDE0_ON;  { Turn Video On }
2391 BEGIN port[vdispSS]:=vonSS; END;
2401 BEGIN
2411 if iOresult=0 then
2421 begin
2431 exist:=TRUE;
2441 blockread screenfile, blood[1, 3];
2451 VIDEO(FF);
2461 move [blood[0], "video", vdispSS];
2471 VIDEO(8);  
2481 end;
2491 else exist:=FALSE;
2501 if not exist then

175
Program Listing

```pascal
begin
  color (15, 4);
gotoxy (25, 11);
write ('G, 'Part of SPLICE.SCR is missing.');
end;
retSS := ';
END; { DISPLAY_SCREEN }

PROCEDURE GETITEM;

COL, LIN, LEN : BYTE;
 Vit : CHAR;
VAR WITEM : STRO8;
P ICT : STRO8;
ITEM LOW, ITEM HIGH : STRO8;
VAR RET : STR2;
RETRIEVE : BOOL8;
FGR_COLOR, DEC_COLOR : BYTE
);

TYPE

PICT TYPE = set of CHAR;

CONST

confirm=FALSE; ( If FALE auto-skip to next field when field is filled )
l='R'; r='R'; m='M'; u='U'; s='S'; b='R'; i='I'; u='U';

END OF _function _keys _

f2='<'; f3='>'; f4='='; f5='?';
f6='='; f7='A'; f8='B'; f9='C'; f10='D';


PROCEDURE FUNCTION DATE_CHECK (datevar: STR80); BOOLEAN;

VAR

VAR mm, dd, yy: STR2;

error: INTEGER;

BEGIN

END;
```
APPENDIX B: MAINTENANCE MANUAL  Page 40

SPLICE.PAS—include file SPLICE1.PAS  Program Listing

301 BEGIN
302 if ord(datevar[0])>8 then
303 DATE_CHECK:=FALSE
304 else
305 begin
306 ch_date:=TRUE;
307 mm:=copy(datevar,1,2);
308 dd:=copy(datevar,4,2);
309 yy:=copy(datevar,7,2);
310 val(mm,mm_error);
311 if (error<>0) or (mm<1) or (mm>12) then ch_date:=FALSE;
312 if ch_date then
313 begin
314 val(dd,dd_error);
315 if (error<>0) or (dd<1) or (dd>month_days(mm)) then ch_date:=FALSE;
316 end;
317 if ch_date then
318 begin
319 val(yy,yy_error);
320 if (error<>0) then ch_date:=FALSE;
321 end;
322 DATE_CHECK:=ch_date;
323 end;
324 END;
325
326 FUNCTION CHECK_DATE(DATE, DATE_LOW, DATE_HIGH: STRING): BOOLEAN;
327 { Check Validity If date and whether it falls between low and high }
328 { if low range date is higher than high range date then we assume }
329 { we crossed centuries eg. 09/09/84 to 01/01/10 }
330 { Also a null date is ignored }
331 const null = '/' / '/';
332 VAR
333 ch_date: BOOLEAN;
334 BEGIN
335 if DATE=nil then ch_date:=DATE_CHECK(DATE) else ch_date:=TRUE;
336 if ch_date and (DATE=nil) and (DATE_LOW=null) and (DATE_HIGH=null) then
337 begin
338 if ch_date then ch_date:=DATE_CHECK(DATE_LOW);
339 if ch_date then ch_date:=DATE_CHECK(DATE_HIGH);
340 begin
341 date:=copy(DATE,7,2)+copy(DATE,1,6);
342 date:=copy(date_low,7,2)+copy(date_low,1,6);
343 date:=copy(date_high,7,2)+copy(date_high,1,6);
344 if date>date-high then ch_date:=FALSE;
345 begin
346 if (date>date-high) or (DATE_HIGH=null) then ch_date:=FALSE;
347 end;
348 if date>date-low and (DATE_LOW=null) then ch_date:=FALSE;
349 end;
350 end;
APPENDIX B: MAINTENANCE MANUAL  Page 41

SPLICE.PAS-include file SPLICE1.PAS Program Listing

J51 I if ch_date then CHECK_DATE:=TRUE else begin CHECK_DATE:=FALSE; end;
J52 I END; [PROCEDURE CHECK_DATE]
J53 I
J54 I FUNCTION CHECK_RANGE(VAR item, item_low, item_high: STR80): BOOLEAN;
J55 I (Check to see whether item is within and including low and high )
J56 I VAR item, low, high: REAL;
J57 I error1, error2, error3: INTEGER;
J58 I BEGIN
J59 I CHECK_RANGE:=TRUE;
J60 I val(item_low,low, error1);
J61 I val(item_high,high, error2);
J62 I val(item,item, error3);
J63 I if (error1=0) and (error2=0) and (error3=0) then
J64 I begin
J65 I if item<low then CHECK_RANGE:=FALSE
J66 I else if item>high then CHECK_RANGE:=FALSE;
J67 I end else
J68 I CHECK_RANGE:=FALSE;
J69 I END; [PROCEDURE CHECK_RANGE ]

J70 1
J71 1 PROCEDURE MESSAGE(mess_num: BYTE);
J72 I (Displays A Message On Line 25 and sets global clear25 to TRUE )
J73 I VAR mess, temp_item: STR79; mess_length, start_col: INTEGER;
J74 I BEGIN
J75 I color (14,1); gotoxy (1, 25); clreol;
J76 I case mess_num of
J77 I 1: mess:= 'Only 0 thru 9 Allowed ';
J78 I 2: mess:= 'Only 0 thru 9 or a space Allowed ';
J79 I 3: mess:= 'BAD Date OR Not Within '+item_low+ '& '+item_high+ ' Range ';
J80 I 4: mess:= 'Number Not Within '+item_low+ '& '+item_high+ ' Range ';
J81 I 5: mess:= 'Only 0 thru 9, decimal point OR - sign Allowed ';
J82 I 6: mess:= 'Only Y or N Allowed ';
J83 I 7: mess:= 'Only M or F Allowed ';
J84 I 8: mess:= 'No More Room For Digits, Use [Del] key to remove ';
J85 I 9: mess:= 'No Space For Negative Numbers, Input Positions Must Be Large ';
J86 I end; [ case ]
J87 I mess_length:=ord(mess[0]);
J88 I start_col:=(79-mess_length) DIV 2;
J89 I clear25:=TRUE;
J90 I gotoxy(start_col,25);
J91 I COLOR(15, 4);
J92 I write('G', mess);
J93 I gotoxy(color,lin);
J94 I COLOR (14, 1);
J95 I CLEAR FLD;
J96 I END; [MESSAGE PROCEDURE]

J98 I
J99 I FUNCTION CHECKRANGE(VAR item, item_low, item_high: STR80): BOOLEAN;
J100 I (If CHECKRANGE TRUE on return then returns end of function otherwise)

178
APPENDIX B: MAINTENANCE MANUAL

Page 42

Page 9

SPLICE.PAS-include file SPLICE1.PAS Program Listing

4011 {if GETCHAR=FALSE on return then kchar is alpha numeric chars}
4021 {ctype must be one of the following}
4031 {U=Uppercase, L=Lower Case, X=Any Char, 9=0..9, ', #=0..9,-,...}
4041 {GETCHAR will filter out any control characters}
4051 TYPE PICT_TYPE = set of CHAR;
4061 CONST esc = 27; cr = 13; bk = 8;
4071 l='X'; r='W'; u='H'; d='P'; dl='S'; ins='R'; ps='I'; pd='Q';
4081 f1='K'; f2='X'; f3='A'; f4='E'; f5='G'; f6='C'; f7='H'; f8='B'; f9='C'; f10='D';
4091 special_keys: PICT_TYPE = {l,r,u,d,dl,ins,ps,pd};
4101 func_keys: PICT_TYPE = {f1,f2,f3,f4,f5,f6,f7,f8,f9,f10};
4111 var func: PICT_TYPE = [f1,f2,f3,f4,f5,f6,f7,f8,f9,f10];
4121 var str: CHAR; correct: BOOLEAN;
4131 temps: STRING;
4141 BEGIN
4151 kchar:='';
4161 GETCHAR:=TRUE; {correct:=FALSE;}
4171 repeat [until getchar = TRUE]
4181 special:=TRUE;
4191 repeat [until a valid picture character]
4201 repeat until keypressed;
4211 read(kbd,kchar1);
4221 if keypressed and (kchar1=chr(esc)) then
4231 begin
4241 read(kbd,kchar2);
4251 kchar(1):=chr(0);
4261 kchar(0):=chr(2);
4271 end else
4281 kchar(0):=chr(1);
4291 (Clear Line 25)
4301 if clear25 then
4311 begin
4321 color (14, 1);
4331 gotoxy (1,25);
4341 clrscr;
4351 gotoxy(hcol,lin);
4361 clear25:=FALSE;
4371 color (FGR_COLOR, BGR_COLOR);
4381 end; (Clear Line 1)
4391 if (not (ord(kchar[1]) in [esc,cr,bk])) and (ord(kchar[0])=1) then
4401 begin
4411 str:=kchar[1];
4421 if (str=':') and (str[2]='') then
4431 new_type of
4441 'U': correct:=TRUE;
4451 'A': end;
4461 if str in ['1'..'9'] then str:=chr(ord(str)+49);
4471 kchar[1]:=str; correct:=TRUE;
4481 end;
4491 'U': begin
4501 if str in ['A','Z'] then str:=chr(ord(str)+39);
SPLICE.PAS—include file SPLICE1.PAS Program Listing

Program SPLICE;

var
    kchar: string;
    correct: boolean;

begin
    kchar[1] := str; correct := true;
end;

'9': if str in ['0', '9', '.', ''] then correct := true else message(5);

'S': if str in ['0', '9', '. '] then correct := true else message(1);

t case
    end

else
    begin
        getChar := false;
        correct := true;
        str := kchar[1];
    end;

until correct;

if (ord(kchar[0]) = 2) then
    begin
        special := false;
        getChar := true;
        if (kchar[2] in special_keys) or (kchar[2] in func_keys) then
            begin
                getChar := false;
                special := true;
            end;
        end else
            begin
                beep(keybs);
            end;

until special;

get:

PROCEDURE DECII; { Positions Cursor At the Next Non Edit Character }

VAR
    elem_end: boolean;
    tempb: byte;

BEGIN
    if hcol > (col + tcol - 1) then begin
        tempb := pcol;
        elem_end := false;
        repeat
            tempb := tempb - 1;
            if (pict[tempb] in pict_elements) or (tempb < 0) then elem_end := true;
            until elem_end;
        end;
        if tempb = 1 then
            begin
                hcol := hcol - (pcol - tempb);
                pcol := tempb;
                end;
    end else
        begin_of_line := true;
    end; { DECII PROCEDURE }

PROCEDURE INCH; { Positions Cursor At the Next Non Edit Character }

VAR
    elem_end: boolean;
    tempb: byte;

BEGIN

END; { INCH PROCEDURE }

180
APPENDIX B: MAINTENANCE MANUAL  Page 44

SPLICE.PAS include file SPLICE1.PAS Program Listing

5011 BEGIN
5021 if hcol>(col+len-1) then
5031 begin
5041 tempbl:=1;
5051 elem_end:=FALSE;
5061 repeat
5071 if (pict[pool+tempbl-1] in pict_elements) or ((pool+tempbl)>(len)) then
5081 elem_end:=TRUE;
5091 until elem_end;
5101 if tempbl<(len) then
5111 begin
5121 hcol:=hcol+tempbl-1;
5131 pool:=pool+tempbl-1;
5141 end;
5151 end else
5161 end_of_field:=TRUE;
5171 END; [END PROCEDURE ]
5181
5201 PROCEDURE STRIP_BLANKS(VAR temp_item: STRING);
5202 [Strip Blanks On Both Sides Of passed item ]
5203 VAR i,j:BYTE;
5204 BEGIN
5205 if temp_item=' ' then
5206 begin
5207 j:=ord(temp_item[0]);
5208 i:=0;
5209 while (temp_item[i+1]=' ') and (i<j) do i:=i+1;
5210 if (i>0) and (i<j) then temp_item:=copy(temp_item,i+1,j-1)
5211 else if (i=j) and (temp_item[j]=' ') then temp_item=' ';
5212 i:=pos(' ',temp_item);  // strip trailing blanks
5213 if i=0 then temp_item:=copy(temp_item,1,i-1);
5214 end;
5215 END; [END PROCEDURE ]
5216
5301 BEGIN [Main Procedure Of GETITEM ]
5302 item:=item1;  // Store Actual Item In A Work Variable
5303 clear25:=FALSE;
5304 if item='D' then
5305 begin
5306 pict:='88/88/88';
5307 len:=8;
5308 end;
5309 if item='Y' then
5310 begin
5311 if not (item[1] in ['Y','B']) then item:='Y';
5312 pict:='0';
5313 len:=1;
5314 end;
5511 if itype='M' then
5521 begin
5531 if not (item[1] in ['M','F']) then item:=M;
5541 pict:=U;
5551 len:=1;
5561 end;
5571 end_of_field:=FALSE;
5581 begin_of_field:=FALSE;
5591 if (pict='') and (itype='C') then pict:=X';
5601 plen:=ord(pict[0]);
5611 fchar:=pict[plen];
5621 item:=ord(item[0]);
5631 (* Fill Item with blanks *)
5641 if itype<>N' then (* If item is non numeric *)
5651 begin
5661 while ilen<len do
5671 begin
5681 if item=' then item:=0;
5691 ilen:=ord(item[0]);
5701 end;
5711 while plen<len do
5721 begin
5731 pict:=pict+fchar;
5741 plen:=pln+1;
5751 end;
5761 end else (* If item is numeric *)
5771 begin
5781 strip_blanks(item);
5791 if item=' then item:=0;
5801 ilen:=ord(item[0]);
5811 while ilen<len do
5821 begin
5831 item:=*item;
5841 ilen:=ilen+1;
5851 end;
5861 while plen<len do
5871 begin
5881 pict:=pict+fchar;
5891 plen:=pln+1;
5901 end;
5911 if ord(pict[0])<ilen then pict:=copy(pict,ord(pict[0])-len+1,len);
5921 if ord(item[0])<ilen then item:=copy(item,1,ilen);
5931 if pict_decr<ilen then pict:=ord(pict[0]);
5941 pict_decr:=ord(pict[0]);
5951 item_decr:=ord(item[0]);
5961 (* Align Decimal Positions If Necessary *)
5971 if pict_decr>item_decr then
5981 begin (alignment)
5991 check:=TRUE;
6001 (* If picture has no decimal point and item does)
APPENDIX B: MAINTENANCE MANUAL

SPLICE.PAS—include file SPLICE1.PAS Program Listing

601 I if (pict_dec = 0) and check then
602 I begin
603 I item:=copy(item,1,item_dec-1);
604 I fillchar(temp_item,ord(pict[0])-ord(item[0]),' ');
605 I item:=temp_item+item;
606 I check:=FALSE;
607 I end;
608 I { If item has no decimal point and pict does}
609 I if (item_dec = 0) and check then
610 I begin
611 I strip_blanks(item);
612 I tempb2:=plen-pict_dec; { # of decimal points };
613 I fillchar(temp_item,tempb2,item[ord(item[0])]);
614 I item=item+'.'+temp_item; { Add decimal trailing digits }
615 I ilen:=ord(item[0]); { Get length of item }
616 I while ilen<plen do { Add blanks left }
617 I begin
618 I item=' '+item;
619 I ilen:=ilen+1;
620 I end;
621 I if ilen>plen then { If The Item > Picture }
622 I begin
623 I item:=copy(item,1,pict_dec-1);
624 I item:=item+'.'+temp_item;
625 I end;
626 I check:=FALSE;
627 I end;
628 I { If item decimal is further right than pict dec}
629 I if (item_dec>pict_dec) and check then
630 I begin { Move the item to the left dropping off numbers picts}
631 I plen:=ord(pict[0]);
632 I ilen:=ord(item[0]);
633 I item:=copy(item,1,item_dec-pict_dec+1,ilen-(item_dec-pict_dec));
634 I item:=item+'.'+temp_item;
635 I tempbl:=plen-ord(item[0]);
636 I fillchar(temp_item,tempbl,item[ilen]);
637 I item:=item+temp_item;
638 I ilen:=ord(item[0]);
639 I while ilen<plen do { Add blanks left }
640 I begin
641 I item=' '+item;
642 I ilen:=ilen+1;
643 I end;
644 I check:=FALSE;
645 I end;
646 I { If pict decimal is further right than item's}
647 I if (pict_dec>item_dec) and check then
648 I begin
649 I tempb2:=plen-pict_dec;
650 I item:=copy(item,1,item_dec+tempb2);
6511  ilen:=ord(item[0]);
6521  while ilen<len do
6531     begin
6541       item:= ' ' + item;
6551       ilen:=ilen+1;
6561     end;
6571  check:=FALSE;
6581  end;
6591  end [ alignment ];
6601  end [ fillings];
6611  (* Copy edit characters to item *)
6621  for tempbl:=1 to len do
6631     if not (pict[tempbl] in pict_elements) then item[tempbl]:=pict[tempbl];
6641  (* Display The item on the screen *)
6651  gotoxy(col,lin);
6661  writec(item);
6671  (* Get Data From Screen If Retrieve is True)
6681  if retrieve then
6691  begin [ Retrieve ]
6701  (* Move cursor to first position by bypassing edit chars *)
6711  pcoll:=1;
6721  while (not (pict[pcol] in pict_elements)) and (pcol=len) do pcol:=pcol+1;
6731  (* Readjust column *)
6741  tcol:=pcol;
6751  if (itype<> 'N') and (pcol=len) then
6761  (* Handle Non Numeric Type Of Item *)
6771  begin [ Non Numeric Field ]
6781  repeat
6791  end of field:=FALSE;
6801  begin of field:=FALSE;
6811  special:=FALSE;
6821  if getchar(pict[pcol],kchar) then
6831  begin [ Non Numeric Field ]
6841  writec(kchar);
6851  item[pcol]:=kchar[1];
6861  gotoxy(hcol,lin);
6871  end else
6881  special:=TRUE;
6891  if special then
6901  begin [ Special Key Pressed ]
6911  ret:=kchar;
6921  special:=FALSE;
6931  if kchar[1]=chr(bk) then { It is backspace }
begin

dech;

gotoxy(hcol,lin); {Left}
end else
if (ord(kchar[0])=2) and (kchar[2] in [l,r,dl,ins]) then
begin
    case kchar[2] of
    l: begin dech; gotoxy(hcol,lin); end; {Left}
    r: begin inch; gotoxy(hcol,lin); end; {Right}
    dl: begin (Delete)
        tempb2:=pcol+1; {Find where the next edit char starts}
        while (pict[tempb2] in pict_elements) and (tempb2<=len) do
            { tempbl=start, tempb2=end}
            tempb2:=tempb2+1;
            tempb2:=tempb2-1;
        for tempb1:=pcol to tempb2-1 do (move chars left)
            begin [ & put blank at end]
                item[tempb1]:=item[tempb1+1];
                end;
        item[tempb2]:='';
        (rewrite the item)
        gotoxy(col,lin);
        writec(item);
        gotoxy(hcol,lin);

    end;
    ins: begin (Insert)
        tempb2:=pcol+1;
        while (pict[tempb2] in pict_elements) and (tempb2<=len) do
            { tempbl=start, tempb2=end}
            tempb2:=tempb2+1;
            tempb2:=tempb2-1;
        for tempb1:=tempb2 downto pcol+1 do
            begin
                item[tempb1]:=item[tempb1-1];
            end;
        item[pcol]:='';
        gotoxy(col,lin);
        writec(item);
        gotoxy(hcol,lin);

    end;
end
else (esc,cr,µup,pydn,up,dn)
    special:=TRUE;
end {if backspace };
if end_of_field or begin_of_field then BEEP(DeepOnSS);
until (end_of_field and (not confirm)) or begin_of_field or special;
tempbl:=len; { Strip Trailing Blanks }
if itype='C' then
    while (item[tempb1]=' ') and (tempb1>0) do tempb1:=tempb1-1;
    item[0]:=chr(tempb1);
range_check:=TRUE;
begin
  range_check:=check_date(item, item_low, item_high);
  if not range_check then message[3];
end;
if itype='Y' then
begin
  if not (item[1] in ['Y','N']) then
  begin
    range_check:=FALSE;
    message[6];
  end;
if itype='M' then
begin
  if not (item[1] in ['M','F']) then
  begin
    range_check:=FALSE;
    message[7];
  end;
end;
until range_check;
end
/* If non numeric type of item */
else /* if Numeric */

if (itype='N') then
begin
  tcol:=len;
  repeat ( Until range_check=TRUE )
  len:=tcol;
  tempbl:=len;
  len:=pos('.',item);
  range_check:=FALSE;
  if len=0 then len:=tempbl
else len:=len-1; /* Item has decimal point */
  hcol:=col+len-1;
  pcol:=len;
  gotoxy(hcol,lin);
  special:=FALSE;
  sign_flag:=FALSE;
  end_of_field:=FALSE;
  dec_flag:=FALSE;
  repeat
  valid_char:=FALSE;
  if yetchar('#',kchar) then
begin
  case kchar of
    ',' : ( Sign ) if not sign_flag then valid_char:=TRUE;
    '.' : ( Decimal point )
    if (len=tempbl) and (not dec_flag) then
begin
  hcol:=hcol+2; pcol:=len+2; gotoxy(hcol,lin);
  dec_flag:=TRUE; sign_flag:=TRUE;
end;
end;
'0'..'9' : valid_char:=TRUE;
APPENDIX B: MAINTENANCE MANUAL

Page 17

SPLICE.PAS—include file SPLICE1.PAS Program Listing

8011 end (Case kchar);
8021 (sign_flag = if FALSE we allow minus (-) sign)
8031 (dec_flag = if FALSE we allow decimal (.) point)
8041 if (valid_char) and (not dec_flag) then (Integer Portion)
8051 begin
8061 if (item[1]=' ') and (len>tempbl) and (sign_flag) and
8071 not ((ord(item[0])<1) and (item[1]='-' and (item[2]='0'))
8081 message(8) (Overflow Numeric Field)
8091 else begin
8101 if (not sign_flag) then (Erase Old Entry, Start New One)
8111 begin
8121 if (len-1)<>0 then
8131 begin
8141 message(9);
8151 sign_flag:=FALSE;
8161 end else begin
8171 item[len-1]:='-
8181 item[len]:='0';
8191 end;
8201 end else begin
8211 item[len]=kchar;
8221 gotoxy(col,lin);
8231 writec(item);
8241 gotoxy(hcol,lin);
8251 end else begin
8261 gotoxy(col,lin);
8271 writec(item);
8281 gotoxy(hcol,lin);
8291 end
8301 if (item[1]='-' then
8311 if (len-1)<0 then
8321 begin
8331 message(9);
8341 sign_flag:=FALSE;
8351 end else begin
8361 item[len-1]:='-';
8371 item[len]:='0';
8381 end;
8391 end else begin
8401 gotoxy(col,lin);
8411 writec(item);
8421 gotoxy(hcol,lin);
8431 end else begin
8441 gotoxy(col,lin);
8451 writec(item);
8461 gotoxy(hcol,lin);
8471 end;
8481 if (item[1]='-') and (len>tempbl) then end_of_field:=TRUE;
8491 end;
8501 end (Integer Portion)
8511 else { Decimal Portion }  
8521 if valid_char and (sign_flag) then  
8531 begin  
8541 item[pcol]:=kchar[J;  
8551 writec(item[pcol]);  
8561 if not end_of_field then  
8571 begin  
8581 hcol:=hcol+1;  
8591 pcol:=pcol+1  
8601 end;  
8611 if pcol>tempbl then  
8621 begin  
8631 hcol:=hcol-1;  
8641 pcol:=pcol-1;  
8651 end_of_field:=TRUE  
8661 gotoxy(hcol,lin);  
8671 end;  
8681 end  
8691 end { getchar is FALSE } else { getchar is TRUE }  
8701 special:=TRUE;  
8711 { Special Keys: DEL }  
8721 if special then  
8731 begin  
8741 ret:=kchar;  
8751 special:=FALSE;  
8761 if ord(kchar[0])=2 then  
8771 begin  
8781 case kchar(2) of  
8791 [ DELETE KEY PRESSED OR LEFT ARROW KEY ]  
8801 case dec_flag of  
8811 False: { Integer Portion }  
8821 begin  
8831 sign_flag:=TRUE;  
8841 for pcol:=len downto 2 do item[pcol]:=item[pcol-1];  
8851 if (item[len] in [',','-']) then  
8861 begin  
8871 item[len]:=,';  
8881 sign_flag:=FALSE;  
8891 end;  
8901 item[1]:='';  
8911 gotoxy(col,lin);  
8921 writec(item);  
8931 gotoxy(hcol,lin);  
8941 end_of_field:=FALSE;  
8951 end { TRUE }  
8961 { Put 0 in Cursor. Check If Going To Integer Part }  
8971 if pict[pcol-1]=',' then { Are We in Integer Part?}  
8981 begin { YES. Initialize Variables}  
8991 hcol:=col-len-1;
APPENDIX B: MAINTENANCE MANUAL

SPLICE.PAS—include file SPLICEI.PAS Program Listing

901 gotoxy(hcol,lin);
902 dec_flag:=FALSE;
903 end_of_field:=FALSE;
904 end else
905 begin
906 if not end_of_field then
907 begin
908 hcol:=hcol-1;
909 pcol:=pcol-1
910 end;
911 gotoxy(hcol,lin);
912 item[pcol]:='0';
913 writec(item[pcol]);
914 gotoxy(hcol,lin);
915 end_of_field:=FALSE;
916 end;
917 } ( T )
918 end { dec_flag CASE };
919 u,d,l,r,pu,pd,f1,f2,f3,f4,f5,f6,f7,f8,f9,f10: special:=TRUE;
920 end; { DELETE KEY CASE }
921 end { Case } else
922 if (ord(kchar[1]) in [cr, esc]) then special:=TRUE;
923 end | Special |)
924 if end_of_field and (not special) then BEEP(DeepOnSS);
925 until special or (end_of_field and (not confirm));
926
927 { Get Old Length back and find point position }
928 len:=tcol;
929 pcol:=pos('.',pict);
930 |
931 if no decimal point and 1st position is minus or blank then set to 0 |
932 if (item[len] in ['-',']) and (pcol=0) then
933 begin
934 item[len]:='0';
935 gotoxy(coi,lin);
936 writec(item);
937 end;
938
939 temp_item:=item;
940 strip_blanks(item);
941 range_check:=check_range(item,item_low,item_high);
942 if not range_check then
943 begin
944 message(4);
945 item:=temp_item;
946 end;
947 until range_check;
948 end;( Numeric )
949 end | Retrieve | else
APPENDIX B: MAINTENANCE MANUAL  Page 53

SPLICE.PAS - include file SPLICE1.PAS Program Listing

951begin
952if itype='N' then strip_blanks(item);
953if itype='C' then
954begin
955  tempbl:=len; { Strip Trailing Blanks }
956  while (item[tempbl]='#') and (tempbl>0) do tempbl:=tempbl-1;
957  item[0]:=chr(tempbl);
958end;
959end;
960item:=item; { Return result Back To item }
961END;{ GETITEM PROCEDURE}
962
963(* This is a summary of the procedures in SPLICE1.PAS
964PROCEDURE BEEP(BeepOn: BOOLEAN); { Sound Beep if BeepOn=TRUE }
965PROCEDURE CLEAR_KBD; { Clear Keyboard Buffer }
966PROCEDURE COLOR(foregr,backgr:BYTE); { Set Color }
967PROCEDURE WRITEC(text: STR80); { Write Chars Using Color }
968FUNCTION SET_MONITOR_Type: INTEGER; { Determine Monitor Type }
969PROCEDURE DISPLAYSCREEN(screenname: STR80); { Display A Screen Sculptor Screen }
970PROCEDURE GETITEM(COL,LIN,LEN:BYTE; Column, Line, Length }
971ITYPE : CHAR; { Type= C, N, D, Y, M }
972Var WITEM : STR80; { Variable Name }
973PICT : STR80; { Picture X, L, 9, 8 # }
974ITEM_LOW,ITEM_HIGH : STR80; { Range - Numerics/Date Only}
975Var RET : STR2; { Returned Code }
976RETRIEVE : BOOLEAN; { False=Disp Only, True=Get }
977FCOLOR,BCOLOR : BYTE { Colors Foregr, Backgr }
978); EXTERN;
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000PROCEDURE ACCEPT_INPUTS;
1001begin { Procedure ACCEPT_INPUTS }
SPLICE.PAS Program Listing

1001  GOTOXY (1, 25);  { Position cursor col 1, row 25 }
1002  CiEol;  { Clear row 25 with blanks }
1003  WRITE (' Do you accept the input values thus far?  Yes or No ');
1004  answerSS := 'N';
1005  GETITEM (70, 25, 1, 'Y', answerSS, 'U', '', retSS, True, 12, 1);
1006  GOTOXY (1, 25);  { Position cursor col 1, row 25 }
1007  TextBackground (1);  { Set background color to BLUE }
1008  CiEol;  { Clear row 25 with blanks }
1009  End;  

PROCEDURE ACCEPTINPUTS;
1010
1011  Check Status Of Variable retSS and return a code in 'actionSS' & set 'varSS'
1012  This procedure is called immediately following GETITEM !
1013
1014  Input to this procedure:
1015  when retSS is length 1 the values are any of the ASCII chars
1016  when retSS is length 2 the values are uSS, lSS, puSS, pdSS, function keys
dSS, rSS
1017
1018  Output:
1019  The following codes are returned in actionSS : nextSS, prevSS,
1020  exitSS, staySS ;
1021
1022  Based upon 'actionSS' this procedure will then set 'varSS' to an integer,
1023  which represents the next item (variable ) to get. 
1024
1025  Begin
1026  
1027  last_field_actionSS := exitSS;
1028  actionSS := nextSS;  { Initialize Action Code }
1029  IF retrieveSS THEN  { is retrieveSS TRUE? }
1030     Begin
1031        IF ord (retSS[1]) = 2 THEN  { Is retSS length 2 ? }
1032            Begin
1033               CASE retSS[2] of
1034                  uSS, lSS : actionSS := prevSS;  { Up Key, Left Key }
1035                  dSS, rSS : actionSS := nextSS;  { Down Key, Right Key }
1036                  puSS : actionSS := staySS;  { Page Up }
1037                  pdSS : actionSS := staySS;  { Page Down }
1038                  f1SS, f2SS, f3SS, f4SS, f5SS, f6SS, f7SS, f8SS, f9SS, f0SS : actionSS := staySS;
1039               End  { Case retSS[2] }
1040
1041               End;  { IF ord (retSS[1]) = 2 THEN }
1042
1043               CASE retSS[1] of
1044                  uSS, lSS : actionSS := prevSS;  { Up Key, Left Key }
1045                  dSS, rSS : actionSS := nextSS;  { Down Key, Right Key }
1046               End;  { Case retSS[1] }
1047               IF retSS[2] = 'Y' THEN actionSS := staySS;  { Escape Key }
1048            End;
1049
1050            End;  { IF retrieveSS }

191
APPENDIX B: MAINTENANCE MANUAL

Page 22

SPLICE.PAS Program Listing

1051 CASE actionSS of
1052 staySS: ;
1053 nextSS: Begin
1054 varSS := varSS + 1;
1055 IF varSS > screen_fieldSS THEN varSS := 1;
1056 IF last_fieldSS AND retrieveSS THEN
1057 actionSS := last_field_actionSS
1058 End;
1059 prevSS: Begin
1060 varSS := varSS - 1;
1061 IF varSS < 1 THEN varSS := screen_fieldSS
1062 End;
1063 exitSS: ;
1064 End; { CASE }
1065 End; { PROCEDURE RET_STATUS}
1066 PROCEDURE GETREAL(COL, LIN, LEN : BYTE;   { Column, Line, Length }
1067 TYPE : CHAR; { Type= C, N, D, Y, M }
1068 Var WITEM : REAL; { Numeric Variable Name }
1069 PICT : STR80; { Picture X, U, L, 9, 8 # }
1070 ITEM_LOW,ITEM_HIGH : REAL; { Range - Numerics/Date Only }
1071 Var RET : STR2; { Returned Code }
1072 RETRIEVE : BOOLEAN; { False=Disp Only, True=Get }
1073 FGR_COLOR,BGR_COLOR : BYTE; { Colors Foregr, Backgr }
1074 { This Procedure converts numeric to string before calling GETITEM }
1075 { It then converts the result back to numeric }
1076 Var
1077 numSS, numloSS, numhiSS: STR80;
1078 errorcodeSS, decposSS: INTEGER;
1079 Begin
1080 decposSS:=ord(pict[10])-ord('.','pict');
1081 { Convert item, low and high range to string }
1082 STR (witem:0:decposSS,numSS);
1083 STR (item_low:0:decposSS,numloSS);
1084 STR (item_high:0:decposSS,numhiSS);
1085 GETITEM (col,lin,len,itype,numSS, pict, numloSS, numhiSS,
1086 RET, retrieves,fgr_color,bgr_color);
1087 { Convert string to numeric item }
1088 VAL (numSS, witem, errorcodeSS);
1089 End; { Procedure GETREAL }

1098 PROCEDURE GETINT(COL, LIN, LEN : BYTE;   { Column, Line, Length }
1099 TYPE : CHAR; { Type= C, N, D, Y, M }
1100 Var WITEM : INTEGER; { Numeric Variable Name }
APPENDIX B: MAINTENANCE MANUAL

SPLICE.PAS Program Listing

```
1101 PICT: STR80;              { Picture X, U, L, 9, 8 # }
1102 ITEM_LOW, ITEM_HIGH: INTEGER;       { Range - Numerics/Date Only }
1103 Var RET: STR2;             { Returned Code }
1104 RETRIEVE: BOOLEAN;       { False=Disp Only, True=Get }
1105 FGR_COLOR, BGR_COLOR: BYTE;      { Colors Foregr, Backgr }
1106{ This Procedure converts numeric to string before calling GETITEM }
1107{ It then converts the result back to numeric }
1108Var
1109numSS, numloSS, numhiSS: STR80;
1110errcodeSS: INTEGER;
1111Begin
1112{ Convert item, low and high range to string }
1113STR (item, low, numSS);
1114STR (item, low, numloSS);
1115GETITEM (col, lin, len, itype, numSS, pict, numloSS, numhiSS, 
1116     ret, retrieve, fgr color, bgr color);
1117{ Convert string to numeric item }
1118VAL (numSS, item, errcodeSS);
1119End;   { Procedure GETINT }
1120PROCEDURE LINE_SETUP;
1121Var
1122Temp1: String[2];
1123Temp2: String[4];
1124Begin{ PROCEDURE LINE_SETUP }
1125IF Siteinfo.siteno < 10 THEN 
1126   STR (Siteinfo.siteno:1, Temp1)
1127ELSE
1128   STR (Siteinfo.siteno:2, Temp1);
1129   Temp2 := Copy (Costtable [1].clin, 1, 4);  { Line Number. (CLIN) }
1130   IF Siteinfo.siteno < 10 THEN 
1131     Line_Number := CONCAT ('0', Temp1, Temp2)   { Build the Contract }
1132     ELSE 
1133     Line_Number := CONCAT ('0', Temp1, Temp2) 
1134     END;
1135     END;
1136     { Accumulate the three totals for each section }
1137     ******************************************
```

193
Subtotals[ORD(mode),1] := Subtotals[ORD(mode),1] + Extended_Price;

IF Mode = Hard THEN
Subtotals[ORD(mode),2] := Subtotals[ORD(mode),2] + (Quantity * Costtable[I].basemain
* Maint_Factor * Maint_Months);
ELSE
IF Mode = Soft THEN
Subtotals[ORD(mode),2] := Subtotals[ORD(mode),2] + (Costtable[I].basemain
* Maint_Factor * Maint_Months);
ELSE
Subtotals[ORD(mode),2] := Subtotals[ORD(mode),2] + (Costtable[I].basecost * Quantity);
ELSE
Subtotals[ORD(mode),3] := Subtotals[ORD(mode),3] + (Costtable[I].basecost * Maint_Factor * Maint_Months);

[-----------------------------]
(* Accumulate the O&M and O&P totals for each section *)
[-----------------------------]

IF (Mode = Hard) OR (Mode = Soft) THEN (Add to O&M Total)
Totals[ORD(mode),2] := Totals[ORD(mode),2] + Extended_Price;
ELSE (Add to O&M Total)
Totals[ORD(mode),1] := Totals[ORD(mode),1] + Extended_Price;
End; (* Procedure LINE_SETUP *)

*************************************************************
| This procedure generates the headers which are written at the top of |
| each section of the delivery order.                                |
*************************************************************

Begin (Procedure HEADERS )

WRITELN (Diskfile, " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , " " , 

End; (* Procedure HEADERS *)

194
PROCEDURE WRITE_A_LINE;

****************************************************************
This procedure is called by two disk file print routines, PRINT_MAINT and PRINT_DOC_or_TRNG to write the data elements associated with each CLIN to the output disk file.
****************************************************************

Begin ( Procedure WRITE_A_LINE )
   LINE_SETUP;
End; ( Procedure WRITE_A_LINE )

PROCEDURE PRINT_DOC_or_TRNG;

****************************************************************
( Sets Parameters for FUC Training Courses and Documentation. )
( Sets both Maint_Months and Maint_Factor to zero (0) )
****************************************************************

Begin ( Procedure PRINT_DOC_or_TRNG )
   Maint_Months := 0; ( No maintenance on training/documentation )
   Maint_Factor := 0; ( No maintenance uplift on training/documentation )
   Extended_Price := Quantity * Costtable[I].purchprice;
   WRITE_A_LINE;
End; ( Procedure PRINT_DOC_or_TRNG )

PROCEDURE COMPUTE_SECTION_TOTALS (Section_Title : Title);

( This procedure prints the totals accumulated for each section after the last contract line number and associated data elements are printed. It prints the title for the next section and prints a new set of headers. After the last contract line number and associated data elements have been printed, the OWMN and OPMN totals for each section and the OWMN and OPMN grand totals are printed. )

Var
   K : Integer;

   OWMN_Total, OPMN_Total, Maint_Totals : Real;
**APPENDIX B: MAINTENANCE MANUAL**

**Page 26**

SPLICE.PAS Program Listing

---

```pascal
1251 Begin (Procedure COMPUTE_SECTION_TOTALS )
1252 {**************************************************************}
1253 ( Add maintenance and installation costs for each section to O&MN )
1254 {******************************************************}
1255 ( section totals .)
1256 {**********************************}
1257 Totals [ ORD ( mode), 1 ] := Totals [ ORD ( mode), 1 ] + Subtotals [ ORD ( mode), 2 ]
1258 + Subtotals [ ORD ( mode), 3 ];
1259 IF Section_Title = 'Other' THEN
1260 {******************************}
1261 ( If processing the last section, check to see if "SITE POWER"
1262 ( PREPARATIONS" are to be included, then print the O&MN and O&M
1263 ( section totals and grand totals .) 
1264 {******************************}
1265 Begin
1266 ( Compute Total amount of funds associated with maintenance )
1267 Maint_Totals := Subtotals [0,2] * Subtotals [1,2] * Subtotals [4,2];
1268 ( Write Maintenance Section Totals. Also, write the Hardware
1269 ( and Software Section Totals. Show the total amount of funds
1270 ( required for Maintenance. )
1271 WRITELN (Diskfile);
1272 WRITELN (Diskfile, """, "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "`,"`,"`,\n```
```pascal
WRITELN (Diskfile, '""""""SUBTOTALS:"""", '""""ONN"""",
WRITELN (Diskfile);"""", '""""HARDWARE"", Totals [0, 1],
WRITELN (Diskfile, '""""SOFTWARE"", Totals [1, 1],
WRITELN (Diskfile, '""""DOCUMENTATION"", Totals [2, 1],
WRITELN (Diskfile, '""""HARDWARE"", Totals [2, 2];
WRITELN (Diskfile, '""""SOFTWARE"", Totals [3, 1], """", Totals [3, 2]);
WRITELN (Diskfile, '""""DOCUMENTATION"", Totals [4, 1], """", Totals [4, 2]);
WRITELN (Diskfile, '""""OTHER"", Totals [5, 1],
END
```

PROCEDURE INITIALIZE;

PROCEDURE INIT_TOTALS;

Var
  Row, Col : Integer;

{ Initialize the subtotals and totals for each section to zero. }

Begin  
  FOR Row := 0 to 5 DO
    FOR Col := 1 to 3 DO
      Subtotals [Row, Col] := 0;
      Total [Row, Col] := 0;
  End;

{ Initialize the following global components }

1 := 1;  { Global index counter }
System_Downtime_Component := 0;
Mode := Hard;
DeepOnSS := False;  { Set to TRUE if sound is desired }
vtypeSS := SET_MONITOR_TYPE;  { 2 = Color, 3 = Monochrome }
TextBackground color := 11;  { Initialize background color to BOLD }
ClrScr;  { Clear the input screen }
 ASSIGN (Screenfile, File3);  { User responsible for I/O error check }
RESET (Screenfile);  { System will check for I/O errors }

PROCEDURE OPENING_SCREEN;

{ This procedure displays the opening screen to the user. }

Begin  
  Procedure OPENING_SCREEN  
  DISPLAY_SCREEN (Screenfile);  { Display Screen }
  DELAY (1500);  
End;  { Procedure OPENING_SCREEN }

198
PROCEDURE PICK_A_SITE;

This procedure has four main functions. First, it determines the site to be configured. Then it obtains the effective date for the delivery order. It obtains the file name for the output file from this session. And finally, it builds the SITE.INFO array which contains site specific data from the CONFIG.SIT file.

Var

Datain : String [80];
Siteno, Element : Integer;
Err, Temp_Site : Integer;
Textin : Text;

PROCEDURE GET_SITE_NUMBER;

Begin
( Procedure GET_SITE_NUMBER )
( Initialize Variables To Default Values )
Siteno := 1;

Present the user with a list of the SPLICE sites by name and number.

screen_fieldSS := '1';
varSS := 1;
retrievesSS := FALSE;
last_fieldSS := FALSE;
DISPLAY_SCREEN (Screenfile); ( Display Screen )

REPEAT ( until answerSS = 'Y' )
( Display Items. Change retrievesSS to TRUE and INPUT items )
REPEAT ( until actionSS = exitSS )
REPTEAT
GETINT(69,24,'N',Siteno,'#',1,58,retrievesSS,14,11);
IF Siteno = 23 THEN
Begin
WriteXY (20, 25); Color (15, 4);
WHITE ('G,' Site INACTIVE and not available for selection ');
End;
UNTIL Siteno = 23;
IF varSS = screen_fieldSS THEN last_fieldSS := TRUE;
RET_STATUS; ( Check the code in "retSS", set "varSS" and "retrieveSS" )

( Check to see whether to switch retrievesSS to true )
IF last_fieldSS and (not retrieveSS) THEN
Begin
    retrieveSS := TRUE;
    last_fieldSS := FALSE;
    actionSS := staySS;
    varSS := 1;
End
ELSE
    last_fieldSS := FALSE;
UNTIL actionSS = exitSS;
ACCEPT_INPUTS;
UNTIL answerSS = 'Y';
End;  // Procedure GET_SITE_NUMBER

Begin  // Procedure PICK_A_SITE  
GET_SITE_NUMBER;
ASSIGN (Textin, File2);
RESET (Textin);
{ Initialize "TempSite" and "StockPoint" }  
Temp_Site := 0;
Stock_Point := ' ';
WHILE Not EOF (Textin) AND (Temp_Site < siteno) DO
{ Read the file "CONFIG.SIT" until the site number in the file is:
  equal to the site number input by the user.  }  
Begin
    READING (Textin, Datain);
    Val (Copy (Datain, 1, 1), Temp_Site, Err);
    IF siteno = Temp_Site THEN
    Begin
        { Builds the site information record }  
        Siteinfo.siteno := siteno;
        Siteinfo.sitename := Copy (Datain, 1, 80);
        Val (Copy (Datain, 81, 1), Siteinfo.documentation, Err);
        Val (Copy (Datain, 83, 1), Siteinfo.information, Err);
        Siteinfo.maint_options := Copy (Datain, 84, 1);
        Siteinfo.maint_response := Copy (Datain, 84, 1);
        Siteinfo.site_type := Copy (Datain, 84, 1);
        Val (Copy (Datain, 84, 1), Siteinfo.site_type_cost, Err);
        End;
    End;
    Stock_Point := Siteinfo.site_type;
    TRUE (Textin);
End;  // Procedure PICK_A_SITE  

PROCEDURE BUILD_COST_TABLE;

| This procedure's primary function is to build the COSTTABLE array. This |
| contains the identification data for each component from the COSTS.IN file |
| as well as cost/maintenance data, which is updated by the applicable up- |
| lift or discount factors. The array currently contains room for 200 |
| entries. |

Var

Textin : Text;  
Errorcode, Count : Integer;

LCN_Purch_Esc_Rate, LCN_Maint_Esc_Rate, Document_Esc_Rate : Real;
Purch_Esc_Rate, Install_Esc_Rate, Train_Esc_Rate : Real;
SPLICE_Sw_Maint_Esc_Rate, SPLICE_Sw_Purch_Esc_Rate : Real;

PROCEDURE GET_RATES;

This procedure serves three main functions: it obtains the name of the |
current user, then obtains all the escalation/discount rates, and finally |
several numbers of Maint_Months, which are used for maintenance calculations.

Var

Month_Index : String [12];

PRN_Name, Effective_Date : String [8];

Index, Position : Integer;

PROCEDURE INITIALIZE_RATES;

(Initialize Variables To Default Values)

Begin (Procedure INITIALIZE_RATES)

Purch_Esc_Rate := 0.00;
Purch_Esc_Rate := 0.00;
SPLICE_Sw_Maint_Esc_Rate := 0.00;
SPLICE_Sw_Purch_Esc_Rate := 0.00;

Emerg_Maint_Rate := 0.0;

FDC_Sw_Purch_Esc_Rate := 0.00;

LCN_Maint_Esc_Rate := 0.000;

SPLCE_Sw_Esc_Rate := 0.000;

Install_Esc_Rate := 0.000;

Train_Esc_Rate := 0.00;

Document_Esc_Rate := 0.00;

Maint_Esc_Rate := 0.000;
APPENDIX B: MAINTENANCE MANUAL

SPLICE.PAS Program Listing

1601 10: GETREAL(71,17,4,'N',Train_Esc_Rate,
1602 11: '#.##',0.00,9.99,retSS,retrieveSS,15,3);  
1603 12: GETREAL(70,18,5,'N',Document_Esc_Rate,
1604 13: '#.##',-1.00,9.99,retSS,retrieveSS,15,3);  
1605 14: GETREAL(70,19,5,'N',Maint_Esc_Rate,
1606 15: '#.##',0.000,9.999,retSS,retrieveSS,15,3);  
1607 16: GETITEM(63,21,8,'C',PRN_NAME,
1608 17: 'XXXXXXXXX',retSS,retrieveSS,15,3);  
1609 18: GETITEM(37,23,2,'N',Maint_Months,
1610 19: '#',0,12,retSS,retrieveSS,15,3);  
1611 20: GETITEM(67,23,8,'D',Effective_Date,
1612 21: '88/88/88',retSS,retrieveSS,15,3);  
1613 End;  ( CASE )
1614 IF varSS = screen_fieldSS THEN last_fieldSS := TRUE;
1615 RET_STATUS;  ( Check code in "retSS". Set "varSS" & "actionSS" )
1616 IF last_fieldSS AND (not retrieveSS) THEN
1617 begin
1618 retrieveSS := TRUE;
1619 actionSS := staySS;
1620 varSS := 1;
1621 UNTIL actionSS = exitSS;
1622 ACCEPT_INPUTS;
1623 IF answerSS = 'Y'
1624 End;  ( Procedure GET_RATE_INPUTS )
1625 Begin  ( Procedure GET_RATES )
1626 INITIALIZE_RATES;
1627 GET_RATE_INPUTS;
1628 Generate the correct escalation & discount rates ;
1629 Document_Esc_Rate := FC_Run_Purch_Esc_Rate + 1;
1630 Purch_Esc_Rate := 1 + Purch_Esc_Rate;
1631 LCN_Purch_Esc_Rate := 1 + LCN_Purch_Esc_Rate;
1632 SPLICE SW Maint Esc_Rate := SPLICE SW Maint Esc_Rate + 1;
1633 SPLICE SW Run_Purch_Esc_Rate := SPLICE SW Purch_Esc_Rate + 1;
1634 Install Esc_Rate := 1 + Install Esc_Rate;
1635 Document Esc_Rate := 1 + Document Esc_Rate;
1636 Maint Esc_Rate := Maint Esc_Rate + 1;
1637 Train Esc_Rate := 1 + Train Esc_Rate;
1638 LCN Maint Esc_Rate := 1 + LCN Maint Esc_Rate;
1639 SPLICE SW Maint Esc_Rate := SPLICE SW Maint Esc_Rate + 1;
1640 Main Maint Esc_Rate := 1 + Main Maint Esc_Rate;
1641 Generate the complete output file name, with format 1/2/3 "un" output.
APPENDIX B: MAINTENANCE MANUAL

Page 32

SPLICE.PAS Program Listing:

```pascal
PRN_Name := 'SPLICE';
Maint_Months := 0;
Effective_Date := '09/01/85';
End;  (* Procedure INITIALIZE_RATES *)

PROCEDURE GET_RATE_INPUTS;

Begin (* Procedure GET_RATE_INPUTS *)
screen_fieldSS := 15;
varSS := 1;
retrieveSS := FALSE;
last_fieldSS := FALSE;
DISPLAY_SCREEN (Screenfile);  (* Display Screen *)
If the site selected is a MAP site, blank out the fields related to
INTERchanged (IHN) escalation and discount rates.
IF Stock_Point <> 'S' THEN
Begin
COLOR (1, 1);
QUITXY (70, 11);
WRITE ('');
QUITXY (70, 15);
WRITE ('');
End;
REPEAT (* until answerSS = 'Y' *)
{ Display items. Change retrieveSS to TRUE and INPUT item.)
REPEAT (* until actionSS = exitSS *)
CASE varSS of
  1: GETREAL(71, 8, '1', ENC_S1A_Purch_Price, Rate,
    '#.##', 0.00, 2.99, rateSS, retrieveSS,15, 31);
  2: GETREAL(71, 4, '1', ENC_S1A_Purch_Price,
    '#.##', 0.00, 2.99, rateSS, retrieveSS,15, 31);
  3: IF Stock_point = 'S' THEN
    GETREAL(71, 10, '1', ENC_S1A_Purch_Price,
      '#.##', 0.00, 2.99, rateSS, retrieveSS,15, 31);
  4: GETREAL(71, 11, '1', ENC_S1A_Purch_Price,
    '#.##', 0.00, 2.99, rateSS, retrieveSS,15, 31);
  5: GETREAL(71, 12, '1', ENC_S1A_Purch_Price,
    '#.##', 0.00, 2.99, rateSS, retrieveSS,15, 31);
  6: GETREAL(72, 11, '1', ENC_S1A_Purch_Price,
    '#.##', 0.00, 2.99, rateSS, retrieveSS,15, 31);
  7: IF Stock_Point = 'S' THEN
    GETREAL(71, 11, '1', ENC_S1A_Purch_Price,
      '#.##', 0.00, 2.99, rateSS, retrieveSS,15, 31);
  8: IF Stock_Point = 'S' THEN
    GETREAL(71, 15, '1', ENC_S1A_Purch_Price,
      '#.##', 0.00, 2.99, rateSS, retrieveSS,15, 31);
  9: GETREAL(70, 16, '1', ENC_S1A_Purch_Price,
    '#.##', 0.00, 2.99, rateSS, retrieveSS,15, 31);
```

203
APPENDIX B: MAINTENANCE MANUAL

Page 67

SPLICE.PAS Program Listing

1651 PRN_File_Name := CONCat (PRN_Name, '.PRN');
1652 Day := Copy (Effective_Date, 4, 2);
1653 Month_Index := Copy (Effective_Date, 1, 2);
1654 Val (Month_Index, Index, Errorcode);
1655 Month := Month_Name [Index];
1656 (Strip trailing blanks off the name of the month )
1657 Position := POS (' ', Month);
1658 IF Position > 0 THEN Month := Copy (Month, 1, Position - 1);
1659 Year := Copy (Effective_Date, 7, 2);
1660 Year CONCAT ('19', Year);
1661 END; /* Procedure GET_RATES */
1662
1663
1664 BEGIN { Procedure BUILD_COST_TABLE }
1665 ASSIGN (Textin, Filel);
1666 RESET (Textin);
1667 Count := 1;
1668 GET_RATES; (task user for all discount and escalation rates to be used )
1669 ClrScr;
1670 ClrOK (15, 1);
1671 ClrXY (16, 13);
1672 WRITE ('Constructing cost escalation and discount table.');
1673 READLN (Textin, Datain);
1674 WHILE Not EOF (Textin) DO
1675 BEGIN
1676 (* Build the Costtable array *)
1677 Costtable [Count].feature := Copy (Datain, 1, 6);
1678 Costtable [Count].clm := Copy (Datain, 1, 4);
1679 Costtable [Count].descr := Copy (Datain, 13, 27);
1680 Val (Copy (Datain, 40, 10), Costtable [Count].maint, Errorcode);
1681 Costtable [Count].basemaint := Costtable [Count].maint;
1682
1683 (* I.AN H/W Base Maintenance *)
1684 IF (Costtable [Count].feature = '320100') AND
1685 (Costtable [Count].feature = '420400') THEN
1686 Costtable [Count].basemaint := Costtable [Count].maint
1687 * I.AN_Maint_Base_Date
1688
1689 (* I.AN S/W Base Maintenance *)
1690 ELSE IF (Costtable [Count].feature = '550801') OR
1691 (Costtable [Count].feature = '550801') OR
1692 (Costtable [Count].feature = '551001') OR
1693 (Costtable [Count].feature = '551001') OR
1694 (Costtable [Count].feature = '551201') OR
1695 (Costtable [Count].feature = '551201') THEN
1696 Costtable [Count].basemaint := Costtable [Count].maint
1697 * I.AN_S/W_Base_Date
1698
1699 (* SPLICENet S/W Base Maintenance *)
1700 ELSE IF (Costtable [Count].feature = '550710') OR
1701
APPENDIX B: MAINTENANCE MANUAL

Page 35

SPLICE.PAS Program Listing

1701 (Costtable [Count].featureno = '550711') OR
1702 (Costtable [Count].featureno = '550803') OR
1703 (Costtable [Count].featureno = '550903') OR
1704 (Costtable [Count].featureno = '551003') OR
1705 (Costtable [Count].featureno = '551103') OR
1706 (Costtable [Count].featureno = '551203') OR
1707 (Costtable [Count].featureno = '551303') OR
1708 (Costtable [Count].featureno = '551403') OR
1709 (Costtable [Count].featureno = '551503') OR
1710 (Costtable [Count].featureno = '551504') OR
1711 (Costtable [Count].featureno = '551501') OR
1712 (Costtable [Count].featureno = '551502') OR
1713 (Costtable [Count].featureno = '551503') OR
1714 (Costtable [Count].featureno = '551504') THEN
1715 Costtable[Count].basemaint := Costtable [Count].momaint
1716 * SPLICENet_SW_Maint_Esc_Rate
1717
1718 ( Normal Maintenance Escalation )
1719 ELSE Costtable [Count].momaint := Costtable [Count].momaint
1720 * Maint_Esc_Rate;
1721
1722 ( 6100 H/W Purchase Escalation )
1723 Val (Copy (Datain, 50, 11), Costtable [Count].purchprice, Errorcode);
1724 IF (Costtable [Count].featureno (> '450300') AND
1725 (Costtable [Count].featureno (< '450400') THEN
1726 Costtable [Count].purchprice := Costtable [Count].purchprice
1727 IF (Costtable [Count].featureno (> '550701') AND
1728 (Costtable [Count].featureno (< '550710') THEN
1729 Costtable [Count].purchprice := Costtable [Count].purchprice
1730 (SPLICENet S/W Base Maintenance)
1731 ELSE IF (Costtable [Count].featureno = '550710') OR
1732 (Costtable [Count].featureno = '550711') OR
1733 (Costtable [Count].featureno = '550803') OR
1734 (Costtable [Count].featureno = '550903') OR
1735 (Costtable [Count].featureno = '551003') OR
1736 (Costtable [Count].featureno = '551103') OR
1737 (Costtable [Count].featureno = '551203') OR
1738 (Costtable [Count].featureno = '551303') OR
1739 (Costtable [Count].featureno = '551403') OR
1740 (Costtable [Count].featureno = '551503') OR
1741 (Costtable [Count].featureno = '551504') OR
1742 (Costtable [Count].featureno = '551501') OR
1743 (Costtable [Count].featureno = '551502') OR
1744 (Costtable [Count].featureno = '551503') OR
1745 (Costtable [Count].featureno = '551504') THEN
1746 Costtable [Count].basemaint := Costtable [Count].momaint
1747 * SPLICENet_SW_Purch_Esc_Rate

205
APPENDIX B: MAINTENANCE MANUAL  Page 69

SPLICE.PAS Program Listing

1751 | ( Training Escalation )
1752 |  ELSE IF (Costtable [Count].featureno = '39XXXX') or
1753 | (Costtable [Count].featureno = '3XXX') THEN
1754 |  Costtable [Count].purcprice := Costtable [Count].purcprice
1755 |  * Train_Esc_Rate
1756 |
1757 |  ( LCN H/W Purchase Escalation )
1758 |  ELSE IF (Costtable [Count].featureno = '320000') AND
1759 | (Costtable [Count].featureno < '420000') THEN
1760 |  Costtable [Count].purcprice := Costtable [Count].purcprice
1761 |  * LCN_Purch_Esc_Rate
1762 |
1763 |  ( FDC SNA Purchase Escalation )
1764 |  ELSE IF (Costtable [Count].featureno = '550700') THEN
1765 |  Costtable [Count].purcprice := Costtable [Count].purcprice
1766 |  * FDC_SNA_Purch_Esc_Rate
1767 |
1768 |  ( LCN S/W Purchase Escalation )
1769 |  ELSE IF (Costtable [Count].featureno = '550800') OR
1770 | (Costtable [Count].featureno = '550900') OR
1771 | (Costtable [Count].featureno = '551000') OR
1772 | (Costtable [Count].featureno = '551100') OR
1773 | (Costtable [Count].featureno = '552000') OR
1774 | (Costtable [Count].featureno = '551300') THEN
1775 |  Costtable [Count].purcprice := Costtable [Count].purcprice
1776 |  * LCN_SW_Esc_Rate
1777 |
1778 |  ( Documentation Purchase Escalation )
1779 |  ELSE IF (Costtable [Count].featureno = '710000') AND
1780 | (Costtable [Count].featureno < '749999') THEN
1781 |  Costtable [Count].purcprice := Costtable [Count].purcprice
1782 |  * Document_Esc_Rate
1783 |
1784 |  ( Site Preparation Installation Escalation )
1785 |  ELSE IF Costtable [Count].featureno = '100010' THEN
1786 |  Costtable [Count].purcprice := Siteinfo.sitelicts
1787 |  * Instal_Esc_Rate
1788 |
1789 |  ( Normal S/W Purchase Escalation )
1790 |  ELSE Costtable [Count].purcprice := Costtable [Count].purcprice
1791 |  * Purch_Esc_Rate
1792 |
1793 |  ( Installation Cost Escalation )
1794 |  VAL (Copy Datain, 62, 10), Costtable [Count]. instead, Errorcode;
1795 |  IF (Costtable [Count].featureno = '400100') AND
1796 | (Costtable [Count].featureno = '400100') THEN
1797 |  Costtable [Count].instcost := Costtable [Count].instead
1798 |  ELSE IF (Costtable [Count].featureno = '550300') AND
1799 | (Costtable [Count].featureno = '550800') THEN
206
APPENDIX B: MAINTENANCE MANUAL  Page 70

SPLICE.PAS Program Listing

1801 Costtable [Count].instcost := Costtable [Count].instcost
1802 ELSE Costtable [Count].instcost := Costtable [Count].instcost
1803 * Instal_Esc_Rate;
1804
1806 READLN (Textin, Datain);
1807 Count := Count + 1;
1808 End;
1809 CLOSE (Textin);
1810 End;  { Procedure BUILD_COST_TABLE }
1811
1813 PROCEDURE DELIVERY_ORDER_TITLE;
1814
1815 {**********************************************************************
1816 { This procedure generates the title page data and first headers to be
1817 { by the "Hardware" section. The data is written out to the diskfile
1818 { specified by the user when prompted for an output file Name.
1819 {**********************************************************************
1820 Begin { Procedure DELIVERY_ORDER_TITLE }
1821 ASSIGN (Diskfile, PRN_File_Name);
1822 REWRITE (Diskfile);
1823 WRITELN (Diskfile, "", "", "", "", "", "", "", "", "", "", "", "");
1824 WRITELN (Diskfile, "Naval Supply Systems Command SPLICE Delivery Order");
1825 WRITELN (Diskfile);
1826 WRITELN (Diskfile);
1828 WRITELN (Diskfile);
1829 WRITELN (Diskfile);
1830 WRITE (Diskfile, "Hardware" );
1831 HEADERS;
1832 End; { Procedure DELIVERY_ORDER_TITLE }
1833
1838 Begin { Procedure INITIALIZE }
1839 INIT TOTALS;
1840 OPENING SCREEN;
1841 PICK_A_SITE;
1842 BUILD_COST_TABLE;
1843 DELIVERY_ORDER_TITLE;
1844
1846 End; { Procedure INITIALIZE }
1847 { SPLICE.PAS }  { Name of work program on disk file }
1848 PROCEDURE CONFIGURE_COMPONENTS;
1849
1850 I Var
APPENDIX B: MAINTENANCE MANUAL

Page 71

Page 38

SPLICE.PAS--include file SPLICE2.PAS Program Listing

18511 { Input Variables Used For Documentation, Training & Maintenance }
18521 Computer_Ops, Hardware_Manual, Programmer_Ref : Integer;
18531 Sys_Programmer, Training_Group, Data_Communication : Integer;
18541 Hardware_Overview, Operator_Training, Sys_Resource : Integer;
18551 SPLICENet_Workshop, Sys_Tuning_Xray, CAL, Per_Call_Months : Integer;
18561
18571
18581 PROCEDURE CONFIGURE_HARDWARE;
18591
18601 Var
18611 Cable_Distance : String [1];
18621 Add_Expansion, Add_HYPERchannel, Add_Patchpanel, Add_System : Integer;
18631 AsyncCtrl, AsyncExtbd, AsyncPatch, A510, ByteSync, Bytesync : Integer;
18641 Cts, D128MB, D240MB, D540MB, ExpPanCab :
18651 HYPERCab, LPM1000, LPM600, PatchPanel, Printers, PdnPunch :
18661 SysCab, TxDrv, Trunks
18671
18681 PROCEDURE INITIALIZE_HARDWARE_INPUT;
18691
18701 Begin [ Procedure INITIALIZE_HARDWARE_INPUT ]
18711 { Initialize Variables To Default Values }
18721 Add_Expansion := 0;
18731 Add_HYPERchannel := 0;
18741 Add_Patchpanel := 0;
18751 Add_System := 0;
18761 AsyncCtrl := 0;
18771 AsyncExtbd := 0;
18781 AXX := 0;
18791 A140 := 0;
18801 A150 := 0;
18811 A200 := 0;
18821 A400 := 0;
18831 A510 := 0;
18841 BitSync := 0;
18851 Bytesync := 0;
18861 Cable_Distance := '0';
18871 CarRdr := 0;
18881 Crts := 0;
18891 D128MB := 0;
18901 D240MB := 0;
18911 D540MB := 0;
18921 HYPER Cab := 0;
18931 LPM1000 := 0;
18941 LPM600 := 0;
18951 Printers := 0;
18961 PdnPunch := 0;
18971 TapeDrv := 0;
APPENDIX B: MAINTENANCE MANUAL  Page 72
SPLICE.PAS-include file SPLICE2.PAS Program Listing

Page 39

SPLWCE.PAS

Program Listing

19011  TYPE2channels := 0;
19021  Trunks := 0;
19031  End;  { Procedure INITIALIZE_HARDWARE_INPUTS }
19041  PROCEDURE ODD_ERROR;
19061  PROCEDURE CLEAR_MESSAGE;
19061  PROCEDURE GET_HARDWARE_INPUTS;
19071  PROCEDURE GET_HARDWARE_INPUTS;
19071  PROCEDURE GET_HARDWARE_INPUTS;
19081  PROCEDURE CLEAR_MESSAGE;
19081  PROCEDURE CLEAR_MESSAGE;
19081  PROCEDURE CLEAR_MESSAGE;
19091  PROCEDURE CLEAR_MESSAGE;
19091  PROCEDURE CLEAR_MESSAGE;
19101  PROCEDURE CLEAR_MESSAGE;
19111  PROCEDURE CLEAR_MESSAGE;
19121  PROCEDURE CLEAR_MESSAGE;
19131  PROCEDURE CLEAR_MESSAGE;
19141  PROCEDURE CLEAR_MESSAGE;
19151  PROCEDURE CLEAR_MESSAGE;
19161  PROCEDURE CLEAR_MESSAGE;
19171  PROCEDURE CLEAR_MESSAGE;
19181  PROCEDURE CLEAR_MESSAGE;
19191  PROCEDURE CLEAR_MESSAGE;
19201  PROCEDURE CLEAR_MESSAGE;
19211  PROCEDURE CLEAR_MESSAGE;
19221  PROCEDURE CLEAR_MESSAGE;
19231  PROCEDURE CLEAR_MESSAGE;
19241  PROCEDURE CLEAR_MESSAGE;
19251  PROCEDURE CLEAR_MESSAGE;
19261  PROCEDURE CLEAR_MESSAGE;
19271  PROCEDURE CLEAR_MESSAGE;
19281  PROCEDURE CLEAR_MESSAGE;
19291  PROCEDURE CLEAR_MESSAGE;
19301  PROCEDURE CLEAR_MESSAGE;
19311  PROCEDURE CLEAR_MESSAGE;
19321  PROCEDURE CLEAR_MESSAGE;
19331  PROCEDURE CLEAR_MESSAGE;
19341  PROCEDURE CLEAR_MESSAGE;
19351  PROCEDURE CLEAR_MESSAGE;
19361  PROCEDURE CLEAR_MESSAGE;
19371  PROCEDURE CLEAR_MESSAGE;
19381  PROCEDURE CLEAR_MESSAGE;
19391  PROCEDURE CLEAR_MESSAGE;
19401  PROCEDURE CLEAR_MESSAGE;
19411  PROCEDURE CLEAR_MESSAGE;
19421  PROCEDURE CLEAR_MESSAGE;
19431  PROCEDURE CLEAR_MESSAGE;
19441  PROCEDURE CLEAR_MESSAGE;
19451  PROCEDURE CLEAR_MESSAGE;
19461  PROCEDURE CLEAR_MESSAGE;
19471  PROCEDURE CLEAR_MESSAGE;
19481  PROCEDURE CLEAR_MESSAGE;
19491  PROCEDURE CLEAR_MESSAGE;
19501  PROCEDURE CLEAR_MESSAGE;
APPENDIX B: MAINTENANCE MANUAL  Page 73

SPLICE.PAS—include file SPLICE2.PAS Program Listing

1951  IF ODD (DS40M8) THEN ODD_ERROR
1952  ELSE CLEAR MESSAGE;
1953  UNTIL not ODD (DS40M8);
1954  7: GETINT(40,10,3,'N',AsyncCtrl,'###',0,64,retSS,retrieveSS,14,1);
1955  8: GETINT(40,11,3,'N',AsyncExtCtrl,'###',0,2,retSS,retrieveSS,14,1);
1956  9: GETINT(40,12,3,'N',BitSync,'###',0,128,retSS,retrieveSS,14,1);
1957  10: GETINT(40,13,3,'N',ByteSync,'###',0,128,retSS,retrieveSS,14,1);
1958  11: GETINT(40,14,3,'N',TapeDrv,'###',0,128,retSS,retrieveSS,14,1);
1959  12: GETINT(40,15,3,'N',RdrPunch,'###',0,12,retSS,retrieveSS,14,1);
1960  13: GETINT(40,16,3,'N',CardRdr,'###',0,12,retSS,retrieveSS,14,1);
1961  14: GETINT(40,17,3,'N',LPM1000,'###',0,16,retSS,retrieveSS,14,1);
1962  15: GETINT(40,18,3,'N',LPM600,'###',0,16,retSS,retrieveSS,14,1);
1963  16: IF StockPoint = 'S' THEN
1964     GETINT(40,19,3,'N',Trunks,'###',0,2,retSS,retrieveSS,14,1);
1965  17: GETINT(40,20,3,'N',LIU,'###',0,256,retSS,retrieveSS,14,1);
1966  18: IF StockPoint = 'S' THEN
1967     GETINT(73,3,3,'N',A400,'###',0,256,retSS,retrieveSS,14,1);
1968  19: IF StockPoint = 'S' THEN
1969     GETINT(73,4,3,'N',A510,'###',0,256,retSS,retrieveSS,14,1);
1970  20: IF StockPoint = 'S' THEN
1971     GETINT (73,5,3,'N',A150,'###',0,256,retSS,retrieveSS,14,1);
1972  21: IF StockPoint = 'S' THEN
1973     GETINT (73,6,3,'N',A220,'###',0,256,retSS,retrieveSS,14,1);
1974  22: IF StockPoint = 'S' THEN
1975     GETINT(73,7,3,'N',A410,'###',0,256,retSS,retrieveSS,14,1);
1976  23: IF StockPoint = 'S' THEN
1977     GETINT(73,8,3,'N',A510,'###',0,256,retSS,retrieveSS,14,1);
1978  24: IF StockPoint = 'S' THEN
1979     GETINT(73,9,3,'N',THPChannels,'###',0,128,retSS,retrieveSS,14,1);
1980  25: IF StockPoint = 'S' THEN
1981     REPEAT
1982        GETITEM(75,20,1,'C',Cable_Distance, 'U', 'W', retSS,retrieveSS,14,1);
1983         IF (Cable_Distance < 'A') OR (Cable_Distance = 'F') THEN
1984            BEGIN
1985                COLOR (15, 4);
1986                OTXY (28, 25);
1987                WRITE ('G', ' Not within range A to F');
1988            END
1989         END
1990         ELSE CLEAR_MESSAGE;
1991     UNTIL (Cable_Distance = 'A') AND (Cable_Distance = 'F');
1992  26: IF varSS = Screen_fieldSS THEN last_fieldSS := True;
1993  27: IF STATUS = -1 Check code in [retSS]. Set "varSS" and "retSS";
1994  28: IF last_fieldSS AND (not retrieveSS) THEN
1995     BEGIN
2000
APPENDIX B: MAINTENANCE MANUAL

Page 74

SPLICE.PAS-include file SPLICE2.PAS Program Listing

20011 retrieveSS := True;
20021 last_fieldSS := False;
20031 actionSS := staySS;
20041 varSS := 1;
20051
20061 ELSE
20071 last_fieldSS := False;
20081 UNTIL actionSS = exitSS;
20091 ACCEPTINPUTS;
20101 UNTIL answerSS = 'Y';
20111 End;

20121 PROCEDURE ADDITIONAL_CABINETS;
20131 Begin

20141 screenfieldSS := 3;
20151 varSS := 1;
20161 retrieveSS := False;
20171 last_fieldSS := False;
20181
20191 DISPLAY_SCREEN (Screenfile); // Display Screen

20311 GETINT(40,3,'N',Processors,'##',0,256,retSS,False,14,1);
20321 GETINT(40,43,'N',Printers,'##',0,12,retSS,False,14,1);
20331 GETINT(40,5,'N',Crt,'##',0,999,retSS,False,14,1);
20341 GETINT(40,6,'N',DI28MB,'##',0,128,retSS,False,14,1);
2051  End;
2052  GETINT(68,13,2,'N',PatchPanel,'##',0,16,retSS,False,14,1);
2053  GETINT(68,14,2,'N',SysCab,'##',0,16,retSS,False,14,1);
2054  GETINT(68,15,2,'N',ExpanCab,'##',0,16,retSS,False,14,1);
2055  
2056  REPEAT { until answerSS = 'Y' }
2057  { Display Items. Change retrieveSS to True and INPUT items}
2058  REPEAT { until actionSS = exitSS }
2059  CASE varSS of
2060       1: GETINT(75,13,2, 'N', AddPatchPanel,
2061                        '##',0,8,retSS,retrieveSS,14,1);
2062       2: GETINT(75,14,2, 'N', AddSystem,
2063                        '##',0,8,retSS,retrieveSS,14,1);
2064       3: GETINT(75,15,2, 'N', AddExpansion,
2065                        '##',0,8,retSS,retrieveSS,14,1);
2066       End;  { CASE }
2067  
2068  IF varSS = screen_fieldSS THEN last_fieldSS := True;
2069  IF last_fieldSS AND (not retrieveSS) THEN
2070       Begin
2071           retrieveSS := True;
2072           last_fieldSS := False;
2073           actionSS := staySS;
2074       CASE varSS := 1;
2075       End;
2076  
2077  IF last_fieldSS := False;
2078  { PROCEDURE PRINT HW }
2079  PROCEDURE PRINT_HW;
2080  { This routine is used in the hardware generation process to set up the }
2081  { necessary parameters to be used by PRINT_HW when called. }
2082  { ********************************************************* }
2083  Begin  { PROCEDURE PRINT_HW }
2084     Maint_Factor := Maint_Prcp_RATE;
2085     Extended_Price := Quantity * CostTable[3],pricepers;
2086     ITEM_SUM;
2087     { Compute System Downtime Credit Component Factor per month }
2088     { ********************************************************* }
2089     System_Downtime_Component := System_Downtime_Component +
APPENDIX B: MAINTENANCE MANUAL

SPLICE.PAS--include file SPLICE2.PAS Program Listing

21011 (Quantity * CostTable[I].basemain
21021 * Maint_Factor);
21031 " Compute the Component Downtime Credit Factor per hour "
21041 {*******************************************************}
21051 Downtime_Credit := 
21061 (((CostTable[I].purchprice + CostTable[I].instcost) / 48) 
21071 + (CostTable[I].basemain * Maint_Factor) * 0.005;
21081 WRITELN (Diskfile, "", Line_Number:7, "", CostTable[I].featureno:8,
21091 "", CostTable[I].descript:28, "", Quantity:3,
21101 CostTable[I].purchprice:13:2, Extended_Price:12:2,
21111 CostTable[I].basemain:9:2, Maint_Factor:8:3, Maint_Months:5,
21121 Quantity * CostTable[I].basemain * Maint_Factor
21131 * Maint_Months:12:2, CostTable[I].instcost:8:2,
21141 CostTable[I].instcost * Quantity:9:2, Downtime_Credit:9:2,
21151 (Quantity * CostTable[I].basemain * Maint_Factor):9:2);
21161 End; { Procedure PRINT_IW }
21171
21181
21191 PROCEDURE CONFIGURE_PROCESSING_SUBSYSTEM;
21201
21211 Var
21221 OSP : Integer;
21231
21241 PROCEDURE COMPUTE_PROCESSORS;
21251 {*******************************************************}
21261 This procedure outputs a series of screens prompting the user to pro-
21271 vide the necessary inputs required to generate the processor related
21281 data for the desired configuration. Each input is checked to determine
21291 whether OR not the response is positive OR within the necessary limits.
21301 {*******************************************************}
21311 Begin ( Procedure COMPUTE_PROCESSORS )
21321 Quantity := Processors;
21331 I := 1 + 1; 1 1=2 Processors on delivery order
21341 IF Quantity > 0 THEN PRINT_IW;
21351 I := I * 1; 1 I=3 Uses # of Processors to determine
21361 IF Quantity > 0 THEN PRINT_IW;
21371 I := I + 1; 1 I=4 Floating Point Arithmetic,
21381 Only ordered by IBM sites.
21391 IF Quantity > 0 THEN PRINT_IW;
21401 ELSE I := I + 1;
21411 IF Quantity > 0 THEN PRINT_IW;
21421 I := I + 1; 1 I=5 OSP
21431 OSP := Processors;
2151 WHILE OSP MOD 16 > 0 DO
2152 OSP := OSP + 1;
2153 Quantity := OSP DIV 16;
2154 IF Quantity > 0 THEN PRINT_HW;
2155 END; { Procedure COMPUTE_PROCESSORS }
2156
2157 PROCEDURE COMPUTE_CRTS_PTRS;
2158 { This routine computes the number of Centronics Printers, CRTs and OSP -
2159 { interfaces required on the delivery order. }
2160 BEGIN { Procedure COMPUTE_CRTS_PTRS }
2161 Quantity := Printers;
2162 IF Quantity > 0 THEN PRINT_HW;
2163 Quantity := CRTs;
2164 IF Quantity > 0 THEN PRINT_HW;
2165 Quantity := OSP DIV 16;
2166 I := I + 1;
2167 IF Quantity > 0 THEN PRINT_HW;
2168 END; { Procedure COMPUTE_PROCESSORS }
2169
2170 PROCEDURE COMPUTE_CABINETS;
2171 { The following routine estimates the number of Patch Panel Cabinets and permits the user to increase this for reserve/expansion. }
2172 BEGIN { Procedure COMPUTE_CABINETS }
2173 Contigs, Slots, Temp : Integer;
2174 BEGIN { Procedure COMPUTE_CABINETS }
2175 Temp := Processors;
2176 IF Temp MOD 4 > 0 THEN Temp := Temp + 1;
2177 SysCab := Temp DIV 4;
2178 IF (Processors > 0) AND (SysCab = 0) THEN
2179 SysCab := 1;
2180 IF SysCab > 0 THEN PatchPanel := 1;
2181 ELSE IF SysCab = 0 THEN
2182}
The following routine estimates the number of Expansion Cabinets:

```
PatchPanel := 0;
Else IF SysCab > 1 THEN
  PatchPanel := SysCab - 1;
{ The following routine estimates the number of Expansion Cabinets: }
ExpanCab := 0;
Config16 := Processors DIV 16;
IF (Processors > (16 * Config16 + 4)) AND
  (Processors < (16 * (Config16 + 1) + 5)) THEN
  ExpanCab := Config16 + 1
ELSE
  IF (Processors > 4) AND (Processors < 21) THEN
    ExpanCab := 1;
  ELSE
    ADDITIONAL_CABINETS;
  Slots := SysCab * 24;
{ The following permits the user to increase the number of: }
{ Patch Panel Cabinets for reserve/expansion. }
Quantity := PatchPanel + Add_PatchPanel;
1 := 1 * 1;
{ 1-9 Patch Panel Cabinets: }
IF Quantity > 0 THEN PRINT 'N;
{ The following permits the user to increase the number of: }
{ System Cabinets for reserve/expansion. }
Quantity := SysCab + Add_System;
1 := 1 * 1;
{ 1-10 System Cabinets: }
IF Quantity > 0 THEN PRINT 'N;
{ The following permits the user to increase the number of: }
{ Expansion Cabinets for reserve/expansion. }
Quantity := ExpanCab + Add_Expansion;
1 := 1 * 1;
{ 1-12 Expansion Cabinets: }
IF Quantity > 0 THEN PRINT 'N;
END;
```

22011
22021
22031
22041
22051
22061
22071
22081
22091
22101
22111
22121
22131
22141
22151
22161
22171
22181
22191
22201
22211
22221
22231
22241
22251
22261
22271
22281
22291
22301
22311
22321
22331
22341
22351
22361
22371
22381
22391
22401
22411
22421
22431
22441
22451
22461
22471
22481
22491
22501
22511
22521
22531
22541
22551
22561
22571
22581
22591
22601
22611
22621
22631
22641
22651
22661
22671
22681
22691
22701
22711
22721
22731
22741
22751
22761
22771
22781
22791
22801
22811
22821
22831
22841
22851
22861
22871
22881
22891
22901
22911
22921
22931
22941
22951
22961
22971
22981
22991
23001
23011
23021
23031
23041
23051
23061
23071
23081
23091
23101
23111
23121
23131
23141
23151
23161
23171
23181
23191
23201
23211
23221
23231
23241
23251
23261
APPENDIX B: MAINTENANCE MANUAL  Page 79

SPLICE.FAS—include file SPLICE2.FAS Program listing

22511 22521 Var
22531  DiscCtrlr, DiscPatchPnl, THYPERPatchPnl : Integer;
22541
22551  ********************************************************************************************
22561 | The following procedures determine the number of discs, disc
22571 | controllers, disc patch panels, and Patch Panel Cabinets to be ordered
22581 | The reason that PATCHPnl must be called, which includes TNL and ASYNC
22591 | SYNC routines, from the disc procedure is to maintain the WAVE
22601 | required delivery order sequence. Discs are in even quantities due to
22611 | the "mirrored-disc" requirement in SPLICE.

22621  *******************************************************************************************
22631
22641  PROCEDURE COMPUTE_PATCH_PANELS;
22651
22661  Writeln Procedure COMPUTE_PATCH_PANELS;
22671  DiscCtrlr := (128+24) DIV 2;
22681  IF (DiscCtrlr MOD 2) = 0 THEN DiscCtrlr := DiscCtrlr + 1;
22691  quantity := DiscCtrlr;
22701  WHILE (quantity MOD 4) > 0 Do
22711   quantity := quantity + 1;
22721  DiscPatchPnl := quantity DIV 4;  // 4 disc controllers per Disc Patch Panel
22731  quantity := DiscPatchPnl;
22741  i := 1 + 1;  // i=13  Disc Patch Panels
22751  IF Quantity > 0 THEN PRINT_IW;
22761  i := i + 1;  // i=14  TNL/TYME HYPER Link Patch Panels
22771  IF Stock_Point = 'S' THEN  // i is the site Stock Point site
22781    IF THYPERChannels > 0 and THYPERChannels < 5) THEN
22791      Begin
22801        Quantity := 1;
22811        PRINT_IW;
22821      End
22831  ELSE
22841      Begin
22851        THYPERPatchPnl := THYPERChannels * 2;
22861        WHILE (THYPERPatchPnl MOD 4) > 0 Do
22871          THYPERPatchPnl := THYPERPatchPnl + 1;
22881        THYPERPatchPnl := THYPERPatchPnl DIV 4;
22891        quantity := THYPERPatchPnl DIV 4;
22901        IF quantity > 0 THEN PRINT_IW;
22911      End;
22921  i := i + 1;
22931  IF quantity > 0 THEN PRINT_IW;
22941  i := i + 1;
22951  IF ByteSync > 0 THEN
22961    Begin
22971      SITE SYNC lines require (SITE Patch Panels +
22981      10) Patch Panels
22991    End
23001  Quantity := ByteSync;
23011  PRINT_IW;

216
APPENDIX B: MAINTENANCE MANUAL

Page 47
SPLICE.PAS include file SPLICE2.PAS
Program Listing

23010  End;
23015  End;  ( Procedure COMPUTE_PATCH_PANeLS )
23031
23041
23051  PROCEDURE COMPUTE_DISK_COMPONENTS;
23061
23081  Begin  ( Procedure COMPUTE_DISK_COMPONENTS )
23091  Quantity := Disc Ctrlr;
23101  I := 1 + 1;  ( 1=17 Disc Controllers )
23111  IF Quantity > 0 THEN PRINT HW;
23121  Quantity := U128MB DIV 2;  ( Two drawers in each 128MB drive )
23131  I := 1 + 1;  ( 1=18 1st Drawer of 128MB Discs )
23141  IF Quantity > 0 THEN
23151    BEGIN
23161      PRINT HW;
23171      I := I + 1;  ( 1=19 2nd Drawer of 128MB Discs )
23181    END;
23191  PRINT HW;
23201  ELSE I := I + 1;
23211  Quantity := U240MB;
23221  I := 1 + 1;  ( 1=20 240MB Discs )
23231  IF Quantity > 0 THEN PRINT HW;
23241  Quantity := D540MB;
23251  I := 1 + 1;  ( 1=21 540MB Discs )
23261  IF Quantity > 0 THEN PRINT HW;
23271  END;  ( Procedure COMPUTE_DISK_COMPONENTS )
23281
23291
23301  BEGIN  ( Procedure COMPUTE_DISK )
23311   COMPUTE_PATCH_PANeLS;
23321   COMPUTE_DISK_COMPONENTS;
23331  END;  ( Procedure COMPUTE_DISK )
23341
23351
23361  PROCEDURE COMPUTE_TAPE;
23371
23381  **********************************************************************
23391  | This procedure determines the number of Tape Drives and Tape |
23401  | Controllers to be output on the delivery order. |
23411  **********************************************************************
23421
23431  BEGIN  ( Procedure COMPUTE_TAPE )
23441  Quantity := Tape Drv;
23451  IF Quantity > 0 THEN
23461    BEGIN
23471      I := I + 1;  ( 1=22 Tape Controllers )
23481      PRINT HW;
23491      I := I + 1;  ( 1=23 Tape Drives )
23501      PRINT HW;
23511  END;
APPENDIX B: MAINTENANCE MANUAL  Page 81

SPLICE.PAS-includes file SPLICE2.PAS Program Listing

2351 I  End
2352 ELSE I := I + 2;
2353 I End;  ( Procedure COMPUTE_TAPE )
2354 I
2355 I
2356 I Begin ( Procedure CONFIGURE_STORAGE_SUBSYSTEM )
2357 I COMPUTE_DISK;
2358 I COMPUTE_TAPE;
2359 I End;  ( Procedure CONFIGURE_STORAGE_SUBSYSTEM )
2360 I
2361 I
2362 I PROCEDURE CONFIGURE_INPUT_OUTPUT_SUBSYSTEM;
2363 I
2364 I
2365 I PROCEDURE COMPUTE_READER_PUNCHES;
2366 I **************************************************
2367 I This procedure determines the number of Reader/Punches and Card Readers 
2368 I to be output on the delivery order.
2369 I **************************************************
2370 I
2371 I Begin ( Procedure COMPUTE_READER_PUNCHES )
2372 I Quantity := RdrPunch;
2373 I := I + 1;
2374 I IF Quantity > 0 THEN PRINT_IW;
2375 I Quantity := CardRdr;
2376 I := I + 1;
2377 I IF Quantity > 0 THEN PRINT_IW;
2378 I End;  ( Procedure COMPUTE_READER_PUNCHES )
2379 I
2380 I
2381 I PROCEDURE COMPUTE_LINE_PRINTERS;
2382 I
2383 I **************************************************
2384 I This procedure determines the number of 1000 LPM and 600 LPM Printers 
2385 I to be output on the delivery order.
2386 I **************************************************
2387 I
2388 I Begin ( Procedure COMPUTE_LINE_PRINTERS )
2389 I Quantity := RdrPunch * CardRdr * LPM1000 * LPM600;
2390 I := I + 1;
2391 I IF Quantity > 0 THEN PRINT_IW;
2392 I Quantity := LPM1000;
2393 I := I + 1;
2394 I IF Quantity > 0 THEN PRINT_IW;
2395 I Quantity := LPM600;
2396 I := I + 1;
2397 I IF Quantity > 0 THEN PRINT_IW;
2398 I End;  ( Procedure COMPUTE_LINE_PRINTERS )
2399 I
2400 I

218
APPENDIX B: MAINTENANCE MANUAL

Page 82

Page 49

SPIICE.PAS-include file SPLICE2.PAS Program List:

```
2401 I Begin { Procedure CONFIGURE_INPUT_OUTPUT_SUBSYSTEM }
2402 I COMPILE READER PUNCHES;
2403 I COMPUTE LINE PRINTERS;
2404 I End; { Procedure CONFIGURE_INPUT_OUTPUT_SUBSYSTEM }
2405 I
2406 I
2407 I PROCEDURE CONFIGURE_COMMUNICATIONS_SUBSYSTEM;
2408 I
2409 I
2410 I PROCEDURE COMPUTE_FOX;
2411 I (*****************************************************************)
2412 I { This procedure determines the number of FOX fibre optic controllers and }
2413 I { lines to be output on the delivery order. FOX permits SPLICE nodes of }
2414 I { 16 or less Processors (which are co-located within 1000 meters) to be }
2415 I { directly interconnected. }
2416 I (*****************************************************************)
2417 I
2418 I Begin { Procedure COMPUTE_FOX }
2419 I I := I + 1; { I=29 Skips Interprocessor Bus }
2420 I IF Processors > 16 THEN
2421 I Begin
2422 I I := I + 1; { I=30 FOX CNTRLS for >16 unit system }
2423 I Quantity := Processors; { Processors > 16? If so, order FOX }
2424 I WHILE Quantity MOD 16 > 0 DO
2425 I Quantity := Quantity * I;
2426 I Quantity := Quantity DIV 16;
2427 I PRINT_HW;
2428 I I := I + 1; { I=31 FOX cables }
2429 I Quantity := Quantity - 1;
2430 I PRINT_HW;
2431 I End
2432 I ELSE I := I + 2;
2433 I End; { Procedure COMPUTE_FOX }
2434 I
2435 I
2436 I PROCEDURE COMPUTE_HYPERCHANNELS;
2437 I (*****************************************************************)
2438 I { This procedure is called by COMPUTE_COMMUNICATIONS_SUBSYSTEM and }
2439 I { is invoked only for the configuration of Stock Point Sites. It }
2440 I { uses the user inputs for HYPERchannel adapters and connections to }
2441 I { write out the correct HYPERchannel component site quantities to }
2442 I { the delivery order. Selected components are written to disk via }
2443 I { the PRINT_HW routine. }
2444 I (*****************************************************************)
2445 I
2446 I
2447 I PROCEDURE EXTRA_HYPERCABINETS;
2448 I
2449 I
2450 I End; { Procedure EXTRA_HYPERCABINETS }
```

---

219
Add_CHANNEL := 0;
COLOR (15, 1);
GOTOXY (51, 16);
WRITE ('HYPERchannel');
screen FieldSS := 1;
varSS := 1;
retrieveSS := False;
last_fieldSS := False;
retSS := '';

IF Stock_Point = 'S' THEN
GETINT(58, 16, 2, 'N', HYPERchannel, '#', 0, 16, retSS, False, 14, 1);
REPEAT ( until answerSS = 'Y' )
{ Display Items, Change retrieveSS to True and INPUT items }
REPEAT ( until actionSS = exitSS )
IF Stock_Point = 'S' THEN
GETINT(75, 16, 2, 'N', Add_CHANNEL, '#', 0, 8, retSS, retrieveSS, 14, 1);

last_fieldSS := True;
RET_STATUS; { Check the code in "retSS". Set "varSS" and "actionSS" }

{ Check to see whether to switch retrieveSS to true }
IF last_fieldSS AND (not retrieveSS) THEN
Begin
    retrieveSS := True;
    last_fieldSS := False;
    actionSS := staySS;
    varSS := 1;
End
ELSE
last_fieldSS := False;
UNTIL actionSS = exitSS;
ACCEPT_INPUTS;
UNTIL answerSS = 'Y';
End; { Procedure EXTR CHANNELS }

Begin; Procedure COMPUTECHANNELS }
A400 := A400 + CHANNELS;
Stores all minicomputer CHANNEL Adapter requirements :
Quantity := A400;
1 := 1 + 1;
IF Quantity = 0 THEN PRINT INT;
1 := 1 + 1;
IF Trunks = 'J' THEN
2 := 1;
Quantity := 1;
PRINT INT;
25011 End;
25021 HYPERCab := (A400 DIV 2) + A150 + AXXX + A220 + A140 + AS10) DIV 2;
25031 EXTRA_HYPERCABINETS;
25041 Quantity := HYPERCab + Add_HYPERChannel;
25051 
25061 [**************************************************************************]
25071 { The above line determines the number of HYPERchannel cabinets to }
25081 { be estimated for the user. It assumes that all TANDEM and P-E }
25091 { HYPERchannels can reside in the same cabinet and that one cabinet }
25101 { for every two additional adapters will suffice. }
25111 [**************************************************************************]
25121 
25131 I := I + 1; { 1=34 HYPERchannel Cabinets }
25141 IF Quantity > 0 THEN PRINT_HW;
25151 Quantity := HYPERchannels;
25161 I := I + 1; { 1=35 THL controllers }
25171 IF Quantity > 0 THEN PRINT_HW;
25181 Quantity := Trunks; { 1=36 - 41 LCN Trunk Line }
25191 IF Trunks > 0 THEN
25201 BEGIN
25211 Case Cable Distance of
25221 'A': BEGIN
25231 I := I + 1;
25241 PRINT_HW;
25251 I := I + 5; { 1=36 ~ 500 ft }
25261 END;
25271 'B': BEGIN
25281 I := I + 2; { 1=37 - 1000 ft }
25291 PRINT_HW;
25301 I := I + 4; { 1=38 - }
25311 END;
25321 'C': BEGIN
25331 I := I + 1;
25341 PRINT_HW;
25351 I := I + 3; { 1=39 - 1500 ft }
25361 END;
25371 'D': BEGIN
25381 I := I + 4; { 1=39 - 2500 ft }
25391 PRINT_HW;
25401 I := I + 2; { 1=40 - }
25411 END;
25421 'E': BEGIN
25431 I := I + 5; { 1=40 - 4000 ft }
25441 PRINT_HW;
25451 I := I + 1; { 1=41 - }
25461 END;
25471 'F': BEGIN
25481 I := I + 6; { 1=41 - 5400 ft }
25491 PRINT_HW;
25501 END;
APPENDIX B: MAINTENANCE MANUAL

Page 52

SPLICE.PAS include file SPLICE2.PAS Program Listing

2551 End;
2552 End
2553 ELSE I := I + 6;
2554 Quantity := A150;
2555 I := I + 1;
2556 IF Quantity > 0 THEN PRINT_LINE;
2557 END2 := AXXX;
2558 I := I + 1;
2559 IF Quantity > 0 THEN PRINT_LINE;
2560 Q := A150 + AXXX + A220;
2561 I := I + 1;
2562 END3 := A150 + AXXX + A220;
2563 IF Quantity > 0 THEN PRINT_LINE;
2564 Quantity := A460 - HYPERchannels;
2565 I := I + 1;
2566 IF Quantity > 0 THEN PRINT_LINE;
2567 Quantity := A220;
2568 I := I + 1;
2569 IF Quantity > 0 THEN PRINT_LINE;
2570 Q := A150 + AXXX + A220;
2571 I := I + 1;
2572 IF Quantity > 0 THEN PRINT_LINE;
2573 Quantity := A510;
2574 I := I + 1;
2575 IF Quantity > 0 THEN PRINT_LINE;
2576 I := I + 1;
2577 End; ( Barrois & IBM hosts require ASCII to EBCDIC Conversion Board )
2578 PROCEDURE COMPUTE_HYPERCHANNELS;
2579 PROCEDURE COMPUTE_TERMINAL_COMMUNICATIONS_COMPONENTS;
2580 Var
2581 Communication, K : Integer;
2582 Begin
2583 ************ This procedure is used to handle all SPLIT terminal oriented communications requirements. PRINT_LINE is called to write output components to the output file. ************
2584 Begin ( Procedure COMPUTE TERMINAL COMMUNICATIONS COMPONENTS )
2585 AsyncExcl := AsyncYri * AsyncExcl;
2586 IF AsyncYri = 0 THEN
2587 IF AsyncExcl = 0 THEN
2588ifs
2589 */
2590 //
APPENDIX B: MAINTENANCE MANUAL  Page 86
SPLICE.PAS include file SPLICE2.PAS Program Listing

Begin
Quantity := AsyncExtbd;
I := I + 1; ( I=51 ASYNC Extension Boards )
PRINT_IW;
End
Else I := I + 1;

End

Else I := I + 2;
K := (LIU -1) DIV 45;
Quantity := LIU;
IF LIU > 0 THEN
Begin
Quantity := K + I;
I := I + 1;
PRINT_IW;
IF (LIU > 45*K) AND (LIU <= 45*K+15) THEN
Begin
Quantity := 2*K;
CableOpt := 6*K+2;
End;
IF (LIU > 45*K+15) AND (LIU <= 45*K+30) THEN
Begin
Quantity := 2*K+1;
CableOpt := 6*K+4;
End;
IF (LIU > 45*K+30) AND (LIU <= 45*(I+1)) THEN
Begin
Quantity := 2*(K+1);
CableOpt := 6*K+6;
End;
Else I := I + 2;
PRINT_IW;
Quantity := LIU;
I := I + 1;
PRINT_IW;
Quantity := CableOpt;
6100 cable; 2 / base, & 64 add on
1
I := I + 1;
IF Quantity > 0 THEN PRINT_IW;
Else I := I + 2;
Print 54-98 if no 6400 Controllers
If LIU = 0 PRINT_IW;
PRINT_IW;
ELSE PRINT_IW;
End
223
**APPENDIX B: MAINTENANCE MANUAL**

SPLICE.PAS—include file SPLICE2.PAS Program listing

---

### Variables Section For C:SOFTWARE

```pascal
26831 | Variables Section For C:SOFTWARE |
26841 | SW6100 : Char;
26851 | ADCCP_6100, ATP_6100, BSC_6100, SNAX_6100, TISIT_6100, NM_6529 : Str[11];
26861 | LON, FDC DLANet, FDC SNA, NMF Performance : Str[11];
26871 | NMF Accounting, NMF Base Facility, NMF Diagnostics, NMF Group : Str[11];
26881 | FOLL_SELECT, FILE_SECURITY, LCN_FUP, T_TEXT, TR_3271 : Str[11];
26891 | NETEX_Months, SPLICE2Net_Months : Integer;
26901 |
```

---

### Procedure CONFURESOFTWARE

```pascal
26921 | This procedure is used to determine the software requirements for |
26931 | the delivery order. Please see the rules in the Programmer Mau |
26941 | tenance Manual to determine which packages are ordered PER |
26951 | the software packages. See the BUILD TABLE procedure for |
26961 | specific factors and how they are incorporated into the TABLE |
26971 | array. |
```

---

### Example Code

```pascal
26511 | Begin |
26521 | Quantity := Bytensync;
26531 | PRINT_HW;
26541 | End;
26551 | | Skips 1=61-62; ARCLI items ordered |
26561 | End; | Procedure: COMPUTE TERMINAL_COMMUNICATIONS_COMPONENTS |
26571 |
26581 |
26591 | Begin | Procedure CONFURE_TERMINAL_COMMUNICATIONS_COMPONENTS |
26601 | COMPUTE_FOX; |
26611 | IF Stock_Point = 'S' THEN COMPUTE_HYPERCHANNELS |
26621 | Else i := i + 18; |
26631 | COMPUTE_TERMINAL_COMMUNICATIONS_COMPONENTS; |
26641 | End; | Procedure: CONFURE_TERMINAL_COMMUNICATIONS_COMPONENTS |
26651 |
26661 |
26671 | Begin | Procedure CONFURE_HARDWARE |
26681 | INITIALIZE_HARDWARE_INPUTS; |
26691 | GET_HARDWARE_INPUTS; |
26701 | CONFURE_PROCESSING_SUBSYSTEM; |
26711 | CONFURE_STORAGE_SUBSYSTEM; |
26721 | CONFURE_INPUT_OUTPUT_SUBSYSTEM; |
26731 | CONFURE_COMMUNICATIONS_SUBSYSTEM; |
26741 | COMPUTE_SECTION_TOTALS ('Software'); |
26751 | Mode := Soft; |
26761 | End; | Procedure: CONFURE_HARDWARE |
26771 |
26781 |
26791 | PROCEDURE CONFURE_SOFTWARE; |
26801 |
26811 |
26821 | Var |
26831 | Variables Section For C:SOFTWARE |
26841 | SW6100 : Char;
26851 | ADCCP_6100, ATP_6100, BSC_6100, SNAX_6100, TISIT_6100, NM_6529 : Str[11];
26861 | LON, FDC DLANet, FDC SNA, NMF Performance : Str[11];
26871 | NMF Accounting, NMF Base Facility, NMF Diagnostics, NMF Group : Str[11];
26881 | FOLL_SELECT, FILE_SECURITY, LCN_FUP, T_TEXT, TR_3271 : Str[11];
26891 | NETEX_Months, SPLICE2Net_Months : Integer;
26901 |
```

---

### Procedure CONFURE_HARDWARE

```pascal
26601 | COMPUTE_FOX; |
26611 | IF Stock_Point = 'S' THEN COMPUTE_HYPERCHANNELS |
26621 | Else i := i + 18; |
26631 | COMPUTE_TERMINAL_COMMUNICATIONS_COMPONENTS; |
26641 | End; | Procedure: CONFURE_HARDWARE |
26651 |
26661 |
26671 | Begin | Procedure CONFURE_HARDWARE |
26681 | INITIALIZE_HARDWARE_INPUTS; |
26691 | GET_HARDWARE_INPUTS; |
26701 | CONFURE_PROCESSING_SUBSYSTEM; |
26711 | CONFURE_STORAGE_SUBSYSTEM; |
26721 | CONFURE_INPUT_OUTPUT_SUBSYSTEM; |
26731 | CONFURE_COMMUNICATIONS_SUBSYSTEM; |
26741 | COMPUTE_SECTION_TOTALS ('Software'); |
26751 | Mode := Soft; |
26761 | End; | Procedure: CONFURE_HARDWARE |
26771 |
26781 |
26791 | PROCEDURE CONFURE_SOFTWARE; |
26801 |
26811 |
26821 | Var |
26831 | Variables Section For C:SOFTWARE |
26841 | SW6100 : Char;
26851 | ADCCP_6100, ATP_6100, BSC_6100, SNAX_6100, TISIT_6100, NM_6529 : Str[11];
26861 | LON, FDC DLANet, FDC SNA, NMF Performance : Str[11];
26871 | NMF Accounting, NMF Base Facility, NMF Diagnostics, NMF Group : Str[11];
26881 | FOLL_SELECT, FILE_SECURITY, LCN_FUP, T_TEXT, TR_3271 : Str[11];
26891 | NETEX_Months, SPLICE2Net_Months : Integer;
26901 |
```

---

### Example Code

```pascal
26511 | Begin |
26521 | Quantity := Bytensync;
26531 | PRINT_HW;
26541 | End;
26551 | | Skips 1=61-62; ARCLI items ordered |
26561 | End; | Procedure: COMPUTE TERMINAL_COMMUNICATIONS_COMPONENTS |
26571 |
26581 |
26591 | Begin | Procedure CONFURE_TERMINAL_COMMUNICATIONS_COMPONENTS |
26601 | COMPUTE_FOX; |
26611 | IF Stock_Point = 'S' THEN COMPUTE_HYPERCHANNELS |
26621 | Else i := i + 18; |
26631 | COMPUTE_TERMINAL_COMMUNICATIONS_COMPONENTS; |
26641 | End; | Procedure: CONFURE_TERMINAL_COMMUNICATIONS_COMPONENTS |
26651 |
26661 |
26671 | Begin | Procedure CONFURE_HARDWARE |
26681 | INITIALIZE_HARDWARE_INPUTS; |
26691 | GET_HARDWARE_INPUTS; |
26701 | CONFURE_PROCESSING_SUBSYSTEM; |
26711 | CONFURE_STORAGE_SUBSYSTEM; |
26721 | CONFURE_INPUT_OUTPUT_SUBSYSTEM; |
26731 | CONFURE_COMMUNICATIONS_SUBSYSTEM; |
26741 | COMPUTE_SECTION_TOTALS ('Software'); |
26751 | Mode := Soft; |
26761 | End; | Procedure: CONFURE_HARDWARE |
26771 |
26781 |
26791 | PROCEDURE CONFURE_SOFTWARE; |
26801 |
26811 |
26821 | Var |
26831 | Variables Section For C:SOFTWARE |
26841 | SW6100 : Char;
26851 | ADCCP_6100, ATP_6100, BSC_6100, SNAX_6100, TISIT_6100, NM_6529 : Str[11];
26861 | LON, FDC DLANet, FDC SNA, NMF Performance : Str[11];
26871 | NMF Accounting, NMF Base Facility, NMF Diagnostics, NMF Group : Str[11];
26881 | FOLL_SELECT, FILE_SECURITY, LCN_FUP, T_TEXT, TR_3271 : Str[11];
26891 | NETEX_Months, SPLICE2Net_Months : Integer;
26901 |
```
27011 {****************************************************************************************************}
27021
27031 PROCEDURE INITIALIZE_SOFTWARE_INPUTS;
27041
27051 Begin  (Procedure INITIALIZE_SOFTWARE_INPUTS)
27061  (Initialize Variables To Default Values)
27071 AIDCCP_6100 := 'N';
27081 AM_6520 := 'Y';
27091 ATP_6100 := 'Y';
27101 BSC_6100 := 'N';
27111 DDN := 'N';
27121 DCCDLANet := 'N';
27131 DCCSNA := 'Y';
27141 FILE_SECURITY := 'N';
27151 FILE_SECURITY := 'N';
27161 FILE_SECURITY := 'N';
27171 LCN_FUP := 'N';
27181 NETEX_Months := 0;
27191 NMF_AmOut := 'N';
27201 NMF_BFACility := 'N';
27211 NMF_Diagnostics := 'N';
27221 NMF_Group := 'N';
27231 NMF_Performance := 'N';
27241 PULL_SELECT := 'Y';
27251 SHAX_6100 := 'Y';
27261 SPLICENet_Months := 0;
27271 T_TEXT := 'Y';
27281 TIMET_6100 := 'N';
27291 TR_3271 := 'N';
27301 End;  (Procedure INITIALIZE_SOFTWARE_INPUTS)
27311
27321
27331 PROCEDURE GET_SOFTWARE_INPUTS;
27341
27351 Begin
27361 screen_fieldSS := 21;
27371 varSS := 1;
27381 retrieveSS := False;
27391 last_fieldSS := False;
27401 DISPLAY_SCREEN (Screenfile);  I Display Screen I
27411
27421 REPEAT (until answerSS = 'Y')
27431 I Display Items. Change retrieveSS to True and INPUT item(s)
27441 REPEAT (until actionSS = 'exitSS')
27451 ONE varSS of
27461 1: GETITEM(6,5,1,'Y',FILE_SECURITY,'Y',',',' ',retSS,retrieveSS,10,11);
27471 2: IF Stock Point = 'Y' THEN
27481 2: GETITEM(6,6,1,'Y',AM_FUP,'Y',',',' ',retSS,retrieveSS,15,11);
27491 3: GETITEM(6,8,1,'Y',ATP_6100,'Y',',',' ',retSS,retrieveSS,15,11);
27501 4: GETITEM(6,9,1,'Y',BSC_6100,'Y',',',' ',retSS,retrieveSS,15,11);
APPENDIX B: MAINTENANCE MANUAL

SPLICE.PAS—include file SPLICE2.PAS Program Listing

```pascal
2751I 5: GETITEM(36,10,1,'Y', ADCP_6100, 'U', '', '', retSS, retrieveSS,15,1);
2752I 6: GETITEM(36,11,1,'Y', POLL_SELECT, 'U', '', '', retSS, retrieveSS,15,1);
2753I 7: GETITEM(36,12,1,'Y', SNAX_6100, 'U', '', '', retSS, retrieveSS,15,1);
2754I 8: GETITEM(36,13,1,'Y', TIME_6100, 'U', '', '', retSS, retrieveSS,15,1);
2755I 9: GETITEM(36,14,1,'Y', IR_3271, 'U', '', '', retSS, retrieveSS,15,1);
2756I 10: GETITEM(36,15,1,'Y', AM_6520, 'U', '', '', retSS, retrieveSS,15,1);
2757I 11: GETITEM(36,16,1,'Y', T_TEXT, 'U', '', '', retSS, retrieveSS,15,1);
2758I 12: GETITEM(36,17,1,'Y', SRC_6100, 'U', '', '', retSS, retrieveSS,15,1);
2759I 13: GETITEM(36,18,1,'Y', SRC_SNAX, 'U', '', '', retSS, retrieveSS,15,1);
2760I 14: GETITEM(36,19,1,'Y', SRC_2771, 'U', '', '', retSS, retrieveSS,15,1);
2761I 15: GETITEM(36,20,1,'Y', SRC_TIM, 'U', '', '', retSS, retrieveSS,15,1);
2762I 16: IF NMF_Group = 'N' THEN
2763I   GETITEM(74,12,1,'Y', NMF_Base_Facility, 'U', '', '', retSS, retrieveSS,15,1);
2764I 17: IF NMF_Group = 'N' THEN
2765I   GETITEM(74,13,1,'Y', NMF_Performance, 'U', '', '', retSS, retrieveSS,15,1);
2766I 18: IF NMF_Group = 'N' THEN
2767I   GETITEM(74,14,1,'Y', NMF_Diagnostics, 'U', '', '', retSS, retrieveSS,15,1);
2768I 19: IF NMF_Group = 'N' THEN
2769I   GETITEM(74,15,1,'Y', NMF_Accounting, 'U', '', '', retSS, retrieveSS,15,1);
2770I 20: IF Stock_Point = 'S' THEN
2771I   GETITEM(60,21,2,'N', NETEX_Months, '#', 0,12, retSS, retrieveSS,15,1);
2772I 21: GETITEM(60,22,2,'N', SPLICE_Months, '#', 0,12, retSS, retrieveSS,15,1);
2773I End; ( CASE )
2774I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2775I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2777I IF last_fieldSS AND (not retrieveSS) THEN
2778I BEGIN
2779I   retrieveSS := True;
2780I   last_fieldSS := false;
2781I   actionSS := staySS;
2782I   varSS := 1;
2783I END;
2784I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2785I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2786I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2787I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2788I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2789I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2790I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2791I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2792I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2793I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2794I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2795I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2796I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2797I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2798I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2799I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
2800I IF varSS = screen_fieldSS THEN last_fieldSS:=true;
```

2801 **END**
28011 ( used in any maintenance computations. )
28021 {*******************************************************************************}
28031 28041 Begin ( Procedure PRINT_SW )
28051 CASE Type_Software of
28061 : Begin ( Per Processor Basis )
28071 Maint_Factor := Maint_Esc_Rate;
28081 Extended_Price := Quantity * CostTable[I].purchprice;
28091 End;
28101 2: Begin ( Per Site Basis )
28111 Maint_Factor := Maint_Esc_Rate;
28121 Extended_Price := CostTable[I].purchprice;
28131 End;
28141 3: Begin ( Per Processor Basis )
28151 Maint_Factor := 1;
28161 Extended_Price := Quantity * CostTable[I].purchprice;
28171 End;
28181 End; ( End of CASE Statement )
28191 LINE_SETUP;
28201 {*******************************************************************************}
28211 Compute System Downtime Component Factor per month 
28221 {*******************************************************************************}
28231 System_Downtime_Component := System_Downtime_Component +
28241 (Quantity * CostTable[I].basemaint +
28251 Maint_Factor);
28261 {*******************************************************************************}
28271 Compute the Component Downtime Credit Factor per hour
28281 {*******************************************************************************}
28291 Downtime_Credit := (((CostTable[I].purchprice + CostTable[I].instcost) / 40) +
28301 (CostTable[I].basemaint * Maint_Factor)) * 0.995;
28311 WRITEIN (Diskfile, "", Line_Number:"", "", CostTable[I].feature,"",
28321 "", CostTable[I].descript:"", "", Quantity:"",
28331 CostTable[I].purchprice:13:2, Extended_Price:12:2,
28341 CostTable[I].basemaint:9:2, Maint_Factor:8:1, Maint_Months:5,
28351 CostTable[I].basemaint * Maint_Factor * Maint_Months:12:2,
28361 CostTable[I].instcost:8:2,
28371 CostTable[I].instcost * Quantity:9:2, Downtime_Credit:9:2,
28381 (Quantity * CostTable[I].basemaint * Maint_Factor:9:2));
28391 End; ( Procedure PRINT_SW )
28401
28411 PROCEDURE COMPUTE_PROCESSOR_SOFTWARE;
28421
28431 Begin ( Procedure COMPUTE_PROCESSOR_SOFTWARE )
28441 IF Quantity = 0 THEN
28451 {*******************************************************************************}
28461 PRINT_SW (1);
28471 IF Quantity = 0 THEN
28481 PRINT_SW (1);
28491 PRINT_SW (1);
28501 PRINT_SW (1);
APPENDIX B: MAINTENANCE MANUAL  Page 91

Page 58 SPLICE.PAS-include file SPLICE2.PAS Program Listing

28511  PRINTSW (2);  | PER-SITE SOFTWARE |
28521   I := I + 1;   | I=65  System Utilities |
28531  PRINTSW (2);  | PER-SITE SOFTWARE |
28541   I := I + 1;   | I=66  ENCOMPASS |
28551  PRINTSW (1);  | PER-PROCESSOR SOFTWARE |
28561   I := I + 5;   | Skips 5 p/o software packages |
28571   I := I + 1;   | I=72  TPS Software |
28581  PRINTSW (2);  | PER-SITE SOFTWARE |
28591   I := I + 5;   | Skips 5 p/o software packages |
28601  
28611  ELSE I := I + 15;  | I=78  File Security Software |
28621  I := I + 1;   | I=79  Card Reader Software |
28631 IF File_Security = 'Y' THEN PRINTSW (2); |
28641   I := I + 1;   | PER-SITE SOFTWARE |
28651 IF CardRdr > 0 THEN PRINTSW (2); |
28661   I := I + 1;   | Skips 3 p/o software packages |
28671  End;  | Procedure COMPUTE_PROCESSOR_SOFTWARE |
28681  
28691  PROCEDURE COMPUTE_PROCESSORSOFTWARE; |
28701  
28711  PROCEDURE COMPUTE_COMMUNICATIONSOFTWARE; |
28721  
28731 Var |
28741  Temp_Months : Integer; |
28751  
28761  PROCEDURE COMPUTE_TANDEM_SOFTWARE; |
28771  
28781  BEGIN  | Procedure COMPUTE_TANDEM_SOFTWARE |
28791  
28801  Quantity := Processors; |
28811  IF Quantity > 0 THEN |
28821    BEGIN |
28831      I := I + 1;   | I=83  EXPAND Software |
28841  PRINTSW (1);   | PER-PROCESSOR SOFTWARE |
28851   I := I + 1;   | I=84  Skips Exchange RIE Software |
28861 IF Possibly need to add choices to software screen for next two rows THEN |
28871      I := I + 1;   | I=85  AM 1270 Software |
28881  PRINTSW (1);   | PER-PROCESSOR SOFTWARE |
28891   I := I + 1;   | I=86  X.25 ACCESS Software |
28901  PRINTSW (1);   | PER-PROCESSOR SOFTWARE |
28911  
28921  ELSE I := I + 4; |
28931   I := I + 1;   | Skips I=87 HYPERLINK Access Method S/W |
28941   I := I + 1;   | I=88  S/T F/F |
28951 IF (SiteInfo.Site[1]* = 'S') AND (SiteInfo.F/F = 'Y') THEN |
28961  PRINTSW (1);   |
28971   I := I + 1;   | I=89  Skip SITE Terminal Support |
28981   I := I + 1;   | I=90  AIP 6100 |
28991 IF AIP 6100 = 'Y' THEN PRINTSW (1); |
29001   I := I + 1;   | I=91  AIP 6100 |

228
APPENDIX B: MAINTENANCE MANUAL Page 92

Page 59 SPLICE.PAS-include file SPLICE2.PAS Program Listing

2901 IF BSC_6100 = 'Y' THEN PRINT_SW (1);
2902 I := I + 1; \{ I=92 ADCCP 6100 \}
2903 IF ADDCP_6100 = 'Y' THEN PRINT_SW (1);
2904 I := I + 1; \{ I=93 DINET 6100 \}
2905 IF DINET_6100 = 'Y' THEN PRINT_SW (1);
2906 I := I + 1; \{ I=94 BURROUGHS POLL-SELECT \}
2907 IF POLL_SELECT = 'Y' THEN PRINT_SW (1);
2908 I := I + 1; \{ I=95 SNAX 6100 \}
2909 IF SNAX_6100 = 'Y' THEN PRINT_SW (1);
2910 I := I + 1; \{ I=96 TR 3271 \}
2911 IF TR_3271 = 'Y' THEN PRINT_SW (1);
2912 I := I + 1; \{ I=97 AM 6520 \}
2913 IF AM_6520 = 'Y' THEN PRINT_SW (1);
2914 I := I + 1; \{ I=98 FDC SNA Interface Package \}
2915 IF FDC_SNA = 'Y' THEN PRINT_SW (2); \{ PER-SITE SOFTWARE \}
2916 I := I + 1; \{ I=99 FDC MRT Interface Package \}
2917 IF FDC_MRT = 'Y' THEN PRINT_SW (2); \{ PER-SITE SOFTWARE \}
2918 End; \{ Procedure COMPUTE_TANDEM_SOFTWARE \}
2919
2920 PROCEDURE COMPUTE_HYPERCHANNEL_SOFTWARE;
2921
2922 Begin \{ Procedure COMPUTE_HYPERCHANNEL_SOFTWARE \}
2923 I := I + 1;
2924 Temp_Months := Maint_Months;
2925 Maint_Months := NETEX_Months;
2926 IF (SiteInfo.site_type = 'S') AND (Maint_Months = 0)
2927 Begin
2928 Quantity := 000;
2929 PRINT_SW (3);
2930 I := I + 1;
2931 Quantity := 0;
2932 PRINT_SW (2);
2933 End
2934 Else I := I + 2;
2935 Maint_Months := NETEX_Months;
2936 IF (SiteInfo.site_type = 'S') AND (Maint_Months = 0)
2937 Begin
2938 Quantity := 000;
2939 PRINT_SW (3);
2940 I := I + 2;
2941 Quantity := 0;
2942 PRINT_SW (2);
2943 End
2944 Maint_Months := NETEX_Months;
2945 End
2946 I := I + 1;
APPENDIX B: MAINTENANCE MANUAL

2951  IF (Siteinfo.site_type = 'S') AND ((A400 - THYPERchannels) > 0) THEN:
2952       Begin
2953       Quantity := (A400 - THYPERchannels);
2954       PRINT SW (3);
2955       I := I + 2;  { I+108 CIP, PERKIN-ELMER }
2956       Quantity := 1;
2957       Maint_Months := SPLICENet_Months;
2958       PRINT SW (2);
2959       End
2960   Else I := I + 2;
2961   Maint_Months := NETEX_Months;
2962   I := I + 1;  { I+109 IBM NETEX Software }
2963   IF (Siteinfo.site_type = 'S') AND (A220 > 0) THEN
2964       Begin
2965       Quantity := A220;
2966       PRINT SW (3);
2967       I := I + 2;  { I+111 CIP, IBM SWS }
2968       Quantity := 1;
2969       Maint_Months := SPLICE2Net_Months;
2970       PRINT SW (2);
2971       End
2972   Else I := I + 2;
2973   Maint_Months := NETEX_Months;
2974   I := I + 1;  { I+112 UNIVAC NETEX Software }
2975   IF (Siteinfo.site_type = 'S') AND (A140 > 0) THEN
2976       Begin
2977       Quantity := A140;
2978       PRINT SW (3);
2979       I := I + 2;  { I+114 CIP, UNIVAC }
2980       Quantity := 1;
2981       Maint_Months := SPLICENet_Months;
2982       PRINT SW (2);
2983       End
2984   Else I := I + 2;
2985   Maint_Months := NETEX_Months;
2986   I := I + 1;  { I+115 UNIVAC NETEX Software }
2987   IF (Siteinfo.site_type = 'S') AND (THYPERchannels > 0) THEN
2988       Begin
2989       Quantity := THYPERchannels;
2990       PRINT SW (3);
2991       End;
2992   I := I + 2;  { I+117 CIP, ENA423 }
2993   Quantity := 1;
2994   Maint_Months := SPLICE2Net_Months;
2995   PRINT SW (2);
2996   I := I + 2;  { I+119 CIP, ENA423 }
2997   PRINT SW (1);
2998 End;  { Procedure MILLION, HYPERchannels, IN CONTRAST...}
APPENDIX B: MAINTENANCE MANUAL  Page 94

Page 61

SPLICE.PAS-include file SPLICE2.PAS  Program Listing

3001 PROCEDURE COMPUTE_DON_SOFTWARE;
3002
3003  Begin ( Procedure COMPUTE_DON_SOFTWARE );
3004
3005   I := I + 2;  // SKIPS TWO OLD DON PACKAGES
3006   Quantity := PROCESSORS;  // PER-PROCESSOR SOFTWARE
3007   Maint_Months := SPLICENet_Months;
3008   IF DON = 'Y' THEN PRINT_SW (2);  // PER-SITE SOFTWARE
3009   Maint_Months := Temp_Months;
3010   I := I + 1;  // I = 122 NETWORK MnF FACILITY GROUP
3011   IF NMF_Group = 'Y' THEN PRINT_SW (2);
3012   I := I + 1;  // I = 123 NMF BASE FACILITY
3013   IF NMF_Base_Facility = 'Y' THEN PRINT_SW (2);
3014   I := I + 1;  // I = 124 NMF PERFORMANCE MONITORING
3015   IF NMF_Performance = 'Y' THEN PRINT_SW (2);
3016   I := I + 1;  // I = 125 NMF DIAGNOSTIC MONITORING
3017   IF NMF_Diagnostics = 'Y' THEN PRINT_SW (2);
3018   I := I + 1;  // I = 126 NMF ACCOUNTING APPLICATION
3019   Quantity := Processors;
3020   I := I + 2;  // Skips 2 p/e software packages
3021   End;  // Procedure COMPUTE_DON_SOFTWARE )
3022
3023  Begin ( Procedure COMPUTE_DON_SOFTWARE );
3024
3025  COMPUTE_TANDEM_SOFTWARE;
3026  COMPUTE_HYPERCHANNEL_SOFTWARE;
3027  COMPUTE_DON_SOFTWARE;
3028  End;  // Procedure COMPUTE_DON_SOFTWARE )
3029
3030  PROCEDURE COMPUTE.Utility_SOFTWARE;
3031
3032  Begin ( Procedure COMPUTE.Utility_SOFTWARE );
3033
3034   I := I + 1;  // I = 1 Processor Utility
3035   IF Processors = 0 THEN PRINT_SW (2);  // PER-PROCESSOR SOFTWARE
3036   Quantity := Processors;
3037   I := I + 1;  // I = 1 Processor Utility
3038   IF Processors = 0 THEN PRINT_SW (2);  // PER-PROCESSOR SOFTWARE
3039   I := I + 1;  // I = 1 Processor Utility
3040   IF GateInformation = 0 THEN PRINT_SW (2);  // PER-PROCESSOR SOFTWARE
3041  Begin
3042     I := 1 + 1;  // I = 2 Processor Utility
3043   IF Quantity = 0 THEN PRINT_SW (2);  // PER-PROCESSOR SOFTWARE
3044  End
3045   Else I := 1 + 1;
3046   IF GateInformation = 0 THEN PRINT_SW (2);  // PER-PROCESSOR SOFTWARE
3047  Begin
3048     I := 1 + 1;  // I = 2 Processor Utility
3049   IF Processors = 0 THEN PRINT_SW (2);  // PER-PROCESSOR SOFTWARE
3050  End

231
APPENDIX B: MAINTENANCE MANUAL

SPLICE.PAS—include file SPLICE2.PAS Program Listing

30511 Else I := 1 + 1;
30521 I := I + 15;  
30531 I := I + 1;  
30541 IF Processors > 0 THEN PRINT_SW (1);  
30551 I := I + 1;  
30561 IF T.TEXT = 'Y' AND (Processors > 0) THEN PRINT_SW (2);  
30571 I := I + 2;  
30581 End;  
30591 PROCEDURE CONFIGURE_SOFTWARE
30601 BEGIN
30611 INITIALIZE_SOFTWARE_INPUTS;
30621 GET_SOFTWARE_INPUTS;
30631 COMPUTE_PROCESSOR_SOFTWARE;
30641 COMPUTE_COMMUNICATIONS_SOFTWARE;
30651 COMPUTEUTILITY_SOFTWARE;
30661 COMPUTE_SECTION_TOTALS ('Documentation');
30671 Mode := Document;
30681 End;  
30691 PROCEDURE INITIALIZE_LAST_SCREEN_DATA;
30701 BEGIN
30711 PROCEDURE INITIALIZE_LAST_SCREEN_DATA;
30721 BEGIN
30731 BEGIN
30741 BEGIN
30751 Computer_Ops := 0;
30761 Data_Communication := 0;
30771 Hardware_Manual := 0;
30781 Hardware_Overview := 0;
30791 Operator_Training := 0;
30801 Per_Call_Months := 1;
30811 Programmer_Ref := 0;
30821 Site_Preps := 'N';
30831 SPLICE.Net_Workshop := 0;
30841 Sys_Programmer := 0;
30851 Sys_Resource := 0;
30861 Sys_Tuning_Xray := 0;
30871 TAL := 0;
30881 Training_Group := 5;
30891 End;
30901 PROCEDURE INITIALIZE_LAST_SCREEN_DATA;
30911 PROCEDURE GET_LAST_SCREEN_DATA;
30921 Begin
30931 BEGIN
30941 BEGIN
30951 screen_fields := 14;
30961 varSS := 1;
30971 retrieveSS := False;
30981 Last_fields := False;
30991 DISPLAY_SCREEN (Screenfile);  

APPENDIX B: MAINTENANCE MANUAL

Page 63

SPLICE.PAS - include file SPLICE2.PAS

Program Listing

```pascal
REPEAT t until answerSS = 'Y'
( Display Items. Change retrieveSS to True and INPUT items)
REPEAT ( until actionSS = exitSS
CASE varSS of
1: GETINT(35,5,2,'N',Computer_Ops,'##',0,20,retSS,retrieveSS,15,11);
2: GETINT(35,7,2,'N',Programmer_Ref,'##',0,20,retSS,retrieveSS,15,11);
3: GETINT(35,9,2,'N',Hardware_Manual,'##',0,20,retSS,retrieveSS,15,11);
4: GETINT(35,11,2,'N',Sys_Programmer,'##',0,20,retSS,retrieveSS,15,11);
5: GETINT(70,5,1,'N',Training_Group,'##',1,5,retSS,retrieveSS,15,11);
6: GETINT(75,10,2,'N',Operator_Training,'##',0,20,retSS,retrieveSS,15,11);
7: GETINT(75,11,2,'N',Hardware_Overview,'##',0,20,retSS,retrieveSS,15,11);
8: GETINT(75,12,2,'N',Sys_Resource,'##',0,20,retSS,retrieveSS,15,11);
9: GETINT(75,13,2,'N',Sys_Tuning_Xray,'##',0,20,retSS,retrieveSS,15,11);
10: GETINT(75,14,2,'N',Data_Communication,'##',0,20,retSS,retrieveSS,15,11);
11: GETINT(75,15,2,'N',TAL,'##',0,20,retSS,retrieveSS,15,11);
12: GETINT(75,16,2,'N',SPLICENet_Workshop,'##',0,20,retSS,retrieveSS,15,11);
13: GETINT(75,17,2,'N',Per_Call_Months,'##',0,12,retSS,retrieveSS,15,11);
14: GETITEM(75,23,1,'Y',Site_Preps,'U','##',0,20,retSS,retrieveSS,15,11);
15: IF varSS = screen_fieldSS THEN last_fieldSS := True;
    RET_STATUS; | Check code in "retSS". Set "varSS" and "actionSS" |
    IF last_fieldSS AND (not retrieveSS) THEN
    Begin
    retrieveSS := True;
    last_fieldSS := False;
    actionSS := staySS;
    varSS := 1;
    End;
    ELSE
    last_fieldSS := False;
    UNTIL actionSS = exitSS;
    ACCEPT_INPUTS;
    UNTIL answerSS = 'Y';
End; Procedure GET_LAST_SCREEN_DATA |
PROCEDURE GET_TERMS_DOCUMENTATION;
(* This procedure simply uses the repetitive terminal end procedure *)
PROCEDURE to list the 4 categories of manuals for the user and asks
how many of each should be output on the delivery order. Output is
written to disk via the PRINT_VAR procedure,
*-described above.* Then WRITE_VALUE for actual order of items.
*)
```

Page 96
APPENDIX B: MAINTENANCE MANUAL  Page 97

SPLICE.PAS-include file SPLICE2.PAS Program Listing

3151 Begin ( Procedure CONFIGURE_DOCUMENTATION )
3152  I := 1 + 1; 
3153  Quantity := Computer_Ops; 
3154  IF Quantity > 0 THEN PRINT_DOC_OR_TRNG;
3155  I := I + 1; 
3156  Quantity := Sys_Programmer; 
3157  IF Quantity > 0 THEN PRINT_DOC_OR_TRNG;
3158  I := I + 1; 
3159  Quantity := Hardware_Manual; 
3160  IF Quantity > 0 THEN PRINT_DOC_OR_TRNG;
3161  I := I + 1; 
3162  Quantity := Programmer_Ref; 
3163  IF Quantity > 0 THEN PRINT_DOC_OR_TRNG;
3164  COMPUTE_SECTION_TOTALS ( "Training" );
3165  Mode := Train;
3166 End; (' Procedure CONFIGURE_DOCUMENTATION )
3167
3168
3169 PROCEDURE CONFIGURE_TRAINING;
3170
3171  '-------------------------------------------------------------
3172  ' This procedure simply uses the repetitive terminal out procedures
3173  ' to list the 7 categories of courses for the
3174  ' user and uses which how many of each should be output on the
3175  ' terminal. Of course, output is written to disk via the
3176  ' PRINT_DOC or PRINT procedure, described above. Uses WRITE_A_LINE
3177  ' for terminal output to disk.
3178  '-------------------------------------------------------------
3179
3180 Begin ( Procedure CONFIGURE_TRAINING )
3181  IF Train .. Group 1
3182  THEN 
3183    I := 1 + 1; 
3184    PRINT_DOC or TRNG; 
3185    PRINT_DOC or TRNG; 
3186    END;
3187  ELSE 
3188    I := 1 + 1; 
3189    PRINT_DOC or TRNG; 
3190    END;
3191  END;
3192
3193  IF Train .. Group 2
3194  THEN 
3195    I := 1 + 1; 
3196    PRINT_DOC or TRNG; 
3197    END;
3198  ELSE 
3199    I := 1 + 1; 
3200    PRINT_DOC or TRNG; 
3201    END;
3202
3203  IF Train .. Group 3
3204  THEN 
3205    I := 1 + 1; 
3206    PRINT_DOC or TRNG; 
3207    END;
3208  ELSE 
3209    I := 1 + 1; 
3210    PRINT_DOC or TRNG; 
3211    END;
3212
3213  End;
3214
3215  4: Begin

234
I := I + 4; \{ I=160 Training Group IV \}

Quantity := 1;

PRINT_DOC_or_TRNG;

End;

5: I := I + 4;

I := I + 1; \{ I=161 Operator Training Course \}

Quantity := Operator_Training;

IF Operator_Training > 0 THEN PRINT_DOC_or_TRNG;

I := I + 1; \{ I=162 Hardware Overview Course \}

Quantity := Hardware_Overview;

IF Hardware_Overview > 0 THEN PRINT_DOC_or_TRNG;

I := I + 1; \{ I=163 System Resource Mgmt Course \}

Quantity := Sys_Resource;

IF Sys_Resource > 0 THEN PRINT_DOC_or_TRNG;

I := I + 1; \{ I=164 Systems Tuning and XRAY Course \}

Quantity := Sys_Tuning_Xray;

IF Sys_Tuning_Xray > 0 THEN PRINT_DOC_or_TRNG;

I := I + 1; \{ I=165 Data Communications Course \}

Quantity := Data_Communication;

IF Data_Communication > 0 THEN PRINT_DOC_or_TRNG;

I := I + 1; \{ I=166 TANDEM Application Lang Course \}

Quantity := TAL;

IF TAL > 0 THEN PRINT_DOC_or_TRNG;

I := I + 1; \{ I=167 SPLICENet WORKSHOP \}

Quantity := SPLICENet_Workshop;

IF SPLICENet_Workshop > 0 THEN PRINT_DOC_or_TRNG;

COMPUTE_SECTION_TOTALS ("Maintenance");

Mode := Maint;

End; \{ Procedure CONFIGURE_MAINT \}

PROCEDURE CONFIGURE_MAINT;

\*****************************************************

This procedure is used to write-out the three lines required on 
the delivery orders for maintenance. Both PM On-Call and On-Call are 
written out with Quantity = 1 and all remaining items = 0. The 
Maint Months of Emergency Maintenance are loaded into Quantity 
of the output, multiplied by the updated emergency 
maintenance rate and then written to disk. The applicable 
uplift rate is written out. All other fields are = 0.

\*****************************************************

PROCEDURE PRINT_MAINT;

\*****************************************************

Sets parameters for the three categories to be output on the 
delivery order. Sets MONTHS to 0 and Maint_Factor to the 

APPENDIX B: MAINTENANCE MANUAL

Page 99

SPLICE.PAS--include file SPLICE2.PAS Program Listing

3251 (* emerg_maint_rate input by the user. Uses WRITE_A_LINE to *)
3252 (* actually write to disk. *)
3253 (* *****************************************************************)
3254
3255 Begin (* FDC Emergency Maint *)
3256 Maint_Months := 0;
3257 Maint_Factor := Emerg_Maint_Rate;
3258 Extended_Price := 0;
3259 WRITE_A_LINE;
3260 End; (* Procedure PRINT_MAINT *)
3261
3262 Begin (* Procedure CONFIGURE_MAINTENANCE *)
3263 1 := 1 + 1; (* I=168 PM On-Call *)
3264 IF (Totals [0, 1] > 0) OR (Totals [1, 1] > 0) OR (Totals [2, 1] > 0) OR
3265 (Totals [3, 1] > 0) OR (Totals [4, 1] > 0) OR (Totals [5, 1] > 0) OR
3266 (Totals [0, 2] > 0) OR (Totals [1, 2] > 0) OR (Totals [2, 2] > 0) OR
3267 (Totals [3, 2] > 0) OR (Totals [4, 2] > 0) OR (Totals [5, 2] > 0) THEN
3268 Begin
3269 Quantity := 1;
3270 PRINT_MAINT;
3271 1 := 1 + 1; (* I=169 Skips PM Per-Call Maintenance *)
3272 1 := 1 + 1; (* I=170 On-Call Maint *)
3273 PRINT_MAINT;
3274 End
3275 ELSE 1 := 1 + 2;
3276 Quantity := Per_Call_Months;
3277 IF Quantity > 0 THEN PRINT_MAINT;
3278 COMPUTE_SECTION_TOTALS ('Other');
3279 End; (* Procedure CONFIGURE_MAINTENANCE *)
3280
3281 Begin (* Procedure CONFIGURE_COMPONENTS *)
3282 CONFIGURE_HARDWARE;
3283 CONFIGURE_SOFTWARE;
3284 INITIALIZE_LAST_SCREEN_DATA;
3285 GET_LAST_SCREEN_DATA;
3286 CONFIGURE_DOCUMENTATION;
3287 CONFIGURE_TRAINING;
3288 CONFIGURE_MAINTENANCE;
3289 End; (* Procedure CONFIGURE_COMPONENTS *)
3290 (* Name of work procedures include title *)
3291 PROCEDURE SUMMARIZE;
3292
APPENDIX B: MAINTENANCE MANUAL

SPLICE.PAS Program Listing

3300 Const
3301   LF : Char = #10;  { Decimal Value for an ASCII line feed } 
3302   CR : Char = #13;  { Decimal Value for an ASCII carriage return } 
3303   CtrlZ : Char = ^Z; { Value of ASCII "Control-Z" end-of-file marker } 
3304 
3305 Var
3306   System_Downtime : Real;
3307   
3308 Begin  ( Procedure SUMMARIZE )
3309   System_Downtime := (((Subtotals [0, 1] + Subtotals [0, 3] + Subtotals [1, 1] 
3310     + Subtotals [1, 3] + Costtable[1].purchprice) / 48) 
3311     + System_Downtime_Component) * 0.0125;
3312   WRITELN (Diskfile);
3313   WRITELN (Diskfile, "NOTES:");
3314   WRITELN (Diskfile);
3315   WRITELN (Diskfile, "MAINTENANCE OPTION = ", Siteinfo.maint_options, ");
3316   WRITELN (Diskfile, "MAINTENANCE REPAIR AND RESPONSE = ", 
3317     Siteinfo.maint_response, ");
3318   WRITELN (Diskfile);
3319   WRITELN (Diskfile, "SYSTEM DOWNTIME CREDIT FACTOR PER HOUR EQUALS: $", 
3320     "", "", "", System_Downtime);
3321   WRITELN (Diskfile);
3322   WRITELN (Diskfile, "CARD READER AND CARD READER PUNCH CAPABILITIES TEST REQUIREMENTS ARE WAIVED.");
3323   WRITELN (Diskfile);
3324   WRITELN (Diskfile, ""SYSTEM DOWNTIME CREDIT FACTOR PER HOUR EQUALS: $", 
3325     "", "", "", System_Downtime);
3326   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3327   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3328   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3329   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3330   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3331   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3332   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3333   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3334   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3335   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3336   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3337   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3338   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3339   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3340   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3341   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3342   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3343   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3344   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3345   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3346   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3347   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3348   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3349   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3350   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3351   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3352   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3353   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3354   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3355   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3356   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3357   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3358   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3359   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3360   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3361   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3362   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3363   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3364   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3365   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3366   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3367   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3368   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3369   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3370   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3371   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3372   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3373   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3374   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3375   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3376   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3377   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3378   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3379   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3380   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3381   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3382   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3383   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3384   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3385   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3386   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3387   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3388   WRITELN (Diskfile, "", "", "", "", System_Downtime);
3389   ClrScr;
3390   OKNOXY (4, 91);
3391   WRITELN ("Thank you for using the SPLICE configurer.");
3392   WRITELN;
3393   WRITELN;
3394   TextColor (15);
3395   WRITELN ("Your output file is called ':48, PRF_FILE 'time',");
3396   WRITELN;
3397   WRITELN;
3398   TextColor (11);
3399   WRITELN ("The output file is ready for import into INUS v. ‘5:");
APPENDIX B: MAINTENANCE MANUAL

Page 68

SPLICE.PAS Program Listing

3350 | TextColor (15);
3351 | End;  // Procedure SUMMARIZE
3352
3353
3354 | Begin
3355 | INITIALIZE;
3356 | CONFIGURE_COMPONENTS;
3357 | SUMMARIZE;
3358 | End.


**CONFMOD.PRG Program Listing**

```plaintext
* PROCEDURE CONFMOD.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
*           LCDR WINSTON H. BUCKLEY, SC, USN
*           LCDR ROBERT F. BRADO, USN
*           LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : PROVIDE THE USER THE OPPORTUNITY TO MODIFY OR REVIEW
*            ALL DATA IN THE SITE CONFIGURATION DATABASE.

* INPUT FILES : NONE
* OUTPUT FILE : NONE

* MODULES CALLED : CONFUPD.PRG, CONFPREV.PRG

* CALLED BY : MAINMENU.PRG

* LOCAL VARIABLES: SELEKT

* DATE LAST TIME MODIFIED = 22 DECEMBER 1985

* DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE
* SELECTION.

STORE "1" TO SELEKT
DO WHILE SELEKT < "3"
SET COLOR TO W/B, W/B
CLEAR
?? FLASH "W.CONFMOD/"
SET CONSOLE OFF
WAIT TO SELEKT
SET CONSOLE ON

* PROCESS ROUTINE BASED ON THE USER'S SELECTION.

DO CASE

* CALL THE SITE CONFIGURATION UPDATE PROGRAM.
CASE SELEKT = "1"
   DO CONFUPD

* CALL THE SITE CONFIGURATION REVIEW PROGRAM.
CASE SELEKT = "2"
   DO CONFPREV

* RETURN TO THE MAIN MENU PROGRAM.
CASE SELEKT = "3"

ENDCASE

STORE "1" TO SELEKT
DO WHILE SELEKT < "3"
SET COLOR TO W/B, W/B
CLEAR
?? FLASH "W.CONFMOD/"
SET CONSOLE OFF
WAIT TO SELEKT
SET CONSOLE ON

* PROCESS ROUTINE BASED ON THE USER'S SELECTION.

DO CASE

* CALL THE SITE CONFIGURATION UPDATE PROGRAM.
CASE SELEKT = "1"
   DO CONFUPD

* CALL THE SITE CONFIGURATION REVIEW PROGRAM.
CASE SELEKT = "2"
   DO CONFPREV

* RETURN TO THE MAIN MENU PROGRAM.
CASE SELEKT = "3"

ENDCASE

```

APPENDIX B: MAINTENANCE MANUAL

Page 2

CONFMOD.PRG Program Listing

51 | *
52 ENDDO WHILE SELEKT < "3"
53 | *
54 | RETURN TO THE CALLING PROGRAM
55 | *
56 RETURN
57 | ****************************************************
* PROCEDURE CONFREV.PRG

* AUTHORS
  LCDR EDWARD J. CASE, SC, USN
  LCDR WINSTON H. BUCKLEY, SC, USN
  LCDR ROBERT F. BRADO, USN
  LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE
  TO ENABLE THE USER TO REVIEW ANY DATA ELEMENT IN
  THE SITE NAME DATABASE.

* INPUT FILES
  CONFIG.DBF INDICES: CONFIG.NDX

* OUTPUT FILES
  NONE

* CALLED BY
  CONFMTOD.PRG

* MODULES CALLED
  DELAY.PRG

* GLOBAL VARIABLE
  HISITE, LOSITE

* LOCAL VARIABLES
  ACCEPT, ANS, COICE, ERROR, FIRSTREC, LASTREC,
  MADDI, MADD2, MCITY, MCO, MESSAGE, MNAME, MNAMEFL,
  MNOPT, MRESP, MSITE, MSTATE, MTYPE, MZIP

* DATE LAST TIME MODIFIED
  23 DECEMBER 1985

* CASE SELECTION = 2
  REVIEW EXISTING RECORDS

* USE THE SITE NAME (CONFIG) DATABASE USING
  THE SITE NUMBER INDEX.

SET ESCAPE OFF
SET TALK OFF
USE CONFIG
GO TOP
SET COLOR TO W+/B,W+/B,B
CLEAR
IF EOF() = .T. THEN
  SET COLOR TO W+/R, W+/R
  @13,24 SAY "The SITE NAME Database is EMPTY! "
ENDIF
DO DELAY
RETURN
?? FLASH + "S.SITENAME.SCR/"
@ 24,0 SAY SPACE (80)
SET COLOR TO R+/R, R+/R
@ 3,23 SAY 'SITE ADDRESS DATA REVIEW FORMAT'
STORE 'Enter 00 to start at TOF, 99 to start at EOF, or a site ' +
' number between ' + HISITE + ' and ' + HISITE + ' TO MESSAGE
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE

241
DO WHILE .T.
SET COLOR TO /BR, /BR
STORE '00' TO MSITE
@ 7,25 GET MSITE PICT '99'
READ
IF .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
STORE ' Response must be between '+ LOSITE + ' and '+ HISITE + ', Zero (00) or 99 ' TO ERROR
@ 24,13 SAY ERROR
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
LOOP
ELSE
IF (MSITE = '00' .OR. MSITE = '99') THEN
USE CONFIG
IF MSITE = '00' THEN
GO BOTTOM
STORE RECONO() TO LAST_REC
GO TOP
STORE RECONO() TO FIRST_REC
ELSE
GO TOP
STORE RECONO() TO FIRST_REC
GO BOTTOM
STORE RECONO() TO LAST_REC
ENDIF MSITE = '00'
EXIT
ELSE
USE CONFIG INDEX CONFIG,ndx
GO TOP
STORE RECONO() TO FIRST_REC
GO BOTTOM
STORE RECONO() TO LAST_REC
FIND &MSITE
IF EOF() = .T. THEN
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
STORE " No records exist for site number " + MSITE + ", try again " TO ERROR
@ 24,16 SAY ERROR
SET COLOR TO W+/R, W+/R
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
LOOP
ELSE
APPENDIX B: MAINTENANCE MANUAL

Page 3

CONFREV.PRG Program Listing

101 | EXIT
102 | ENDF EOF() = .T.
103 | ENDF (MSITE = '00' .OR. MSITE = '99')
104 | ENDF
105 | ENDDO WHILE .T.
106 | *
107 | SET COLOR TO W/B, W/B
108 | @ 24,0 SAY SPACE(80)
109 | *
110 | DO WHILE .T.
111 | SET COLOR TO R+/B, R+/B
112 | @ 5,47 SAY RECNO() PICT "999"
113 | SET COLOR TO /BR, /BR
114 | @ 7,25 SAY SITENO PICT "99"
115 | @ 8,25 SAY SITENAME PICT """""""""""""""""""
116 | @ 9,25 SAY SITECO PICT """""""""""
117 | @ 10,25 SAY SITENAMEFL PICT """""""""""
118 | @ 11,25 SAY SITEADD1 PICT """""""""""
119 | @ 12,25 SAY SITEADD2 PICT """""""""""
120 | @ 13,25 SAY SITECITY PICT """""""""""
121 | @ 14,25 SAY SITESTATE PICT """"""
122 | @ 15,25 SAY SITEZIP PICT "9999999999"
123 | @ 16,25 SAY SITETYPE PICT """"""
124 | @ 17,35 SAY MAINTOPT PICT """
125 | @ 18,35 SAY MAINTREP PICT """
126 | SET COLOR TO R+/B, R+/B
127 | STORE "N" TO CHOICE
128 | @ 22,68 GET CHOICE PICT "1"
129 | READ
130 | *
131 | ENSURE THAT THE USER'S PROMPT IS EITHER "N", "P" OR "X"
132 | *
133 | DO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X")
134 | IF .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X") THEN
135 | SET COLOR TO W+/R,W+/R
136 | @ 24,21 SAY "Response must be either N, P or X"
137 | DO DELAY
138 | STORE "N" TO CHOICE
139 | ENDF
140 | SET COLOR TO R+/B, R+/B
141 | @ 22,68 GET CHOICE PICT "1"
142 | READ
143 | ENDDO
144 | *
145 | SKIP TO THE NEXT RECORD TO BE REVIEWED
146 | *
147 | IF CHOICE = ",1" THEN
148 | IF RECNO () = LAST_REC THEN
149 | (Z) TOP
150 | ELSE
APPENDIX B: MAINTENANCE MANUAL

Page 4

CONFREV.PRG Program Listing

151  SKIP
152     ENDF
153  ENDF
154  *
155  * SKIP TO THE PREVIOUS RECORD TO BE REVIEWED
156  *
157  IF CHOICE = "P" THEN
158     IF RECV() = FIRST_REC THEN
159         (G) BOTTOM
160     ELSE
161         SKIP -1
162     ENDF
163  ENDF
164  *
165  * USER HAS DECIDED TO EXIT THE REVIEW
166  *
167  IF CHOICE = "X" THEN
168      EXIT
169  ENDF
170  *
171  ENDDO WHILE .T.
172  *
173  * RETURN TO CALLING PROGRAM.
174  *
175  RELEASE ALL LIKE M*, ACCEPT, ANS, CHOICE, ERROR, FIRST_REC, LAST_REC
176  CLOSE DATABASES
177  RETURN
178  ************************************************************************************
* PROCEDURE CONFUPD.PRG

**

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
* LCDR WINSTON H. BUCKLEY, SC, USN
* LCDR ROBERT F. BRADO, USN
* LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : TO ENABLE THE USER TO MODIFY ANY DATA ELEMENT IN
* THE SITE NAME DATABASE.

* INPUT FILES : CONFIG.DBF INDICES: CONFIG.NDX
* OUTPUT FILES : CONFIG.DBF, INDICES: CONFIG.NDX
* SUBRoutines CALLED : DELAY.PRG
* CALLED BY : CONFUPD.PRG

* GLOBAL VARIABLE: HBSITE, LOSITE

* LOCAL VARIABLES: ACCEPT, ANS, CHOICE, ERROR, FIRST_REC, LAST_REC,
  MADD1, MADD2, MCIY, MOD, MESSAGE, MNAME, MNAMEFL,
  MOPT, MRESP, NSITE, MSTATE, MTYPE, MZIP, SAVEIT

* DATE LAST TIME MODIFIED =========> 23 DECEMBER 1985 <========

BEGIN
CASE SELECTION = 1 UPDATE EXISTING RECORDS
USE THE SITE NAME (CONFIG.DBF) DATABASE USING THE SITE NUMBER INDEX.

SET ESCAPE OFF
SET SCOREBOARD OFF
SET TALK OFF
USE CONFIG
GO TOP
SET COLOR TO W+/B, W+/B, B
CLEAR
IF EXF() = .T. THEN
  SET COLOR TO W*/R, W*/R
  \13,24 SAY " The SITE NAME Database is EMPTY! "
  \0 DELAY
  RETURN
ENDIF

? FLASH : "S.SITENAME.SCR/"
\24,0 SAY SPACE(80)
SET COLOR TO R+/R+/R+
\3,23 SAY ' SITE ADDRESS DATA UPDATE FORMAT '
STORE 'Enter 90 to start at WF, 99 to start at DF, or a site:
  ' + ' + HBSITE + ' and ' + HBSITE + ' TO MESSAGE
APPENDIX B: MAINTENANCE MANUAL  Page 109

Page 2

CONFUPD.PRG Program Listing

51 | SET COLOR TO /W, /W
52 | @ 24,0 SAY MESSAGE
53 | DO WHILE .T.
54 | SET COLOR TO /BR, /BR
55 | STORE '00' TO MSITE
56 | @ 7,25 GET MSITE PICT '99'
57 | READ
58 | IF .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
59 | SET COLOR TO W/B, W/B
60 | @ 24,0 SAY SPACE(80)
61 | SET COLOR TO W+/R, W+/R
62 | STORE 'Response must be between ' + LOSITE + ' and ' + HISITE + ', Zero (00) or 99 ' TO ERROR
63 | @ 24,13 SAY ERROR
64 | DO DELAY
65 | SET COLOR TO /W, /W
66 | @ 24,0 SAY MESSAGE
67 | LOOP
68 | ELSE
69 | IF (MSITE = '00' .OR. MSITE = '99') THEN
70 | USE CONFIG
71 | IF MSITE = '00' THEN
72 | GO BOTTOM
73 | STORE RECNO() TO LAST_REC
74 | GO TOP
75 | STORE RECNO() TO FIRST_REC
76 | ELSE
77 | GO TOP
78 | STORE RECNO() TO FIRST_REC
79 | GO BOTTOM
80 | STORE RECNO() TO LAST_REC
81 | ENDF IF MSITE = '00'
82 | EXIT
83 | ELSE
84 | USE CONFIG INDEX CONFIG.NDX
85 | GO TOP
86 | STORE RECNO() TO FIRST_REC
87 | GO BOTTOM
88 | STORE RECNO() TO LAST_REC
89 | FIND MSITE
90 | IF EOF() = .T. THEN
91 | SET COLOR TO W/B, W/B
92 | @ 24,0 SAY SPACE(80)
93 | STORE "No records exist for site number " + MSITE + "", try again" TO ERROR
94 | @ 24,16 SAY ERROR
95 | DO DELAY
96 | SET COLOR TO W+/R, W+/R
97 | LOOP
98 | SET COLOR TO /W, /W
99 | @ 24,0 SAY MESSAGE
100 |
APPENDIX B: MAINTENANCE MANUAL

Page 3

CONFUPD.PRG Program Listing

101 | LOOP
102 | ELSE
103 | EXIT
104 | ENDIF EOF() = .T.
105 | ENDIF (MSITE = '00' .OR. MSITE = '99')
106 | ENDIF
107 | ENDDO WHILE .T.
108 | *
109 | SET COLOR TO W/B, W/B
110 | @ 24,0 SAY SPACE(80)
111 | *
112 | STORE SPACE(16) + 'Press "Page Down" key to terminate record update' +;
113 | SPACE(16) TO MESSAGE
114 | STORE 1 TO INTRO
115 | DO WHILE .T.
116 | SET COLOR TO /W, /W
117 | @ 24,0 SAY MESSAGE
118 | *
119 | * INFORM THE USER OF HOW TO TERMINATE THE UPDATE OF A RECORD
120 | *
121 | IF INTRO = 1 THEN
122 | STORE 0 TO INTRO
123 | ?? FLASH + "W,CONFUPD/"
124 | SET CONSOLE OFF
125 | WAIT TO ANS
126 | SET CONSOLE ON
127 | ENDIF
128 | *
129 | * STORING THE OLD RECORD TO A WORK RECORD AREA. THE M PREFIX
130 | * INDICATES MEMORY VARIABLES DISTINGUISHING THEM FROM THEIR
131 | * CORRESPONDING DATABASE FIELDS.
132 | *
133 | STORE SITENO  TO MSITE
134 | STORE SITENAME TO MNAME
135 | STORE SITECO  TO MCO
136 | STORE SITENAMEFL TO MNAMEFL
137 | STORE SITEADD1 TO MAUDD1
138 | STORE SITEADD2 TO MAUDD2
139 | STORE SITECITY TO MCITY
140 | STORE SITESTATE TO MSTATE
141 | STORE SITEZIP  TO MZIP
142 | STORE SITETYPE TO MTYPE
143 | STORE MAINTOPT TO MOPT
144 | STORE MAINTRESP TO MRESP
145 | *
146 | SET COLOR TO R+/B, R+/B
147 | @ 5,47 SAY RECNO() PICT "999"
148 | SET COLOR TO /BR, /BR
149 | *
150 | @ 7,25 SAY MSITE PICT "99"
APPENDIX B: MAINTENANCE MANUAL

CONFUPD.PRG Program Listing

151 @ 8,25 GET MNAME PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
152 @ 9,25 GET MCO PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!"
153 @ 10,25 GET MNAMEFL PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
154 @ 11,25 GET MADD1 PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
155 @ 12,25 GET MADD2 PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
156 @ 13,25 GET MCITY PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
157 @ 14,25 GET MSTATE PICT "!!"
158 @ 15,25 GET MZIP PICT "9999999999"
159 @ 16,25 SAY MTYPE PICT "!!!!"
160 @ 17,35 GET MOPP PICT "!!!!"
161 @ 18,35 GET MRESP PICT "!

READ

163 * CHECK TO SEE IF ANY RECORD WAS CHANGED
164 *
165 SET COLOR TO W/B, W/B
166 @ 24,0 SAY SPACE(80)
167 STORE 1 TO SAVEIT
168 IF (SITENO = MSITE)
169 IF (SITENAME = MNAME)
170 IF (SITECO = MCO)
171 IF (SITENAMEFL = MNAMEFL)
172 IF (SITEADD1 = MADD1)
173 IF (SITEADD2 = MADD2)
174 IF (SITEADD2 = MADD2)
175 IF (SITECITY = MCITY)
176 IF (SITESTATE = MSTATE)
177 IF (SITEZIP = MZIP)
178 IF (SITETYPE = MTYPE)
179 IF (MAINIOPP = MOPP)
180 IF (MAINTRESP = MRESP)
181 STORE 0 TO SAVEIT

182 ENDIF
183 ENDIF
184 ENDIF
185 ENDIF
186 ENDIF
187 ENDIF
188 ENDIF
189 ENDIF
190 ENDIF
191 ENDIF
192 ENDIF
193 ENDIF

194 * ASK THE USER IF HE/SHE DESIRES TO ACCEPT THE CHANGES, ONLY IF ANY
195 * CHANGES WERE MADE
196 *
197 IF SAVEIT = 1 THEN
198 SET COLOR TO W+/B, W+/B
200 "Do you want to accept the changes? (Yes or No):"
APPENDIX B: MAINTENANCE MANUAL

CONFUPD.PRG Program Listing

201 | SET COLOR TO R+/B, R+/B
202 | @ 20,49 SAY "Y"
203 | @ 20,56 SAY "N"
204 | STORE "N" TO ACCEPT
205 | @ 20,62 GET ACCEPT PICT ":!
206 | READ
207 | * ENSURE THAT THE USER'S PROMPT IS EITHER "Y" OR "N"
208 | *
209 | DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
210 | IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
211 | SET COLOR TO W/B, W/B
212 | @ 24,0 SAY SPACE(80)
213 | SET COLOR TO W+/R, W+/R
214 | @ 24,24 SAY "Response must be either N or Y"
215 | DO DELAY
216 | STORE "N" TO ACCEPT
217 | ENDIF
218 | SET COLOR TO R+/B, R+/B
219 | @ 20,62 GET ACCEPT PICT ":!
220 | READ
221 | ENDDO
222 | @ 20,10 SAY SPACE (60)
223 | * STORING THE CORRECTED EDIT FIELDS FROM THE WORK AREA.
224 | *
225 | IF ACCEPT = "Y" THEN
226 | REPLACE SITE.NO WITH MSITE
227 | REPLACE SI.TENAME WITH MNAME
228 | REPLACE SITE.CO WITH MCO
229 | REPLACE SITE.NAMEFL WITH MNAMEFL
230 | REPLACE SITE.ADD1 WITH MAD1
231 | REPLACE SITE.ADD2 WITH MAD2
232 | REPLACE SITE.CITY WITH MCITY
233 | REPLACE SITE.STATE WITH MSTATE
234 | REPLACE SITE.ZIP WITH MZIP
235 | REPLACE SITE.TYPE WITH MTYPE
236 | REPLACE MAIN.TOPT WITH MUPT
237 | REPLACE MAIN.IRESP WITH MRESP
238 | ENDIF
239 | ENDIF
240 | *
241 | SET COLOR TO R+/B, R+/B
242 | STORE "N" TO CHOICE
243 | @ 22,68 GET CHOICE PICT ":!
244 | READ
245 | * ENSURE THAT THE USER'S PROMPT IS EITHER "N", "P" OR "X"
246 | *
247 | DO WHILE .NOT. (CHOICE = "P" .OR. CHOICE = "X")
248 | *
249 | *
250 |
IF .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X") THEN
    SET COLOR TO W/B, W/B
    @ 24,0 SAY SPACE(80)
    SET COLOR TO W+/R, W+/R
    @ 24,23 SAY " Response must be either N, P or X "
    DO DELAY
    STORE "N" TO CHOICE
END IF
SET COLOR TO R+/B, R+/B
@ 22,68 GET CHOICE PICI'
READ ENDDO
* * * SK I P TO THE NEXT RECORD TO BE REVIEWED *
* IF CHOICE = "N" THEN
    IF RECNO() = LAST_REC THEN
        GO TOP
    ELSE
        SKIP
    ENDIF
ENDIF
* * * SK I P TO THE PREVIOUS RECORD TO BE REVIEWED *
* IF CHOICE = "P" THEN
    IF RECNO() = FIRST_REC THEN
        GO BOTTOM
    ELSE
        SKIP -1
    ENDIF
ENDIF
* * USER HAS DECIDED TO EXIT THE REVIEW *
* IF CHOICE = "X" THEN
    EXIT
ENDIF
* ENDWHILE .T.
* RETURN TO CALLING PROGRAM.
* RELEASE ALL LIKE M*, ACCEPT, ANS, CHOICE, ERROR, FIRST_REC, LAST_REC, SAVFIT
CLOSE DATABASES
RETURN
* PROCEDURE DATERPTS.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
* LCDR WINSTON H. BUCKLEY, SC, USN
* LCDR ROBERT F. BRADO, USN
* LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : PROVIDE THE USER A SELECTION OF EFFECTIVE DELIVERY
ORDER DATE LEVEL REPORTS.

* INPUT FILES : NONE
* OUTPUT FILES : NONE
* CALLED BY : REPORCMD.PRG
* LOCAL VARIABLES: DATERPTS
* DATE LAST TIME MODIFIED =========> 18 DECEMBER 1985 <=========
* DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE SELECTION.
* STORE "1" TO DATERPTS
DO WHILE DATERPTS < "4"
  SET COLOR TO W/B, W/B, B
  CLEAR
  ?? FLASH + "W.DATERPTS/"
  SET CONSOLE OFF
  WAIT TO DATERPTS
  SET CONSOLE ON
  * PROCESS ROUTINE BASED ON THE USER'S SELECTION.
  * DO CASE
  *
  * CALL THE EQUIPMENT EFFECTIVE DELIVERY ORDER DATE LEVEL REPORT
  * WITH UNIT COST PROGRAM.
  CASE DATERPTS = "1"
    DO EQPDTPRC
  *
  * CALL THE EQUIPMENT EFFECTIVE DELIVERY ORDER DATE LEVEL REPORT
  * WITHOUT UNIT COST PROGRAM.
  CASE DATERPTS = "2"
    DO EQPDTPNFC
  *
  * CALL THE SERIAL NUMBER EFFECTIVE DELIVERY ORDER DATE LEVEL REPORT.
  CASE DATERPTS = "3"
    DO SNODTRPT
RETURN TO THE SPLICE REPORTING LEVEL MENU.
CASE DATERPTS = "4"
ENDCASE
ENDDO (WHILE DATERPTS = "4")
RETURN TO THE CALLING PROGRAM
RETURN
*******************************************************************************
APPENDIX B: MAINTENANCE MANUAL Page 116

Page 1

DELAY.PRG Program Listing

* PROCEDURE DELAY.PRG

* AUTHORS
  LCDR EDWARD J. CASE, SC, USN
  LCDR WINSTON H. BUCKLEY, SC, USN
  LCDR ROBERT J. BRADO, USN
  LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE
  TO PROVIDE A SHORT DELAY AFTER THE DISPLAY OF AN
  ERROR MESSAGE TO THE USER SUFFICIENT TIME TO READ
  THE MESSAGE.

* INPUT FILES
  NONE

* OUTPUT FILES
  NONE

* CALLED BY
  SELECTOR.PRG, MAINMENU.PRG, CONFREV.PRG, CONFUPD.PRG

* LOCAL VARIABLES: DELAY

* DATE LAST TIME MODIFIED 18 DECEMBER 1985

STORE 1 TO DELAY
DO WHILE DELAY < 60
  STORE DELAY + 1 TO DELAY
ENDDO DELAY < 60

* CLEAR OUT THE ERROR MESSAGE
SET COLOR TO W+/B, W+/B
@ 24,0 SAY SPACE (80)

* RETURN TO THE CALLING PROGRAM

RETURN

*********************************************************************

253
**APPENDIX B: MAINTENANCE MANUAL**

DESPMOD.PRG Program Listing

```plaintext
* PROCEDURE DESPMOD.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
  * LCDR WINSTON H. BUCKLEY, SC, USN
  * LCDR ROBERT F. BRADO, USN
  * LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : PROVIDE THE USER THE OPPORTUNITY TO MODIFY OR REVIEW
  * ALL DATA IN THE DESCRIPTION DATABASE.

* INPUT FILES : NONE

* OUTPUT FILE : NONE

* CALLED BY : MAINMENU.PRG

* MODULES CALLED : DESPPUPD.PRG, DESPPREV.PRG, DELAY.PRG

* LOCAL VARIABLES: SELEKT

* DATE LAST TIME MODIFIED: 22 DECEMBER 1985

* DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE
  * SELECTION.

STORE "1" TO SELEKT
DO WHILE SELEKT < "3"
  SET COLOR TO W/B, W/B
  CLEAR
  ?? FLASH + "W,DESPMOD/"
  SET CONSOLE OFF
  WAIT TO SELEKT
  SET CONSOLE ON

* PROCESS ROUTINE BASED ON THE USER'S SELECTION.

DO CASE
  * CALL THE DESCRIPTION UPDATE PROGRAM.
  CASE SELEKT = "1"
    DO DESPPUPD
  * CALL THE DESCRIPTION REVIEW PROGRAM.
  CASE SELEKT = "2"
    DO DESPPREV
  * RETURN TO THE MAIN MENU PROGRAM.
  CASE SELEKT = "3"
ENDCASE
```

Page 254
DESPMOD.PRG Program Listing

51  *
52  ENDCO (WHILE SELEKT = "3")
53  *
54  * RETURN TO THE CALLING PROGRAM
55  *
56  RETURN
57  ********************************************
* PROCEDURE DESPPREV.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
* LCDR WINSTON H. BUCKLEY, SC, USN
* LCDR ROBERT F. BRADO, USN
* LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : TO ENABLE THE USER TO REVIEW ANY DATA ELEMENT IN
* THE DESCRIPTION DATABASE.

* INPUT FILES : DESCRIP.DBF INDICES: DESCRIP.NDX

* OUTPUT FILES : DESCRIP.DBF, INDICES: DESCRIP.NDX

* MODULES CALLED : DELAY.PRG

* CALLED BY : DESPMOD.PRG

* GLOBAL VARIABLE: LOFNUM, HIFNUM

* LOCAL VARIABLES: ACCEPT, ANS, CHOICE, ERROR, FIRSTREC,
* IAST
* MBMAINT, MCLIN, MDESCRIP, MESSAGE, MFEAT, MMODELNO, MTCOMP

* DATE LAST TIME MODIFIED = 23 DECEMBER 1985

BEGIN

CASE SELECTION = 2 REVIEW EXISTING RECORDS

SET ESCAPE OFF
SET TALK OFF
USE DESCRIP
CD TOP
SET COLOR TO W+/B, W+/B, B
CLEAR
IF EOF() = .T. THEN
SET COLOR TO W+/R, W+/R
$13,17 SAY "The EQUIPMENT DESCRIPTION Database is EMPTY!"
CD DELAY
RETURN
ENDIF

?? FLASH + "S.DESCRIPT.SCR/"
@ 24,0 SAY SPACE(80)
@ 2,39 SAY "REVIEW"
STORE 'Enter 00 to start at TOF, 99 to start at EOF, or a six digit '
'MESSAGE
'feature number ' TO MESSAGE
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
DO WHILE .T.
SET COLOR TO /BR, /BR
STORE '00' TO MFEAT
@ 6,45 GET MFEAT PICT '999999'
READ
IF .NOT. ((MFEAT >= LOFNUM AND MFEAT <= HIFNUM) .OR.;
    MFEAT = '00' .OR. MFEAT = '99')
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
STORE 'Response must be between ' + LOFNUM + ' and ' +;
    HIFNUM + ', Zero (00) or 99 ' TO ERROR
@ 24,8 SAY ERROR
DO DELAY
SET COLOR TO /W, /W
Q 24,0 SAY MESSAGE
LOOP
ELSE
IF (MFEAT = '00' .OR. MFEAT = '99') THEN
    USE DESCRIPT
    IF MFEAT = '00' THEN
        GO BOTTOM
        STORE RECN0() TO LAST_REC
        GO TOP
        STORE RECN0() TO FIRST_REC
    ELSE
        IF MFEAT = '99' THEN
            GO TOP
            STORE RECN0() TO FIRST_REC
            GO BOTTOM
            STORE RECN0() TO LAST_REC
            ENDF MFEAT = '99'
        ENDF MFEAT = '00'
    STORE FEATURENO '1 MFEAT
EXIT
ELSE
    USE DESCRIPT INDEX DESCRIPT.NDX
    GO TOP
    STORE RECN0() TO FIRST_REC
    GO BOTTOM
    STORE RECN0() TO LAST_REC
    FIND &MFEAT
    IF EOF() = .T. THEN
        SET COLOR TO W/B, W/B
        @ 24,0 SAY SPACE(80)
        SET COLOR TO W+/R, W+/R
        STORE 'No record exists for feature number ' +;
            MFEAT + ', try again ' TO ERROR
        @ 24,12 SAY ERROR
        DO DELAY
    ELSE
        USE DESCRIPT INDEX DESCRIPT.NDX
        GO TOP
        STORE RECN0() TO FIRST_REC
        GO BOTTOM
        STORE RECN0() TO LAST_REC
        FIND &MFEAT
        IF EOF() = .T. THEN
            SET COLOR TO W/B, W/B
            @ 24,0 SAY SPACE(80)
            SET COLOR TO W+/R, W+/R
            STORE 'No record exists for feature number ' +;
                MFEAT + ', try again ' TO ERROR
            @ 24,12 SAY ERROR
            DO DELAY
        ELSE
            USE DESCRIPT INDEX DESCRIPT.NDX
            GO TOP
            STORE RECN0() TO FIRST_REC
            GO BOTTOM
            STORE RECN0() TO LAST_REC
            FIND &MFEAT
            IF EOF() = .T. THEN
                SET COLOR TO W/B, W/B
                @ 24,0 SAY SPACE(80)
                SET COLOR TO W+/R, W+/R
                STORE 'No record exists for feature number ' +;
                    MFEAT + ', try again ' TO ERROR
                @ 24,12 SAY ERROR
                DO DELAY
            ELSE
                USE DESCRIPT INDEX DESCRIPT.NDX
                GO TOP
                STORE RECN0() TO FIRST_REC
                GO BOTTOM
                STORE RECN0() TO LAST_REC
                FIND &MFEAT
                IF EOF() = .T. THEN
                    SET COLOR TO W/B, W/B
                    @ 24,0 SAY SPACE(80)
                    SET COLOR TO W+/R, W+/R
                    STORE 'No record exists for feature number ' +;
                        MFEAT + ', try again ' TO ERROR
                    @ 24,12 SAY ERROR
                    DO DELAY
                ELSE
                    USE DESCRIPT INDEX DESCRIPT.NDX
                    GO TOP
                    STORE RECN0() TO FIRST_REC
                    GO BOTTOM
                    STORE RECN0() TO LAST_REC
                    FIND &MFEAT
                    IF EOF() = .T. THEN
                        SET COLOR TO W/B, W/B
                        @ 24,0 SAY SPACE(80)
                        SET COLOR TO W+/R, W+/R
                        STORE 'No record exists for feature number ' +;
                            MFEAT + ', try again ' TO ERROR
                        @ 24,12 SAY ERROR
                        DO DELAY
                    ELSE
                        USE DESCRIPT INDEX DESCRIPT.NDX
                        GO TOP
                        STORE RECN0() TO FIRST_REC
                        GO BOTTOM
                        STORE RECN0() TO LAST_REC
                        FIND &MFEAT
                        IF EOF() = .T. THEN
                            SET COLOR TO W/B, W/B
                            @ 24,0 SAY SPACE(80)
                            SET COLOR TO W+/R, W+/R
                            STORE 'No record exists for feature number ' +;
                                MFEAT + ', try again ' TO ERROR
                            @ 24,12 SAY ERROR
                            DO DELAY
                        ELSE
                            USE DESCRIPT INDEX DESCRIPT.NDX
                            GO TOP
                            STORE RECN0() TO FIRST_REC
                            GO BOTTOM
                            STORE RECN0() TO LAST_REC
                            FIND &MFEAT
                            IF EOF() = .T. THEN
                                SET COLOR TO W/B, W/B
                                @ 24,0 SAY SPACE(80)
                                SET COLOR TO W+/R, W+/R
                                STORE 'No record exists for feature number ' +;
                                    MFEAT + ', try again ' TO ERROR
                                @ 24,12 SAY ERROR
                                DO DELAY
                            ELSE
                                USE DESCRIPT INDEX DESCRIPT.NDX
                                GO TOP
                                STORE RECN0() TO FIRST_REC
                                GO BOTTOM
                                STORE RECN0() TO LAST_REC
                                FIND &MFEAT
                                IF EOF() = .T. THEN
                                    SET COLOR TO W/B, W/B
                                    @ 24,0 SAY SPACE(80)
                                    SET COLOR TO W+/R, W+/R
                                    STORE 'No record exists for feature number ' +;
                                        MFEAT + ', try again ' TO ERROR
                                    @ 24,12 SAY ERROR
                                    DO DELAY
                                ELSE
                                    USE DESCRIPT INDEX DESCRIPT.NDX
                                    GO TOP
                                    STORE RECN0() TO FIRST_REC
                                    GO BOTTOM
                                    STORE RECN0() TO LAST_REC
                                    FIND &MFEAT
                                    IF EOF() = .T. THEN
                                        SET COLOR TO W/B, W/B
                                        @ 24,0 SAY SPACE(80)
                                        SET COLOR TO W+/R, W+/R
                                        STORE 'No record exists for feature number ' +;
                                            MFEAT + ', try again ' TO ERROR
                                        @ 24,12 SAY ERROR
                                        DO DELAY
                                    ELSE
                                        USE DESCRIPT INDEX DESCRIPT.NDX
                                        GO TOP
                                        STORE RECN0() TO FIRST_REC
                                        GO BOTTOM
                                        STORE RECN0() TO LAST_REC
                                        FIND &MFEAT
                                        IF EOF() = .T. THEN
                                            SET COLOR TO W/B, W/B
                                            @ 24,0 SAY SPACE(80)
                                            SET COLOR TO W+/R, W+/R
                                            STORE 'No record exists for feature number ' +;
                                                MFEAT + ', try again ' TO ERROR
                                            @ 24,12 SAY ERROR
                                            DO DELAY
                                        ELSE
                                            USE DESCRIPT INDEX DESCRIPT.NDX
                                            GO TOP
                                            STORE RECN0() TO FIRST_REC
                                            GO BOTTOM
                                            STORE RECN0() TO LAST_REC
                                            FIND &MFEAT
                                            IF EOF() = .T. THEN
                                                SET COLOR TO W/B, W/B
                                                @ 24,0 SAY SPACE(80)
                                                SET COLOR TO W+/R, W+/R
                                                STORE 'No record exists for feature number ' +;
                                                    MFEAT + ', try again ' TO ERROR
                                                @ 24,12 SAY ERROR
                                                DO DELAY
                                            ELSE
                                                USE DESCRIPT INDEX DESCRIPT.NDX
                                                GO TOP
                                                STORE RECN0() TO FIRST_REC
                                                GO BOTTOM
                                                STORE RECN0() TO LAST_REC
                                                FIND &MFEAT
                                                IF EOF() = .T. THEN
                                                    SET COLOR TO W/B, W/B
                                                    @ 24,0 SAY SPACE(80)
                                                    SET COLOR TO W+/R, W+/R
                                                    STORE 'No record exists for feature number ' +;
                                                        MFEAT + ', try again ' TO ERROR
                                                    @ 24,12 SAY ERROR
                                                    DO DELAY
                                                ELSE
                                                    USE DESCRIPT INDEX DESCRIPT.NDX
                                                    GO TOP
                                                    STORE RECN0() TO FIRST_REC
                                                    GO BOTTOM
                                                    STORE RECN0() TO LAST_REC
                                                    FIND &MFEAT
                                                    IF EOF() = .T. THEN
                                                        SET COLOR TO W/B, W/B
                                                        @ 24,0 SAY SPACE(80)
                                                        SET COLOR TO W+/R, W+/R
                                                        STORE 'No record exists for feature number ' +;
                                                            MFEAT + ', try again ' TO ERROR
                                                        @ 24,12 SAY ERROR
                                                        DO DELAY
                                                    ELSE
                                                        USE DESCRIPT INDEX DESCRIPT.NDX
                                                        GO TOP
                                                        STORE RECN0() TO FIRST_REC
                                                        GO BOTTOM
                                                        STORE RECN0() TO LAST_REC
                                                        FIND &MFEAT
                                                        IF EOF() = .T. THEN
                                                            SET COLOR TO W/B, W/B
                                                            @ 24,0 SAY SPACE(80)
                                                            SET COLOR TO W+/R, W+/R
                                                            STORE 'No record exists for feature number ' +;
                                                                MFEAT + ', try again ' TO ERROR
                                                            @ 24,12 SAY ERROR
                                                            DO DELAY
                                                        ELSE
                                                            USE DESCRIPT INDEX DESCRIPT.NDX
                                                            GO TOP
                                                            STORE RECN0() TO FIRST_REC
                                                            GO BOTTOM
                                                            STORE RECN0() TO LAST_REC
                                                            FIND &MFEAT
                                                            IF EOF() = .T. THEN
                                                                SET COLOR TO W/B, W/B
                                                                @ 24,0 SAY SPACE(80)
                                                                 
APPENDIX B: MAINTENANCE MANUAL

Page 3

DESPREV.PRG Program Listing

```pascal
101 SET COLOR TO /W, /W
102 @ 24,0 SAY MESSAGE
103 LOOP
104 ELSE
105 EXIT
106 ENDIF EOF() = .T.
107 ENDIF (MFEAT = '00' OR MFEAT = '99')
108 ENDIF
109 ENDDO WHILE .T.
110 *
111 SET COLOR TO W/B, W/B
112 @ 24,0 SAY SPACE (80)
113 @ 20,20 SAY "To view this field, enter the update mode."
114 DO WHILE .T.
115 SET COLOR TO R+/B, R+/B
116 @ 4,46 SAY RECNO() PICT "99999"
117 SET COLOR TO /BR, /BR
118 @ 6,45 SAY FEATURE() PICT "999999"
119 @ 8,45 SAY CLN PICT "999"
120 @ 10,45 SAY DESCRIPT PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
121 @ 12,45 SAY MODELNO PICT "" " 
122 @ 14,45 SAY FDCMODEL PICT "" " 
123 @ 16,45 SAY TYPECOMPON PICT ""
124 @ 18,45 SAY BASEMAINT PICT "9999.99"
125 SET COLOR TO R+/B, R+/B
126 STORE "N" TO CHOICE
127 @ 22,67 GET CHOICE PICT "!"
128 READ
129 *
130 * ENSURE THAT THE USER'S PROMPT IS EITHER "N", "P" OR "X"
131 *
132 DO WHILE .NOT. (CHOICE = "N" OR. CHOICE = "P" OR. CHOICE = "X")
133 IF .NOT. (CHOICE = "N" OR. CHOICE = "P" OR. CHOICE = "X") THEN
134 SET COLOR TO W+/R, W+/R
135 @ 24,23 SAY "Response must be either N, P or X"
136 DO DELAY
137 STORE "N" TO CHOICE
138 ENDIF
139 SET COLOR TO R+/B, R+/B
140 @ 22,67 GET CHOICE PICT "!"
141 READ
142 END
143 *
144 * SKIP TO THE NEXT RECORD TO BE REVIEWED
145 *
146 IF CHOICE = "N" THEN
147 IF RECNO() = LAST_REC THEN
148 GO TOP
149 ELSE
150 SKIP
```

258
151     ENDIF
152     ENDIF
153 
154 * TO THE PREVIOUS RECORD TO BE REVIEWED
155 *
156 IF CHOICE = "P" THEN
157 IF RECNO() = FIRST_REC THEN
158     GO BOTTOM
159 ELSE
160     SKIP -1
161 ENDIF
162 ENDIF
163 *
164 * USER HAS DECIDED TO EXIT THE REVIEW
165 *
166 IF CHOICE = "X"
167     EXIT
168 ENDIF
169 ENDDO WHILE .T.
170 *
171 * RETURN TO CALLING PROGRAM.
172 *
173 RELEASE ALL LIKE M*, CHOICE, ERROR, FIRST_REC, LAST_REC
174 CLOSE DATABASES
175 RETURN
176*******************************************************************************
PROCEDURE DESPPUPD.PRG

AUTHORS:
LCDR EDWARD J. CASE, SC, USN
LCDR WINSTON H. BUCKLEY, SC, USN
LCDR ROBERT F. BRADO, USN
LCDR ROBERT L. BEARD III, SC, USN

PURPOSE: TO ENABLE THE USER TO MODIFY ANY DATA ELEMENT IN THE DESCRIPTION DATABASE.

INPUT FILES: DESCRIP.DBF INDICES: DESCRIP.NDX

OUTPUT FILES: DESCRIP.DBF, INDICES: DESCRIP.NDX

MODULES CALLED: DELAY.PRG

CALLED BY: DESPPMOG.PRG

GLOBAL VARIABLE: LOFNUM, HIFNUM

LOCAL VARIABLES: ACCEPT, ANS, CHOICE, ERROR, INTRO, MBMAINT, MCLIN, MDESCRIP, MESSAGE, MFDCMODL, MFEAT, MMODELNO, MCOMP

DATE LAST TIME MODIFIED: 23 DECEMBER 1985

BEGIN
CASE SELECTION
= I
UPDATE EXISTING RECORDS

USE DESCRIPTION DATABASE USING FEATURE NUMBER INDEX.

SET ESCAPE OFF
SET TALK OFF
USE DESCRIP
GO TOP
SET COLOR TO W+/B, W+/B, B
CLEAR
IF EOF() = .T. THEN
SET COLOR TO W+/R, W+/R
@ 13,17 SAY "The EQUIPMENT DESCRIPTION Database is EMPTY!"
DO DELAY
RETURN
ENDIF
?? FLASH + "S.DESCRIPT.SCR"
@ 24,0 SAY SPACE(80)
@ 2,39 SAY "UPDATE"
STORE ' Enter 00 to start at TOF, 99 to start at EOF, or a six digit '+
'feature number') TO MESSAGE
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
DO WHILE .T.
APPENDIX B: MAINTENANCE MANUAL

Page 2

Page 2 DESPPUPD.PRG Program Listing

51 SET COLOR TO /BR, /BR
52 STORE '00 ' TO MFEAT
53 @ 6,45 GET MFEAT PICT '999999'
54 READ
55 IF .NOT. ((MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM) .OR.;
56 MFEAT = '00 ' .OR. MFEAT = '99 ')
57 SET COLOR TO W/B, W/B
58 @ 24,0 SAY SPACE(80)
59 SET COLOR TO W+/R, W+/R
60 STORE ' Response must be between '+ LOFNUM + ' and '+;
61 HIFNUM + ', Zero (00) or 99 ' TO ERROR
62 @ 24,8 SAY ERROR
63 DO DELAY
64 SET COLOR TO /W, /W
65 @ 24,0 SAY MESSAGE
66 LOOP
67 ELSE
68 IF (MFEAT = '00 ' .OR. MFEAT = '99 ' ) THEN
69 USE DESCRIP
70 IF MFEAT = '00 ' THEN
71 GO BOTTOM
72 STORE RECNO() TO LAST_REC
73 GO TOP
74 STORE RECNO() TO FIRST_REC
75 ELSE
76 IF MFEAT = '99 ' THEN
77 GO TOP
78 STORE RECNO() TO FIRST_REC
79 GO BOTTOM
80 STORE RECNO() TO LAST_REC
81 ENDIF MFEAT = '99 '
82 ENDIF MFEAT = '00 '
83 STORE FEATURENO TO MFEAT
84 EXIT
85 ELSE
86 USE DESCRIP INDEX DESCRIP.NDX
87 GO TOP
88 STORE RECNO() TO FIRST_REC
89 GO BOTTOM
90 STORE RECNO() TO LAST_REC
91 FIND MFEAT
92 IF EOF() = .T. THEN
93 SET COLOR TO W/B, W/B
94 @ 24,0 SAY SPACE(80)
95 SET COLOR TO W+/R, W+/R
96 STORE ' No record exists for feature number ' +;
97 MFEAT + ', try again ' TO ERROR
98 @ 24,12 SAY ERROR
99 DO DELAY
100 SET COLOR TO /W, /W

261
APPENDIX B: MAINTENANCE MANUAL

Page 125

DESPUPD.PRG Program Listing

101  @ 24,0 SAY MESSAGE
    LOOP
103    ELSE
104        EXIT
105    ENDIF EOF() = .T.
106    ENDIF (MFEAT = '00' .OR. MFEAT = '99')
107    ENDIF
108    ENDDO WHILE .T.
109 *
110    STORE SPACE(16) + 'Press "Page Down" key to terminate record update';
111    SPACE(16) TO MESSAGE
112    STORE 1 TO INTRO
113    DO WHILE .T.
114    *
115    * INFORM THE USER OF HOW TO TERMINATE THE UPDATE OF A RECORD
116    *
117    IF INTRO = 1 THEN
118        STORE 0 TO INTRO
119        ?? FLASH + "W.DESPPUPD/
120        SET CONSOLE OFF
121        WAIT TO ANS
122        SET CONSOLE ON
123    ENDIF
124    *
125    * STORING THE OLD RECORD TO A WORK RECORD AREA. THE M PREFIX
126    * INDICATES MEMORY VARIABLES DISTINGUISHING THEM FROM THEIR
127    * CORRESPONDING DATABASE FIELDS.
128    *
129    STORE FEATURENO TO MFEAT
130    STORE CLIN TO MCLIN
131    STORE DESCRPT TO MDESCRIP
132    STORE MODELNO TO MMODEU1O
133    STORE FDCMODEL TO MFDQDL
134    STORE TYPECOMPON TO MIMl'MP
135    STORE BASEMAINT TO MUMAINT
136    SET COLOR TO R+/B, R+/B
137    @ 4,46 SAY RECNO() PICT "99999"
138    SET COLOR TO /W, /W
139    @ 6,46 SAY MFEAT PICT "99999"
140    SET COLOR TO /BR, /BR
141    @ 8,45 GET MCLIN PICT "9999"
142    @ 10,45 GET MDESCRIP PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
143    @ 12,45 GET MMODELNO PICT "!!!!!!!!!!!!"
144    @ 14,45 GET MFDQDL PICT "!!!!!!!!!!!!!!"
145    @ 16,45 GET MIMCOMP PICT "!
146    @ 18,45 GET MUMAINT PICT "9999,99"
147    READ
148    *

262
APPENDIX B: MAINTENANCE MANUAL

Page 4 DESPPUPD.PRG Program Listing

151 | SET COLOR TO W/B, W/B
152 | @ 24,0 SAY SPACE(80)
153 |
154 | * ASK THE USER IF HE/SHE DESIRES TO ACCEPT THE CHANGES, ONLY IF ANY
155 | CHANGES WERE MADE
156 | *
157 | IF .NOT. (FEATURENO=MFEAT .AND. CLIN=MCLIN .AND. DESCIPT=MDESCRIP .AND. ;
158 | MODELNO=MMODELNO .AND. FDCMODEL=MFDCCMODL .AND. TYPECOMPON =;
159 | MTCOMP .AND. BASEMAINT=MBASEMAINT) THEN
160 | SET COLOR TO W+/ , W+/ R
161 | @ 21,10 SAY SPACE (55)
162 | @ 21,12 SAY "Do you want to accept the changes? (Yes or No):"
163 | SET COLOR TO R+/ , R+/ R
164 | @ 21,49 SAY "Y"
165 | @ 21,56 SAY "N"
166 | STORE "N" TO ACCEPT
167 | @ 21,62 GET ACCEPT PICT "!"
168 | READ
169 |
170 | * ENSURE THAT THE USER’S PROMPT IS EITHER "Y" OR "N"
171 |
172 | DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
173 | IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
174 | SET COLOR TO W/B, W/B
175 | @ 24,0 SAY SPACE(80)
176 | SET COLOR TO W+/R, W+/R
177 | @ 24,24 SAY "Response must be either N or Y"
178 | DO DELAY
179 | STORE "N" TO ACCEPT
180 | SET COLOR TO /W, /W
181 | @ 24,0 SAY MESSAGE
182 | ENDIF
183 | SET COLOR TO R+/ , R+/ R
184 | @ 21,62 GET ACCEPT PICT "!"
185 | READ
186 | ENDDO
187 | SET COLOR TO W+/B, W+/ B
188 | @ 21,10 SAY SPACE (60)
189 |
190 | * STORING THE CORRECTED EDIT FIELDS FROM THE WORK AREA.
191 |
192 | IF ACCEPT = "Y" THEN
193 | REPLACE FEATURENO WITH MFEAT
194 | REPLACE CLIN WITH MCLIN
195 | REPLACE DESCIPT WITH MDESCRIP
196 | REPLACE MODELNO WITH MMODELNO
197 | REPLACE FDCMODEL WITH MFDCCMODL
198 | REPLACE TYPECOMPON WITH MTCOMP
199 | REPLACE BASEMAINT WITH MBASEMAINT
200 | ENDIF
DESPPUPD.PRG Program Listing

201 | ENDIF
202 | * SET COLOR TO W/B, W/B
203 | @ 21,10 SAY SPACE (55)
204 | * ASK THE USER IF HE/SHE DESIRES TO CHANGE THE NOTES FIELD
205 | * SET COLOR TO W+/B, W+/B
206 | @ 20,18 SAY "Edit the NOTES field? (Yes or No):"
207 | SET COLOR TO R+/B, R+/B
208 | @ 20,42 SAY "Y"
209 | @ 20,49 SAY "N"
210 | STORE "N" TO ACCEPT
211 | @ 20,54 GET ACCEPT PICT ":"
212 | READ
213 | * ENSURE THAT THE USER'S PROMPT IS EITHER "Y" OR "N"
214 | DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
215 | IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
216 | SET COLOR TO W/B, W/B
217 | @ 24,0 SAY SPACE(80)
218 | SET COLOR TO W+/R, W+/R
219 | @ 24,24 SAY " Response must be either N or Y "
220 | DO DELAY
221 | STORE "N" TO ACCEPT
222 | SET COLOR TO /W, /W
223 | @ 24,0 SAY MESSAGE
224 | ENDIF
225 | SET COLOR TO R+/B, R+/B
226 | @ 20,54 GET ACCEPT PICT ":"
227 | READ
228 | ENDUO
229 | IF ACCEPT = "Y" THEN
230 | ?? FLASH + "W.NOTES/"
231 | SET CONSOLE OFF
232 | WAIT TO ANS
233 | SET CONSOLE ON
234 | CHANGE FIELDS NOTES
235 | CLEAR
236 | ?? FLASH + "S.DESCRIPT.SCR/"
237 | @ 24,0 SAY SPACE(80)
238 | SET COLOR TO W+/B, W+/B
239 | @ 2,39 SAY "UPDATE"
240 | SET COLOR TO R+/B, R+/B
241 | @ 4,46 SAY RECOM() PICT ":@99999"
242 | SET COLOR TO /HR, /HR
243 | @ 6,45 SAY MFHAT PICT ":@99999"
DESPUPD.PRG Program Listing

Page 6

APPENDIX B: MAINTENANCE MANUAL

Page 128

251  @ 8,45 SAY MCLIN PICT "9999"
252  @ 10,45 SAY MDESCRIP PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
253  @ 12,45 SAY MMODELNO PICT "!!!!!!!!!!!"
254  @ 14,45 SAY MDCMODEL PICT "!!!!!!!!!!!!!!"
255  @ 16,45 SAY MMCOMP PICT "!
256  @ 18,45 SAY MMAINT PICT "9999.99"
257  ENDF
258  *
259  SET COLOR TO W/B, W/B
260  @ 20,18 SAY SPACE (50)
261  SET COLOR TO R+/B, R+/B
262  STORE "N" TO CHOICE
263  @ 22,67 GET CHOICE PICT ":!"
264  READ
265  *
266  ENSURE THAT THE USER'S PROMPT IS EITHER "N", "P" OR "X"
267  *
268  DO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X")
269   IF .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X") THEN
270      SET COLOR TO W/B, W/B
271      @ 24,0 SAY SPACE(80)
272      SET COLOR TO R+/R, R+/R
273      @ 24,23 SAY " Response must be either N, P or X "
274      DO DELAY
275      STORE "N" TO CHOICE
276  ENDF
277  SET COLOR TO R+/B, R+/B
278  @ 22,67 GET CHOICE PICT ":!"
279  READ
280  ENDDO
281  *
282  SKIP TO THE NEXT RECORD TO BE REVIEWED
283  *
284  IF CHOICE = "N" THEN
285     IF RECNO () = LAST_REC THEN
286        GO TOP
287     ELSE
288        SKIP
289     ENDF
290  ENDF
291  *
292  SKIP TO THE PREVIOUS RECORD TO BE REVIEWED
293  *
294  IF CHOICE = "P" THEN
295     IF RECNO () = FIRST_REC THEN
296        GO BOTTOM
297     ELSE
298        SKIP -1
299     ENDF
300  ENDF
APPENDIX B: MAINTENANCE MANUAL

Page 7

DESPUPD.PRG Program Listing

301 | *
302 | * HAS DECIDED TO EXIT THE REVIEW
303 | *
304 | IF CHOICE = "X"
305 | EXIT
306 | ENDIF
307 | ENDDO WHILE .T.
308 | *
309 | * RETURN TO CALLING PROGRAM.
310 | *
311 | RELEASE ALL LIKE M*, ACCEPT, ANS, CHOICE, ERROR, INTRO
312 | CLOSE DATABASES
313 | RETURN
314 | *******************************
1 * PROCEDURE EQPDTNPC.PRG
2 *
3 * AUTHORS : LCDR EDWARD J. CASE, SC, USN
4 * LCDR WINSTON H. BUCKLEY, SC, USN
5 * LCDR ROBERT F. BRADO, USN
6 * LCDR ROBERT L. BEARD III, SC, USN
7 *
8 * PURPOSE : PROVIDE THE USER A SPLICE EQUIPMENT DELIVERY
9 * ORDER DATE LEVEL REPORT WITHOUT UNIT COSTS.
10 *
11 * INPUT FILES : EQUIP.DBF, EQUIPSD.NDX, DESCRIP.DBF, DESCRIP.NDX,
12 * EQUIPSIT.NDX
13 *
14 * OUTPUT FILES : TEMPONE.DBF, TEMPONE.NDX
15 *
16 * MODULES CALLED : DELAY.PRG
17 *
18 * GLOBAL VARIABLE: HIDATE, HISITE, LOCATE, LWSITE
19 *
20 * LOCAL VARIABLES: ACCEPT, CHOICE, COLCNT, ERROR, LINECT, MKEY,
21 * MNDMDATE, MOLDATE, MSITE, PAGENO, SYSDATE,
22 * TODAY, TODATE
23 *
24 * DATE LAST TIME MODIFIED = = = = = = = = = = = = = = 27 DECEMBER 1985 = = = = = = = = = = = = = = = = = = =}
25 *
26 * CASE SELECTION = 1 EQUIPMENT EFFECTIVE DELIVERY ORDER REPORT
27 * WITHOUT UNIT COST
28 *
29 * CALL EQUIPMENT DATABASE INDEXED ON SITE NUMBER. DISPLAY ALL.
30 * EFFECTIVE DATES OF DELIVERY ORDERS FOR THE USER TO SELECT FROM.
31 * CALL EQUIPMENT DATABASE INDEXED ON EFFECTIVE DELIVERY ORDER DATE
32 * AND SITE NUMBER. COPY APPLICABLE RECORDS TO TEMPONE, INDEXED ON
33 * FEATURE NUMBER. RELATE TO DESCRIPTION FILE.
34 *
35 SET ESCAPE OFF
36 SET TALK OFF
37 SET COLOR TO W+/B, W+/D, B
38 CLEAR
39 USE EQUIP
40 GO TOP
41 IF EOF() = .T. THEN
42 SET COLOR TO W+/R, W+/R
43 @ 13,24 SAY " THE EQUIPMENT Database is EMPTY! "
44 DO DELAY
45 RETURN
46 ENDIF
47 ?? FLASH + "S.REPORTS.SCR/"
48 @ 24,0 SAY SPACE(80)
49 SET COLOR TO R+/ , R+/n
50 @ 2,25 SAY " DELIVERY ORDER LEVEL REPORT "

267
SET COLOR TO W+/BR, W+/BR
@ 13,15 SAY "Enter site number for which the report is desired;"
*
* ENSURE THAT TEMPORARY DATABASE AND INDEX DO NOT EXIST, IF SO ERASE THEM
*
SET CONSOLE OFF
ERASE TEMPONE.DBF
ERASE TEMPONE.NDX
SET CONSOLE ON
USE EQUIP INDEX EQUIPSIT
*
DO WHILE .T.
SET COLOR TO /BR, /BR
STORE LOSITE TO MSITE
@ 13,66 GET MSITE PICT '99'
READ
IF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
SET COLOR TO W+/R, W+/R
STORE 'Response must be between ' + LOSITE +; ' + HISITE + ' TO ERROR
@ 24,22 SAY ERROR
DO DELAY
LOOP
ELSE
GO TOP
FIND &MSITE
IF EOF() = .T. THEN
STORE "No equipment exists for site " + MSITE +;
" try another site " TO MESSAGE
SET COLOR TO W+/R, W+/R
@ 24,15 SAY MESSAGE
DO DELAY
LOOP
ELSE
EXIT
ENDIF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE)
ENDO WHILE .T.
*
SET COLOR TO W+/BR, W+/BR
@ 13,15 SAY SPACE(60)
*
SET COLOR TO W+/B, W+/B
@ 05,09 SAY "The following Delivery Order Effective Dates exist for Site"
@ 05,60 SAY MSITE
SET COLOR TO /BR, /BR
@ 13,05 SAY SPACE(70)
STORE 1 TO LINECT
STORE 1.00 TO COUNT
STORE "000000" TO MOLDATE.
APPENDIX B: MAINTENANCE MANUAL

Program Listing

101* DO WHILE SITENO = MSITE
102  IF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00) THEN
103     @LINECT+6,57 SAY EFFDATE
104  ELSE
105     IF (COLCNT - (COLCNT * (COLCNT/2)) = 0.00) THEN
106         @LINECT+6,38 SAY EFFDATE
107     ELSE
108         @LINECT+6,19 SAY EFFDATE
109     ENDIF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00)
110     ENDIF (COLCNT - (COLCNT * (COLCNT/2)) = 0.00)
111     IF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00) THEN
112         LINECT = 1 + LINECT
113     ENDIF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00)
114     COLCNT = 1.00
115  ELSE
116     COLCNT = COLCNT + 1.00
117     ENDIF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00)
118     STORE EFFDATE TO MOLDATE
119  * DO WHILE ((EFFDATE = MOLDATE) .AND. .NOT. EOF())
120     SKIP+2
121  ENDDO
122  IF EOF() THEN
123     EXIT
124  ELSE
125     ENDIF EOF() = .T.
126  ENDDO WHILE SITENO = MSITE
127* STORE DLOC(DATE()) TO SYSDATE
128  STORE SUBSTR(SYSDATE,7,2) + SUBSTR(SYSDATE,1,2) *;
129  SUBSTR(SYSDATE,4,2) TO MDATE
130  STORE SPACE(17) + 'Input Effective Date (Range ' + MDATE + ' to ' + HILDATE + ')' + SPACE(17) TO MESSAGE
131  SET COLOR TO /W, /W
132  @ 24,0 SAY MESSAGE
133  SET COLOR TO W+/B, W+/B
134  @ 3,29 SAY "EFFECTIVE DATE: 
135* USE EQUIP INDEX EQUIPSD.NDX
136  STORE "000000" TO MOLDATE
137  DO WHILE .NOT. (MOLDATE >= LODATE .AND. MOLDATE <= HILDATE)
138      STORE MOLDATE TO MOLDATE
139      SET COLOR TO R+/B, R+/B
140      @ 3,45 GET MOLDATE PICT "9999999"
141      READ
142      DO WHILE .T.
143         IF .NOT. (SUBSTR(MOLDATE,1,2) > "83" .AND. ;
APPENDIX B: MAINTENANCE MANUAL

EQPDINPC.PRG Program Listing

151  SUBSTR(MOLDATE,1,2) <= "99") THEN
152       SET COLOR TO W/B, W/B
153       @ 24,0 SAY SPACE(80)
154       SET COLOR TO W+/R, W+/R
155       @ 24,16 SAY "Year portion of date must be between 84 and 99"
156       DO DELAY
157       SET COLOR TO /W, /W
158       @ 24,0 SAY MESSAGE
159       STORE SUBSTR(MDATE,1,2) TO MYEAR
160       SET COLOR TO R+/B, R+/B
161       @ 3,45 GET MYEAR PICT "99"
162       READ
163       STORE MYEAR + SUBSTR(MOLDATE,3,4) TO MOLDATE
164   ELSE
165       EXIT
166 ENDIF
167 ENDDO WHILE .T.
168 * DO WHILE .T.
169   IF .NOT. (SUBSTR(MOLDATE,3,2) > = "01" .AND.;
170       SUBSTR(MOLDATE,3,2) <= "12") THEN
171       SET COLOR TO W/B, W/B
172       @ 24,0 SAY SPACE(80)
173       SET COLOR TO W+/R, W+/R
174       @ 24,16 SAY "Month portion of date must be between 01 and 12"
175       DO DELAY
176       SET COLOR TO /W, /W
177       @ 24,0 SAY MESSAGE
178       STORE SUBSTR(MDATE,3,2) TO MMONTH
179       SET COLOR TO R+/B, R+/B
180       @ 3,47 GET MMONTH PICT "99"
181       READ
182       STORE SUBSTR(MOLDATE,1,2) + MMONTH +;
183       SUBSTR(MOLDATE,5,2) TO MOLDATE
184   ELSE
185       EXIT
186 ENDIF
187 ENDDO WHILE .T.
188 * DO WHILE .T.
189   IF ((SUBSTR(MOLDATE,3,2) = "04" .OR. SUBSTR(MOLDATE,3,2) = "06") .AND.;
190       SUBSTR(MOLDATE,3,2) = "09" .OR. SUBSTR(MOLDATE,3,2) = "11") .AND.;
191       .NOT. (SUBSTR(MOLDATE,5,2) <= "30") THEN
192       SET COLOR TO W/B, W/B
193       @ 24,0 SAY SPACE(80)
194       @ 24,16 SAY "Day portion of date must be between 01 and 30"
195       DO DELAY
196       SET COLOR TO /W, /W
197 270
APPENDIX B: MAINTENANCE MANUAL

Page 5

EQPITNPC.PRG Program Listing

201 \ @ 24,0 SAY MESSAGE
202 \ STORE SUBSTR(MDATE,5,2) TO MDAY
203 \ SET COLOR TO R+/B, R+B
204 \ @ 3,49 GET MDAY PICT "99"
205 \ READ
206 \ STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
207 \ LOOP
208 \ ELSE
209 \ *\n210 \ IF (SUBSTR(MOLDATE,3,2) = "02" .AND. .NOT.;
211 \ (SUBSTR(MOLDATE,5,2) >= "01" .AND.;
212 \ SUBSTR(MOLDATE,5,2) <= "28") THEN
213 \ SET COLOR TO W/B, W/B
214 \ @ 24,0 SAY SPACE(80)
215 \ SET COLOR TO W+/R, W+/R
216 \ @ 24,16 SAY "Day portion of date must be between 01 and 28"
217 \ DO DELAY
218 \ SET COLOR TO /W, /W
219 \ @ 24,0 SAY MESSAGE
220 \ STORE SUBSTR(MDATE,5,2) TO MDAY
221 \ SET COLOR TO R+/B, R+B
222 \ @ 3,49 GET MDAY PICT "99"
223 \ READ
224 \ STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
225 \ LOOP
226 \ ELSE
227 \ *\n228 \ IF (.NOT. (SUBSTR(MOLDATE,5,2) >= "01" .AND.;
229 \ SUBSTR(MOLDATE,5,2) <= "31") THEN
230 \ SET COLOR TO W/B, W/B
231 \ @ 24,0 SAY SPACE(80)
232 \ SET COLOR TO W+/R, W+/R
233 \ @ 24,16 SAY "Day portion of date must be between 01 and 31"
234 \ DO DELAY
235 \ SET COLOR TO /W, /W
236 \ @ 24,0 SAY MESSAGE
237 \ STORE SUBSTR(MDATE,5,2) TO MDAY
238 \ SET COLOR TO R+/B, R+B
239 \ @ 3,49 GET MDAY PICT "99"
240 \ READ
241 \ STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
242 \ LOOP
243 \ ELSE
244 \ EXIT
245 \ ENDF
246 \ ENDF
247 \ ENDF
248 \ ENDO WHILE .T.
249 \ *\n250 \ GO TOP
FIND &MOLDATE
IF EOF() = .T. THEN
    SET COLOR TO W/B, W/B
    @ 24,0 SAY SPACE(80)
    STORE "EFFECTIVE DATE " + MOLDATE + " does not exist for site " +
            MSITE + ", try another " TO NODATE
    SET COLOR TO W+/R, W+/R
    @ 24,04 SAY NODATE
    DO DELAY
    SET COLOR TO /W, /W
    @ 24,0 SAY MESSAGE
    STORE "000000" TO MOLDATE
    LOOP
ENDIF EOF() = .T.
ENDDO WHILE .NOT. (MOLDATE >= LODOATE .AND. MOLDATE <= HIDATE)
*
SET COLOR TO /BR, /BR
@ 07,2 SAY SPACE(76)
@ 08,2 SAY SPACE(76)
@ 09,2 SAY SPACE(76)
@ 10,2 SAY SPACE(76)
@ 11,2 SAY SPACE(76)
@ 12,2 SAY SPACE(76)
@ 13,2 SAY SPACE(76)
@ 14,2 SAY SPACE(76)
@ 15,2 SAY SPACE(76)
@ 16,2 SAY SPACE(76)
@ 17,2 SAY SPACE(76)
@ 18,2 SAY SPACE(76)
@ 19,2 SAY SPACE(76)
@ 20,2 SAY SPACE(76)
@ 21,2 SAY SPACE(76)
*
SET COLOR TO R+, R+
@ 13,18 SAY " CREATING TEMPORARY DATABASE AND INDEX FILE 
STORE "MOLDATE" + "MSITE" TO MKEY
GO TOP
FIND &MKEY
*
COPY TO TEMPONE FOR SITENO = "&MSITE" .AND. EFFDATE = "&MOLDATE"
SELECT 1
USE TEMPONE
INDEX ON FEATURENO TO TEMPONE
SELECT 2
USE DESCRIPT INDEX DESCRIPT
SELECT TEMPONE
SET RELATION TO FEATURENO INTO DESCRIPT
GO TOP
* CREATE THE SPLICE EQUIPMENT PROJECT REPORT AND CHECK IF THE REPORT
* IS TO BE PRINTED OR DISPLAYED ON THE SCREEN.

SET COLOR TO W+/BR, W+/BR
@ 13,15 SAY SPACE(60)
@ 13,16 SAY "Do you want a printed report? (Yes or No): "
SET COLOR TO /BR, /BR
@ 13,49 SAY "Y"
@ 13,56 SAY "N"
STORE "N" TO ACCEPT
@ 13,62 GET ACCEPT PICT ";"
READ
* ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
* DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
  IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
    SET COLOR TO W+/R, W+/R
    @ 24,24 SAY "Response must be either N or Y"
    DO DELAY
    STORE "N" TO ACCEPT
  ENDIF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
  SET COLOR TO /BR, /BR
  @ 13,62 GET ACCEPT PICT ";"
  READ
ENDDO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")

SET COLOR TO /BR, /BR
@ 13,15 SAY SPACE(55)
IF ACCEPT = "Y" THEN
  ?? FLASH "$W.PRINT"/
  SET CONSOLE OFF
  WAIT TO CHOICE
  SET CONSOLE ON
  SET COLOR TO W/B, W/B
  @ 22,10 SAY SPACE(65)
  STORE DATE(DATE()) TO TODAY
  STORE SUBSTR(TODAY,4,2) + " " + NUM2INT(DATE()) + " 19" ;
  SUBSTR(TODAY,7,2) TO TODAY
  STORE 0 TO PAGE NO
  STORE 61 TO LINE END
  SET (COLOR, LINE) 1+ , 1+
  SET DEVICE TO PRINT
*
DO WHILE .NOT. EOF()
    DO WHILE (LINECT <= 60 .AND. .NOT. EOF())
        @ LINECT,3 SAY SITENO PICT "99"
        @ LINECT,9 SAY B->CLIN PICT "9999"
        @ LINECT,17 SAY FEATURENO PICT "999999"
        @ LINECT,28 SAY B->DESCRIPT PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!"
        @ LINECT,60 SAY QTY PICT "999"
        LINECT = LINECT + 1
    ENDDO WHILE (LINECT <= 60 .AND. .NOT. EOF())

* IF EOF() = .T. THEN
  IF PAGENO > 1 THEN
    ENDF PAGENO > 1
  ENDIF PAGENO > 1
  EJECT
  SET DEVICE TO SCREEN
  @ 13,25 SAY "FINISHED PRINTING THE REPORT"
  DO DELAY
  EXIT
ELSIE
  SET DEVICE TO SCREEN
  @ 13,27 SAY "Printing Page Number " + STR(PAGENO) + 1,2,0) + ""
  SET DEVICE TO PRINT
  ENDF EOF() = .T.
* IF (LINECT > 60 .AND. PAGENO > 1) THEN
  @ 62,37 SAY "Page " + STR(PAGENO) + 2,0)
  ENDF (LINECT > 60 .AND. PAGENO > 1)
  @ 2,25 SAY "DELIVERY ORDER LEVEL REPORT"
  @ 3,29 SAY "EFFECTIVE DATE:"
  @ 3,45 SAY MOLDATE
  @ 4,60 SAY TODATE
  @ 6,2 SAY "SITE CLIN FEATURE# DESCRIPTION"
  @ 6,60 SAY "QTY MODEL NUMBER"
  @ 7,2 SAY "="
  @ 7,51 SAY "="
  @ PAGENO = PAGENO + 1
  STORE 0 TO LINECT
* ENDDO WHILE .NOT. EOF()
ELSE
  SET COLOR TO /GR+/B, /GR+/B
  @ 5,2 SAY "SITE CLIN FEATURE# DESCRIPTION"
  @ 5,60 SAY "QTY MODEL NUMBER"
  SET COLOR TO /BR, /BR
  STORE 0 TO LINECT
* ENDDO WHILE .NOT. EOF()
APPENDIX B: MAINTENANCE MANUAL

EQPDINPC.PRG Program Listing

Page 9

401 DO WHILE LNNECT < 15
402 @ LNNECT+7,3 SAY SITENO PICT "99"
403 @ LNNECT+7,9 SAY B->CLIN PICT "9999"
404 @ LNNECT+7,17 SAY FEATURENO PICT "999999"
405 @ LNNECT+7,28 SAY B->DESCRIPT PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!"
406 @ LNNECT+7,60 SAY QCY PICT "999"
407 @ LNNECT+7,67 SAY B->FDCMODEL PICT "!!!!!!!!!!"
408 LNNECT = LNNECT + 1
409 SKIP
410 IF EOF() = .T. THEN
411 SET COLOR TO W+/R, W+/R
412 @ 24,18 SAY " End of File reached, Press any key to EXIT ",
413 SET CONSOLE OFF
414 WAIT TO ACCEPT
415 SET CONSOLE ON
416 EXIT
417 ENDF EOF() = .T.
418 ENDDO WHILE LNNECT < 15
419 *
420 IF EOF() = .T. THEN
421 EXIT
422 ENDF EOF() = .T.
423 SET COLOR TO R+/B, R+/B
424 STORE "C" TO CHOICE
425 @ 22,57 GET CHOICE PICT ":"!
426 READ
427 *
428 ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
429 *
430 DO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
431 IF .NOT. (CHOICE = "C" .OR. CHOICE = "X") THEN
432 SET COLOR TO W+/R, W+/R
433 @ 24,24 SAY " Response must be either C or X ",
434 DO DELAY
435 STORE "C" TO CHOICE
436 ENDF .NOT. (CHOICE = "C" .OR. CHOICE = "X")
437 SET COLOR TO R+/B, R+/B
438 @ 22,57 GET CHOICE PICT ":"!
439 READ
440 ENDDO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
441 *
442 DETERMINE IF THE USER WANTS TO QUIT OR CONTINUE
443 *
444 IF CHOICE = "C"
445 SET COLOR TO /BR, /BR
446 @ 07,2 SAY SPACE(76)
447 @ 08,2 SAY SPACE(76)
448 @ 09,2 SAY SPACE(76)
449 @ 10,2 SAY SPACE(76)
450 @ 11,2 SAY SPACE(76)

275
EQPDINPC.PRG Program Listing

451 | @ 12,2 SAY SPACE(76)
452 | @ 13,2 SAY SPACE(76)
453 | @ 14,2 SAY SPACE(76)
454 | @ 15,2 SAY SPACE(76)
455 | @ 16,2 SAY SPACE(76)
456 | @ 17,2 SAY SPACE(76)
457 | @ 18,2 SAY SPACE(76)
458 | @ 19,2 SAY SPACE(76)
459 | @ 20,2 SAY SPACE(76)
460 | @ 21,2 SAY SPACE(76)
461 | STORE 0 TO LINECT
462 | ELSE
463 | EXIT
464 | ENDIF CHOICE = "C"
465 | *
466 | ENDWHILE NOT. EOF()
467 | *
468 | ENDIF ACCEPT = "Y"
469 | *
470 | * ERASE ALL TEMPORARY FILES CREATED DURING REPORT GENERATION
471 | *
472 | CLOSE DATABASES
473 | SET CONSOLE OFF
474 | ERASE TEMPOD.DBF
475 | ERASE TEMPOD.NDX
476 | SET CONSOLE ON
477 | SET PRINT OFF
478 | *
479 | * RETURN TO CALLING PROGRAM
480 | *
481 | RELEASE ALL LIKE M*, ACCEPT, CHOICE, COLCAT, ERROR, LINECT, PAGEEND,;
482 | SYSDATE, TODAY, TODATE
483 | RETURN
484 | **************************************************************************

***************
* PROCEDURE EQPDTPRC.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
  LCDR WINSTON H. BUCKLEY, SC, USN
  LCDR ROBERT F. BRADO, USN
  LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : PROVIDE THE USER A SPLICE EQUIPMENT DELIVERY ORDER DATE LEVEL REPORT WITH UNIT COSTS.

* INPUT FILES : EQUIP.DBF, EQUIPSD.NDX, DESCRIPT.DBF, DESCRIPT.NDX, EQUIPSIT.NDX

* OUTPUT FILES : TEMPONE.DBF, TEMPONE.NDX

* MODULES CALLED : DELAY.PRG

* GLOBAL VARIABLE: HIDATE, HSITE, LODATE, LOSITE

* LOCAL VARIABLES: ACCEPT, CHOICE, COLONT, ERROR, LINECT, MKKEY, MMENDATE, MODATE, MSITE, PAGENO, SYSDATE, TODAY, TODATE

** DATE LAST TIME MODIFIED ===========> 27 DECEMBER 1985 <==========

* CASE SELECTION = 1 EQUIPMENT EFFECTIVE DELIVERY ORDER REPORT WITH UNIT COST

CALL EQUIPMENT DATABASE INDEXED ON SITE NUMBER, DISPLAY ALL
* EFFECTIVE DATES OF DELIVERY ORDERS FOR THE USER TO SELECT FROM.
* CALL EQUIPMENT DATABASE INDEXED ON EFFECTIVE DELIVERY ORDER DATE AND SITE NUMBER, COPY APPLICABLE RECORDS TO TEMPONE, INDEXED ON FEATURE NUMBER. RELATE TO DESCRIPTION FILE.

SET ESCAPE OFF
SET TALK OFF
SET COLOR TO W/B, W+/B, B
CLEAR
USE EQUIP
GO TOP
IF EOF() = .T. THEN
  SET COLOR TO W+/R, W+/R
  @ 13,24 SAY "The EQUIPMENT Database is EMPTY!"
  DO DELAY
  RETURN
ENDIF
?? FLASH + "S.REPORTS.SCR/"
@ 24,0 SAY SPACE(80)
SET COLOR TO R+/R, R+/R
@ 2,25 SAY "DELIVERY ORDER LEVEL REPORT"
SET COLOR TO W+/BR, W+/BR

277
APPENDIX B: MAINTENANCE MANUAL

Page 2

EQPDTFRC.PRG Program Listing

51 @ 13,15 SAY "Enter site number for which the report is desired;"
52 *
53 * ENSURE THAT TEMPORARY DATABASE AND INDEX DO NOT EXIST, IF SO ERASE THEM
54 *
55 SET CONSOLE OFF
56 ERASE TEMPONE.DBF
57 ERASE TEMPONE.NDX
58 SET CONSOLE ON
59 USE EQUIP INDEX EQUIPSIT
60 *
61 DO WHILE .T.
62 SET COLOR TO /BR, /BR
63 STORE LOSITE TO MSITE
64 @ 13,66 GET MSITE PICT '99'
65 READ
66 IF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
67 SET COLOR TO W+/R, W+/R
68 STORE 'Response must be between ' + LOSITE +;
69 ' and ' + HISITE +'; TO ERROR
70 @ 24,22 SAY ERROR
71 DO DELAY
72 LOOP
73 ELSE
74 GO TOP
75 FIND &MSITE
76 IF EOF() = .T. THEN
77 STORE "No equipment exists for site " + MSITE +;
78 " try another site " TO MESSAGE
79 SET COLOR TO W+/R, W+/R
80 @ 24,15 SAY MESSAGE
81 DO DELAY
82 LOOP
83 ELSE
84 EXIT
85 ENDIF EOF() = .T.
86 ENDIF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE)
87 ENDWHILE .T.
88 *
89 SET COLOR TO W+/BR, W+/BR
90 @ 13,15 SAY SPACE(60)
91 *
92 SET COLOR TO W+/B, W+/B
93 @ 05,09 SAY "The following Delivery Order Effective Dates exist for Site"
94 @ 05,69 SAY MSITE
95 SET COLOR TO /HR, /BR
96 @ 13,05 SAY SPACE(70)
97 STORE 1 TO LINCNT
98 STORE 1.00 TO COUNNT
99 STORE "000000" TO MOLDATE
100 *
APPENDIX B: MAINTENANCE MANUAL

EQPUPPRC.PRG Program Listing

101| DO WHILE SITENO = MSITE
102| IF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00) THEN
103| @LINCT+6,57 SAY EFFDATE
104| ELSE
105| IF (COLCNT - (COLCNT * (COLCNT/2)) = 0.00) THEN
106| @LINCT+6,38 SAY EFFDATE
107| ELSE
108| @LINCT+6,19 SAY EFFDATE
109| ENDIF (COLCNT - (COLCNT * (COLCNT/2)) = 0.00)
110| ENDIF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00)
111| IF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00) THEN
112| LINCT = 1 + LINCT
113| COLCNT = 1.00
114| ELSE
115| COLCNT = COLCNT + 1.00
116| ENDIF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00)
117| STORE EFFDATE TO MOLDATE
118| * DO WHILE ((EFFDATE = MOLDATE) .AND. .NOT. EOF())
119| SKIP+2
120| ENDDO
121| * IF EOF() THEN
122| EXIT
123| ELSE
124| SKIP
125| ENDF EOF()
126| * USE EQUIP INDEX EQUIPS.NDX
127| DO WHILE .NOT. (MOLDATE >= LODATE .AND. MOLDATE <= HIDATE)
128| STORE MDATE TO MOLDATE
129| * STORE DOC(DATE()) TO SYSDATE
130| STORE SUBSTR(SYSDATE,7,2) + SUBSTR(SYSDATE,1,2) +
131| SUBSTR(SYSDATE,4,2) TO MDATE
132| STORE SPACE(17) + 'Input Effective Date (Range ' + LODATE +
133| ' to ' + HIDATE + ') ' + SPACE(17) TO MESSAGE
134| SET COLOR TO /W, /W
135| @ 24,0 SAY MESSAGE
136| SET COLOR TO W+/B, W+/B
137| @ 3,29 SAY "EFFECTIVE DATE: "
138| * USE EQUIP INDEX EQUIPS.NDX
139| STORE "000000" TO MOLDATE
140| * DO WHILE .NOT. (MOLDATE >= LODATE .AND. MOLDATE <= HIDATE)
141| STORE MDATE TO MOLDATE
142| * SET COLOR TO R+/B, R+/B
143| * @ 3,45 GET MOLDATE PICT "999999"
144| READ
145| * DO WHILE .T.
146| IF .NOT. (SUBSTR(MOLDATE,1,2) > "83" .AND.
147| SUBSTR(MOLDATE,1,2) <= "99") THEN

279
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
@ 24,16 SAY " Year portion of date must be between 84 and 99 ">
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
STORE SUBSTR(MDATE,1,2) TO MYEAR
SET COLOR TO R+/B, R+/B
@ 3,45 GET MYEAR PICT "99"
READ
STORE MYEAR + SUBSTR(MOLDATE,3,4) TO MOLDATE
ELSE
EXIT
ENDIF
ENDDO
* DO WHILE .T.
IF .NOT. (SUBSTR(MOLDATE,3,2) > "00" .AND.;
SUBSTR(MOLDATE,3,2) < "13") THEN
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
@ 24,16 SAY " Month portion of date must be between 01 and 12 ">
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
STORE SUBSTR(MDATE,3,2) TO MMONTH
SET COLOR TO R+/B, R+/B
@ 3,47 GET MMONTH PICT "99"
READ
STORE SUBSTR(MOLDATE,1,2) + MMONTH *;
SUBSTR(MOLDATE,5,2) TO MOLDATE
ELSE
EXIT
ENDIF
ENDDO
* DO WHILE .T.
IF ((SUBSTR(MOLDATE,3,2) = "04" .OR. SUBSTR(MOLDATE,3,2) = "06" .OR.;
SUBSTR(MOLDATE,3,2) = "09" .OR. SUBSTR(MOLDATE,3,2) = "11") .AND.;
.NOT. (SUBSTR(MOLDATE,5,2) = "01" .AND.;
SUBSTR(MOLDATE,5,2) <= "30")) THEN
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
@ 24,16 SAY " Day portion of date must be between 01 and 30 ">
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
APPENDIX B: MAINTENANCE MANUAL

Page 144

EQPDTFRC.PRG Program Listing

201 STORE SUBSTR(MDATE,5,2) TO MDAY
202 SET COLOR TO R+/B, R+B
203 @ 3,49 GET MDAY PICT "99"
204 READ
205 STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
206 LOOP
207 ELSE
208 *
209 IF (SUBSTR(MOLDATE,3,2) = "02" .AND. .NOT.;
210 (SUBSTR(MOLDATE,5,2) >= "01" .AND.;
211 SUBSTR(MOLDATE,5,2) <= "28") THEN
212 SET COLOR TO W/B, W/B
213 @ 24,0 SAY SPACE(80)
214 SET COLOR TO W+/R, W+/R
215 @ 24,16 SAY " Day portion of date must be between 01 and 28 "
216 DO DELAY
217 SET COLOR TO /W; /W
218 @ 24,0 SAY MESSAGE
219 STORE SUBSTR(MDATE,5,2) TO MDAY
220 SET COLOR TO R+/B, R+B
221 @ 3,49 GET MDAY PICT "99"
222 READ
223 STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
224 LOOP
225 ELSE
226 *
227 IF .NOT. (SUBSTR(MOLDATE,5,2) >= "01" .AND.;
228 (SUBSTR(MOLDATE,5,2) <= "31") THEN
229 SET COLOR TO W/B, W/B
230 @ 24,0 SAY SPACE(80)
231 SET COLOR TO W+/R, W+/R
232 @ 24,16 SAY " Day portion of date must be between 01 and 31 "
233 DO DELAY
234 SET COLOR TO /W, /W
235 @ 24,0 SAY MESSAGE
236 STORE SUBSTR(MDATE,5,2) TO MDAY
237 SET COLOR TO R+/B, R+B
238 @ 3,49 GET MDAY PICT "99"
239 READ
240 STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
241 LOOP
242 ELSE
243 EXIT
244 ENDIF
245 ENDIF
246 ENDIF
247 ENDDO WHILE .T.
248 *
249 GO TOP
250 FIND &MOLDATE
APPENDIX B: MAINTENANCE MANUAL

EQPDTPRC.PRG Program Listing

251 IF EOF() = .T. THEN
252 SET COLOR TO W/B, W/B
253 @ 24,0 SAY SPACE(80)
254 STORE " EFFECTIVE DATE " + MOLDATE + " does not exist for site " + MSITE + ", try another " TO NODATE
255 SET COLOR TO W+/R, W+/R
256 @ 24,0 SAY NODATE
257 DO DELAY
258 SET COLOR TO /W, /W
259 @ 24,0 SAY MESSAGE
260 STORE "000000" TO MOLDATE
261 LOOP
262 ELSE
263 EXIT
264 ENDIF EOF() = .T.
265 ENDDO WHILE .NOT. (MOLDATE >= LODATE .AND. MOLDATE <= HIDATE)
266 *
267 SET COLOR TO W+/B, W+/B
268 @ 05,0 SAY SPACE(70)
269 @ 24,0 SAY SPACE(80)
270 *
271 * CLEAR LISTING OF EFFECTIVE DATES FROM SCREEN
272 *
273 SET COLOR TO /BR, /BR
274 @ 07,2 SAY SPACE(76)
275 @ 08,2 SAY SPACE(76)
276 @ 09,2 SAY SPACE(76)
277 @ 10,2 SAY SPACE(76)
278 @ 11,2 SAY SPACE(76)
279 @ 12,2 SAY SPACE(76)
280 @ 13,2 SAY SPACE(76)
281 @ 14,2 SAY SPACE(76)
282 @ 15,2 SAY SPACE(76)
283 @ 16,2 SAY SPACE(76)
284 @ 17,2 SAY SPACE(76)
285 @ 18,2 SAY SPACE(76)
286 @ 19,2 SAY SPACE(76)
287 @ 20,2 SAY SPACE(76)
288 @ 21,2 SAY SPACE(76)
289 *
290 SET COLOR TO R+/R, R+/R
291 @ 13,18 SAY " CREATING TEMPORARY DATABASE AND INDEX FILE "
292 STORE "MOLDATE" + "MSITE" TO MKEY
293 GOTO TOP
294 FIND MKEY
295 *
296 COPY TO TEMPORE FOR SITE = "MSITE", .AND. EFFECT = "MOLDATE"
297 SELECT 1
298 USE TEMPORE
299 INDEX (C1 FEATUREN) TO TEMPORE

282
APPENDIX B: MAINTENANCE MANUAL

Page 7

EQPDTPRC.PRG Program Listing

301 SELECT 2
302 USE DESCRIPT INDEX DESCRIPT
303 SELECT TEMPONE
304 SET RELATION TO FEATURENO INTO DESCRIPT
305 GO TOP
306 *
307 * CREATE THE SPLICE EQUIPMENT PROJECT REPORT AND CHECK IF THE REPORT
308 * IS TO BE PRINTED OR DISPLAYED ON THE SCREEN.
309 *
310 SET COLOR TO W+/BR, W+/BR
311 @ 13,15 SAY SPACE(60)
312 @ 13,16 SAY " Do you want a printed report? (Yes or No): "
313 SET COLOR TO /BR, /BR
314 @ 13,49 SAY "Y"
315 @ 13,56 SAY "N"
316 STORE "N" TO ACCEPT
317 @ 13,62 GET ACCEPT PICT ";"
318 READ
319 *
320 * ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
321 *
322 DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
323 IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
324 SET COLOR TO W+/R, W+/R
325 @ 24,24 SAY " Response must be either N or Y "
326 DO DELAY
327 STORE "N" TO ACCEPT
328 ENDIF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
329 SET COLOR TO /BR, /BR
330 @ 13,62 GET ACCEPT PICT ";"
331 READ
332 ENDDO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
333 *
334 SET COLOR TO /BR, /BR
335 @ 13,15 SAY SPACE(55)
336 *
337 IF ACCEPT = "Y" THEN
338 ?? FLASH + "W.PRINTR/"
339 SET CONSOLE OFF
340 WAIT TO CHOICE
341 SET CONSOLE ON
342 SET COLOR TO W/B, W/B
343 @ 22,10 SAY SPACE(65)
344 STORE DTOC(DATE()) TO TODAY
345 STORE SUBSTR(TODAY,4,2) + " " + SUBSTR(DATE(),2) TO TODATE
346 STORE 0 TO PAGENO
347 STORE 61 TO LINES
348 SET COLOR TO R+/ , R+
349 SET DEVICE TO PRINT

283
APPENDIX B: MAINTENANCE MANUAL

Page 8

EQPDTPRC.FRG Program Listing

351. * DO WHILE .NOT. EOF()
352. DO WHILE (LINECT <= 60 .AND. .NOT. EOF())
353. @ LINECT,3 SAY SITENO PICI "99"
354. @ LINECT,9 SAY B->CLIN PIC "9999"
355. @ LINECT,17 SAY FEATURENO PIC "999999"
356. @ LINECT,28 SAY B->DESCRIPTION PIC "!!!!!!!!!!!!!!!!!!!!!!"
357. @ LINECT,60 SAY UNIT PRICE PIC "99999999.99"
358. LINECT = LINECT + 1
359. SKIP
360. ENDW
361. * ENDDO WHILE (LINECT <= 60 .AND. .NOT. EOF())
362. * IF EOF() = .T. THEN
363. IF PAGENO > 1 THEN
364. @ 62,37 SAY "Page " + STR(PAGENO,2,0)
365. ENDIF PAGENO > 1
366. EJECT
367. SET DEVICE TO SCREEN
368. @ 13,25 SAY "FINISHED PRINTING THE REPORT"
369. DO DELAY
370. EXIT
371. ELSE
372. SET DEVICE TO SCREEN
373. @ 13,27 SAY "Printing Page Number " + STR(PAGENO + 1,2,0) + " "
374. SET DEVICE TO PRINT
375. ENDIF EOF() = .T.
376. * IF (LINECT > 60 .AND. PAGENO > 1) THEN
377. @ 62,37 SAY "Page " + STR(PAGENO,2,0)
378. ENDIF (LINECT > 60 .AND. PAGENO > 1)
379. @ 2,25 SAY "DELIVERY ORDER LEVEL REPORT"
380. @ 3,29 SAY "EFFECTIVE DATE:"
381. @ 3,45 SAY MOLDATE
382. @ 8,60 SAY TODYATE
383. @ 6,2 SAY "SITE CLIN. FEATURE# DESCRIPTION"
384. @ 6,60 SAY "QTY UNIT PRICE"
385. @ 7,2 SAY "=====================================================================
386. @ 7,51 SAY "=====================================================================
387. @ 7,51 SAY "=====================================================================
388. @ 7,2 SAY "=====================================================================
389. @ 7,51 SAY "=====================================================================
390. PAGENO = PAGENO + 1
391. STORE 9 TO LINECT
392. * ENDDO WHILE .NOT. EOF()
393. ELSE
394. SET COLOR TO GR+/B, GR+/B
395. @ 5,2 SAY "SITE CLIN. FEATURE# DESCRIPTION"
396. @ 5,60 SAY "QTY UNIT PRICE"
397. SET COLOR TO /UR, /UR
398. STORE 0 TO LINECT
399. *
APPENDIX B: MAINTENANCE MANUAL

EQPDTPRC.PRG Program Listing

401 DO WHILE .NOT. EOF()
402 DO WHILE LINCT < 15
403 @ LINCT+7,3 SAY SITENO PICT "99"
404 @ LINCT+7,9 SAY B->CLIN PICT "9999"
405 @ LINCT+7,17 SAY FEATURENO PICT "999999"
406 @ LINCT+7,28 SAY B->DESCRIP PICT "!!!!!!!!!!!!!!!!!!!!!!!!!"
407 @ LINCT+7,60 SAY QTY PICT "99"
408 @ LINCT+7,66 SAY UNIT_PRICE PICT "99999999.99"
409 LINCT = LINCT + 1
410 SKIP
411 IF EOF() = .T. THEN
412 SET COLOR TO W+/R, W+/R
413 @ 24,18 SAY " End of File reached, Press any key to EXIT "
414 SET CONSOLE OFF
415 WAIT TO ACCEPT
416 SET CONSOLE ON
417 EXIT
418 ENDIF EOF() = .T.
419 ENDDO WHILE LINEC < 15
420 IF EOF() = .T. THEN
421 EXIT
422 ENDF EOF() = .T.
423 SET COLOR TO R+/B, R+/B
424 STORE "C" TO CHOICE
425 22,57 GET CHOICE PICT ":\"
426 READ
427 ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
428 IF EOF() = .T. THEN
429 EXIT
430 ENDF EOF() = .T.
431 SET COLOR TO W+/R, W+/R
432 @ 24,24 SAY " Response must be either C or X "
433 DO DELAY
434 STORE "C" TO CHOICE
435 ENDF .NOT. (CHOICE = "C" .OR. CHOICE = "X")
436 SET COLOR TO R+/B, R+/B
437 22,57 GET CHOICE PICT ":\"
438 READ
439 ENDWHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
440 * DETERMINE IF THE USER WANTS TO QUIT OR CONTINUE
441 * IF CHOICE = "C"
442 SET COLOR TO /BR, /BR
443 @ 07,2 SAY SPACE(76)
444 @ 08,2 SAY SPACE(76)
445 *
EQPDTPRC.PRG Program Listing

451  @ 11,2 SAY SPACE(76)
452  @ 12,2 SAY SPACE(76)
453  @ 13,2 SAY SPACE(76)
454  @ 14,2 SAY SPACE(76)
455  @ 15,2 SAY SPACE(76)
456  @ 16,2 SAY SPACE(76)
457  @ 17,2 SAY SPACE(76)
458  @ 18,2 SAY SPACE(76)
459  @ 19,2 SAY SPACE(76)
460  @ 20,2 SAY SPACE(76)
461  @ 21,2 SAY SPACE(76)
462  STORE 0 TO LINECT
463  ELSE
464  EXIT
465  ENDIF CHOICE = "c"
466  *
467  ENDDO WHILE .NOT. EOF()
468  *
469  ENDIF ACCEPT = "y"
470  *
471  * ERASE ALL TEMPORARY FILES CREATED DURING REPORT GENERATION
472  *
473  CLOSE DATABASES
474  SET CONSOLE OFF
475  ERASE TEMPONE.DBF
476  ERASE TEMPONE.NDX
477  SET CONSOLE ON
478  SET PRINT OFF
479  *
480  * RETURN TO CALLING PROGRAM
481  *
482  RELEASE ALL LIKE M*, ACCEPT, CHOICE, COLCNT, ERROR, LINECT, PAGEND,;
483  SYSDATE, TODAY, TODATE
484  RETURN
485  ********************************************

286
PROCEDURE EQPRJRT.PRG

AUTHORS:
- LCDR EDWARD J. CASE, SC, USN
- LCDR WINSTON H. BUCKLEY, SC, USN
- LCDR ROBERT F. BRADO, USN
- LCDR ROBERT L. BEARD III, SC, USN

PURPOSE: PROVIDE THE USER A SPLICE EQUIPMENT PROJECT LEVEL REPORT.

INPUT FILES: EQUIP.DBF, DESCRIP.DBF, DESCRIPT.NDX, TEMPODNE.DBF, EFEAT.NDX

OUTPUT FILE: TEMPODNE.DBF

CALLED BY: PROJRPTS.PRG

MODULES CALLED: DELAY.PRG

LOCAL VARIABLES: ACCEPT, CHOICE, LINEC', PAGENO, TODAY, TODATE

DATE LAST TIME MODIFIED: 27 DECEMBER 1985

CASE SELECTION = 1  EQUIPMENT PROJECT LEVEL REPORT

CALL EQUIPMENT DATABASE INDEXED ON CONTRACT LINE NUMBER AND FEATURE

NUMBER AND TOTAL ON QUANTITY. RELATE TO DESCRIP FILE ON FEATURENO.

SET ESCAPE OFF
SET TALK OFF
SET COLOR TO W+/B, W+/B, W+/R
CLEAR
USE EQUIP
GO TOP
IF EOF() = .T. THEN
   SET COLOR TO W+/R, W+/R
   \ 13,24 SAY " The EQUIPMENT Database is EMPTY! "
   DO DELAY
   RETURN
ENDIF
?? FLASH + "S.REPORTS.SER/
\ 24,9 SAY SPACE(80)
<table>
<thead>
<tr>
<th>1.0</th>
<th>1.1</th>
<th>1.25</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.4</td>
<td>2.0</td>
<td>1.6</td>
</tr>
</tbody>
</table>
EQPPJRPT.PRG Program Listing

51 SET COLOR TO /BR, /BR
52 @ 13,49 SAY "Y"
53 @ 13,56 SAY "N"
54 STORE "N" TO ACCEPT
55 @ 13,62 GET ACCEPT PICT "!"
56 READ
57 *
58 ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
59 *
60 DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
61 IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
62 SET COLOR TO W+/R, W+/R
63 @ 24,24 SAY "Response must be either N or Y"
64 DO DELAY
65 STORE "N" TO ACCEPT
66 ENDIF
67 SET COLOR TO /BR, /BR
68 @ 13,62 GET ACCEPT PICT "!"
69 READ
70 ENDDO
71 *
72 SET COLOR TO /BR, /BR
73 @ 13,15 SAY SPACE(55)
74 *
75 SET COLOR TO W+/BR, W+/BR
76 @ 13,19 SAY "COMPUTING TOTALS FOR EACH FEATURE NUMBER"
77 *
78 USE EQUIP INDEX EFEAT
79 GO TOP
80 SET CONSOLE OFF
81 ERASE TEMPONE.DBF
82 SET CONSOLE ON
83 *
84 COMPUTE THE TOTAL QUANTITY FOR EACH FEATURE NUMBER
85 *
86 TOTAL ON FEATURENO TO TEMPONE.DBF FIELDS QTY WHILE FEATURENO = 'XXXXXXX'
87 *
88 SELECT 1
89 USE TEMPONE
90 SELECT 2
91 USE DESCRIP INDEX DESCRIP
92 SELECT TEMPONE
93 SET RELATION TO FEATURENO INTO DESCRIP
94 GO TOP
95 *
96 @ 13,15 SAY SPACE(55)
97 *
98 IF ACCEPT = "Y" THEN
99 ?? FLASH "W.PRINTER/"
100 SET CONSOLE OFF
APPENDIX B: MAINTENANCE MANUAL

EQPPJRPT.PRG Program Listing

101 | WAIT TO CHOICE
102 | SET CONSOLE ON
103 | SET COLOR TO W/B, W/B
104 | @ 22,10 SAY SPACE(65)
105 | STORE DTOC(DATE()) TO TODAY
106 | STORE SUBSTR(TODAY,4,2) + " " + MONTH(DATE()) + " 19" + ;
107 | SUBSTR(TODAY,7,2) TO TODATE
108 | STORE 0 TO PAGENO
109 | STORE 61 TO LINECT
110 | SET COLOR TO R+/ , R+/  
111 | SET DEVICE TO PRINT
112 | * DO WHILE .NOT. EOF()
113 | DO WHILE (LINECT <= 60 .AND. .NOT. EOF())
114 | @ LINECT,10 SAY DESCRIPT*CLIN
115 | @ LINECT,22 SAY FEATURENO
116 | @ LINECT,35 SAY DESCRIPT*DESCIP
117 | @ LINECT,68 SAY QTY
118 | LINECT = LINECT + 1
119 | SKIP
120 | ENDDO WHILE
121 | * IF EOF() = .T. THEN
122 | IF PAGENO > 1 THEN
123 | @ 62,37 SAY "Page " + STR(PAGENO,2,0)
124 | ENDIF
125 | EJECT
126 | SET DEVICE TO SCREEN
127 | @ 13,25 SAY "FINISHED PRINTING THE REPORT"
128 | DO DELAY
129 | EXIT
130 | ELSE
131 | SET DEVICE TO SCREEN
132 | @ 13,27 SAY "Printing Page Number " + STR(PAGENO + 1,2,0) + ""
133 | SET DEVICE TO PRINT
134 | ENDF
135 | * IF (LINECT > 60 .AND. PAGENO > 1) THEN
136 | @ 62,37 SAY "Page " + STR(PAGENO,2,0)
137 | ENDIF
138 | @ 2,25 SAY "EQUIPMENT PROJECT LEVEL REPORT"
139 | @ 4,60 SAY TODATE
140 | @ 6,10 SAY "CLIN FEATURE# DESCRIPTION"
141 | @ 6,68 SAY "QTY"
142 | @ 7,2 SAY "=================================================================
143 | @ 7,51 SAY "=================================================================
144 | PAGENO = PAGENO + 1
145 | STORE 9 TO LINECT
146 | * ENDDO WHILE .NOT. EOF()
APPENDIX B: MAINTENANCE MANUAL

Page 4

EQPPJRPT.PRG Program Listing

151 *
152 ELSE
153   SET COLOR TO GR+/B, GR+/B
154   @ 5,10 SAY "CLIN FEATURE# DESCRIPTION"
155   @ 5,68 SAY "QTY"
156   SET COLOR TO /BR, /BR
157   STORE 0 TO LINECT
158 *
159   DO WHILE .NOT. EOF()
160     DO WHILE LINECT < 15
161       @ LINECT+7,10 SAY DESCRIP->CLIN
162       @ LINECT+7,22 SAY FEATURENO
163       @ LINECT+7,35 SAY DESCRIP->DESCRIPTION
164       @ LINECT+7,68 SAY QTY
165       LINECT = LINECT + 1
166       SKIP
167       IF EOF() = .T. THEN
168         SET COLOR TO W+/R, W+/R
169         @ 24,18 SAY " End of File reached, Press any key to EXIT "
170         SET CONSOLE OFF
171         WAIT TO ACCEPT
172         SET CONSOLE ON
173         EXIT
174     ENDIF
175   ENDDO WHILE LINECT < 15
176 *
177     IF EOF() = .T. THEN
178       EXIT
179     ENDIF
180     SET COLOR TO R+/B, R+/B
181     STORE "C" TO CHOICE
182     @ 22,57 GET CHOICE PICT ":"" 
183     READ
184 *
185     ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
186 *
187     DO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
188       IF .NOT. (CHOICE = "C" .OR. CHOICE = "X") THEN
189         SET COLOR TO W+/R, W+/R
190         @ 24,24 SAY " Response must be either C or X "
191         DO DELAY
192         STORE "C" TO CHOICE
193     ENDIF
194     SET COLOR TO R+/B, R+/B
195     @ 22,57 GET CHOICE PICT ":"" 
196     READ
197     ENDDO
198 *
199     DETERMINE IF THE USER WANTS TO QUIT OR CONTINUE
200 *
EQPJRPT.PRG Program Listing

IF CHOICE = "C"
   SET COLOR TO /BR, /BR
   @ 07,2 SAY SPACE(76)
   @ 08,2 SAY SPACE(76)
   @ 09,2 SAY SPACE(76)
   @ 10,2 SAY SPACE(76)
   @ 11,2 SAY SPACE(76)
   @ 12,2 SAY SPACE(76)
   @ 13,2 SAY SPACE(76)
   @ 14,2 SAY SPACE(76)
   @ 15,2 SAY SPACE(76)
   @ 16,2 SAY SPACE(76)
   @ 17,2 SAY SPACE(76)
   @ 18,2 SAY SPACE(76)
   @ 19,2 SAY SPACE(76)
   @ 20,2 SAY SPACE(76)
   @ 21,2 SAY SPACE(76)
   STORE 0 TO LINECT
   ELSE
      EXIT
      ENDIF
      *  ENDDO WHILE .NOT. EOF()
      *  ENDF
      *  ERASE TEMPORARY DATABASE USED FOR TOTALS
      *  CLOSE DATABASES
      SET CONSOLE OFF
      ERASE TEMPONE.DBF
      SET CONSOLE ON
      SET PRINT OFF
      *  RETURN TO CALLING PROGRAM
      *  RELEASE ACCEPT, CHOICE, LINECT, PAGENO, TODAY, TUDATE
      RETURN

********************
* PROCEDURE EQPSTRPT.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
           LCDR WINSTON H. BUCKLEY, SC, USN
           LCDR ROBERT F. BRADO, USN
           LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : PROVIDE THE USER A SPLICE EQUIPMENT SITE
            LEVEL REPORT FOR A SINGLE SITE.

* INPUT FILES : EQUIP.DBF, EFEAT.NDX, DESCRIP.DBF, DESCRIP.NDX,
                TEMPONE.DBF, EQUIPSIT.NDX

* OUTPUT FILES : NONE.

* CALLED BY : SITERPTS.PRG

* MODULES CALLED : DELAY.PRG

* GLOBAL VARIABLE: HISITE, LOSITE

* LOCAL VARIABLES: ACCEPT, QIOLCE, ERROR, LINEC, MESSAGE, mSLIL,
                   PAGENO, TODAY, TODATE

* DATE LAST TIME MODIFIED = 27 DECEMBER 1985

* CASE SELECTION = 1 EQUIPMENT SITE LEVEL REPORT

* CALL EQUIPMENT DATABASE INDEXED ON SITE NUMBER, CONTRACT LINE NUMBER
  AND FEATURE NUMBER AND TOTAL ON QUANTITY.

SET ESCAPE OFF
SET TALK OFF
SET COLOR TO W+/B, W+/B, B
CLEAR
USE EQUIP
GO TOP
IF EOF() = .T. THEN
  SET COLOR TO W+/R, W+/R
  ' 13,24 SAY " The EQUIPMENT database is EMPTY! "
  DO DELAY
  RETURN
ENDIF
?? FLASH + "S.REPORTS..SCR/"
4 24,0 SAY SPACE(99)
SET COLOR TO R+/ , R+/'
4 2,26 SAY " EQUIPMENT SITE LEVEL REPORT "

* ENSURE THAT TEMPORARY DATABASE DOES NOT EXIST, IF SO ERASE IT
SET CONSOLE OFF
52 ERASE TEMPONE.DBF
53 SET CONSOLE ON
54 *
55 SET COLOR TO W+/BR, W+/BR
56 @ 13,15 SAY "Enter site number for which the report is desired:"
57 *
58 DO WHILE .T.
59 SET COLOR TO /BR, /BR
60 STORE LOSITE TO MSITE
61 @ 13,66 GET MSITE PICT '99'
62 READ
63 IF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE)
64 SET COLOR TO W+/R, W+/R
65 STORE  'Response must be between ' + LOSITE + ;
66   'and ' + HISITE + ' TO ERROR
67 @ 24,22 SAY ERROR
68 DO DELAY
69 LOOP
70 ELSE
71 USE EQUIP INDEX EQUIPSIT
72 DO TOP
73 FIND &MSITE
74 IF EOF() = .T. THEN
75 STORE "No equipment exists for site " + MSITE + ;
76 "try another site" TO MESSAGE
77 SET COLOR TO W+/R, W+/R
78 @ 24,15 SAY MESSAGE
79 DO DELAY
80 LOOP
81 ELSE
82 EXIT
83 ENDIF EOF() = .T.
84 ENDIF
85 ENDDO WHILE .T.
86 *
87 SET COLOR TO W+/BR, W+/BR
88 @ 13,15 SAY SPACE(55)
89 *
90 * CREATE THE SPLICE EQUIPMENT PROJECT REPORT AND CHECK IF THE REPORT
91 * IS TO BE PRINTED OR DISPLAYED ON THE SCREEN.
92 *
93 @ 13,16 SAY " Do you want a printed report? (Yes or No): "
94 SET COLOR TO /BR, /BR
95 @ 13,49 SAY "y"
96 @ 13,56 SAY "n"
97 STORE "n" TO ACCEPT
98 @ 13,62 GET ACCEPT PICT ":"
99 READ
100 *
ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"

DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
  IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
    SET COLOR TO W+/R, W+/R
    @ 24,24 SAY " Response must be either N or Y "
  DO DELAY
  STORE "N" TO ACCEPT
  ENDF
  SET COLOR TO /BR, /BR
  @ 13,62 GET ACCEPT PICT ":"
  READ
  ENDDO
  SET COLOR TO /BR, /BR
  @ 13,62 SAY " COMPTING TOTALS FOR EACH SITE FEATURE NUMBER "
  *
  USE EQUIP INDEX EFEAT
  TOTAL ON FEATURENO TO TEMPONE.DBF FIELDS QT;
  FOR FEATURENO <> 'XXXXXX' .AND. SITENO = '&MSITE'
  SELECT 1
  USE TEMPONE
  SELECT 2
  USE DESCRIP INDEX DESCRIP
  SELECT TEMPONE
  SET RELATION TO FEATURENO INTO DESCRIP
  GO TOP
  *
  @ 13,15 SAY SPACE(55)
  *
  IF ACCEPT = "Y" THEN
    ?? FLASH + "W.PRINTER."
    SET CONSOLE OFF
    WAIT TO CHOICE
    SET CONSOLE ON
    SET COLOR TO W/B, W/B
    @ 22,10 SAY SPACE(65)
    STORE DTOD(DATE(1)) TO TODAY
    STORE SUBSTR(TODAY,4,2) + " " + MONTH(DATE(1)) + " " +
    SUBSTR(TODAY,7,2) TO TODATE
    STORE 0 TO PAGE20
    STORE 61 TO LINECT
    SET COLOR TO R+/R, R+/R
    SET DEVICE TO PRINT
    *
    DO WHILE .NOT. EOF()
    DO WHILE (LINECT <= 60 .AND. NOT. EOF())
      @ LINECT, 9 SAY SITE20
    ENDWHILE
APPENDIX B: MAINTENANCE MANUAL

EQPSTRPT.PRG Program Listing

@ LINECT, 17 SAY DESCRIP-CLIN
@ LINECT, 27 SAY FEATURENO
@ LINECT, 39 SAY DESCRIP-DESCRIPT
@ LINECT, 71 SAY QTY
LINECT = LINECT + 1
SKIP
ENDO WHILE (LINECT <= 60 .AND. .NOT. EOF())

* IF EOF() = .T. THEN
   IF PAGENO > 1 THEN
      @ 62, 37 SAY "Page " + STR(PAGENO, 2, 0)
      EJECT
      PAGENO > 1
      EJECT
   SET DEVICE TO SCREEN
   @ 13, 25 SAY "FINISHED PRINTING THE REPORT"
   DO DELAY
   EXIT
   ELSE
      SET DEVICE TO SCREEN
      @ 13, 27 SAY "Printing Page Number " + STR(PAGENO + 1, 2, 0) + ""
      SET DEVICE TO PRINT
      ENDIF EOF() = .T.

* IF (LINECT > 60 .AND. PAGENO > 1) THEN
   @ 62, 37 SAY "Page " + STR(PAGENO, 2, 0)
   ENDIF (LINECT > 60 .AND. PAGENO > 1)
   @ 2, 25 SAY "EQUIPMENT SITE LEVEL REPORT"
   @ 4, 60 SAY TOTDATE
   @ 6, 8 SAY "SITE CLIN FEATURE# DESCRIPTION"
   @ 6, 71 SAY "QTY"
   @ 7, 2 SAY "-----------------------------------------------"
   @ 7, 51 SAY "---------------------------------------------"
   PAGENO = PAGENO + 1
   STORE 9 TO LINECT

* ENDDO WHILE .NOT. EOF()

ELSE
   SET COLOR TO GR+R, GR+R
   @ 5, 8 SAY "SITE CLIN FEATURE# DESCRIPTION"
   @ 5, 71 SAY "QTY"
   SET COLOR TO /BR, /BR
   STORE 0 TO LINECT

* DO WHILE .NOT. EOF()
   DO WHILE LINECT < 15
      @ LINECT + 1, 9 SAY SITE
      @ LINECT + 7, 17 SAY DESCRIP-CLIN
      @ LINECT + 7, 21 SAY FEATURENO
      @ LINECT + 7, 39 SAY DESCRIP-DESCRIPT
APPENDIX B: MAINTENANCE MANUAL

EQPSIRPT.PRG Program Listing

201  @ LINCT+7,71 SAY QTY
202  LINCT = LINCT + 1
203  SKIP
204  IF EOF() = .T. THEN
205     SET COLOR TO W+/R, W+/R
206     @ 24,18 SAY " End of File reached, Press any key to EXIT "
207     SET CONSOLE OFF
208     WAIT TO ACCEPT
209     SET CONSOLE ON
210     EXIT
211  ENDF EOF() = .T.
212  ENDDO WHILE LINCT < 15
213  *
214  IF EOF() = .T. THEN
215     EXIT
216  ENDF EOF() = .T.
217  SET COLOR TO R+/B, R+/B
218  STORE "C" TO CHOICE
219  @ 22,57 GET CHOICE PICT ":1"
220  READ
221  *
222  ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
223  *
224  DO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
225     IF .NOT. (CHOICE = "C" .OR. CHOICE = "X") THEN
226         SET COLOR TO W+/R, W+/R
227         @ 24,24 SAY " Response must be either C or X "
228         DO DELAY
229         STORE "C" TO CHOICE
230     ENDF .NOT. (CHOICE = "C" .OR. CHOICE = "X")
231     SET COLOR TO R+/B, R+/B
232     @ 22,57 GET CHOICE PICT ":1"
233     READ
234  ENDDO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
235  *
236  DETERMINE IF THE USER WANTS TO QUIT OR CONTINUE
237  *
238  IF CHOICE = "C"
239      SET COLOR TO / BR, / BR
240      @ 07,2 SAY SPACE(76)
241      @ 08,2 SAY SPACE(76)
242      @ 09,2 SAY SPACE(76)
243      @ 10,2 SAY SPACE(76)
244      @ 11,2 SAY SPACE(76)
245      @ 12,2 SAY SPACE(76)
246      @ 13,2 SAY SPACE(76)
247      @ 14,2 SAY SPACE(76)
248      @ 15,2 SAY SPACE(76)
249      @ 16,2 SAY SPACE(76)
250      @ 17,2 SAY SPACE(76)
296
APPENDIX B: MAINTENANCE MANUAL

EQPSTRPT.PRG Program Listing

251  @ 18,2 SAY SPACE(76)
252  @ 19,2 SAY SPACE(76)
253  @ 20,2 SAY SPACE(76)
254  @ 21,2 SAY SPACE(76)
255  STORE 0 TO LINECT
256  ELSE
257     EXIT
258     ENDF IF CHOICE = "C"
259  *
260     ENDDO WHILE .NOT. EOF()
261  *
262     ENDF IF ACCEPT = "Y"
263  *
264     * ERASE THE TEMPORARY DATABASE USED FOR TOTALS
265  *
266  CLOSE DATABASES
267  SET CONSOLE OFF
268  ERASE TEMPONE.DBF
269  SET CONSOLE ON
270  SET PRINT OFF
271  *
272  * RETURN TO CALLING PROGRAM
273  *
274  RELEASE ACCEPT, CHOICE, ERROR, LINECT, MESSAGE, NSITE, PAGEOF, ;
275     TODAY, UUIDATE
276  RETURN
277  ************************************************************************************
PROCEDURE EQUIPM.D.PRG

AUTHORS

LCDR EDWARD J. CASE, SC, USN

LCDR WINSTON H. BUCKLEY, SC, USN

LCDR ROBERT F. BRADO, USN

LCDR ROBERT L. BEARD III, SC, USN

PURPOSE

PROVIDE THE USER THE OPPORTUNITY TO MODIFY OR REVIEW
ALL DATA IN THE EQUIPMENT DATABASE.

INPUT FILES

NONE

OUTPUT FILE

NONE

MODULES CALLED

EQUIPUPD.PRG, EQUIPREV.PRG

CALLED BY

MAINMENU.CMD

LOCAL VARIABLES

SELECT

DATE LAST TIME MODIFIED

22 DECEMBER 1985

DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE
SELECTION.

STORE "1" TO SELECT

DO WHILE SELECT = "1"

CLEAR

INSERT "SELECT NUMBER"

SET OPTIONS = 1

WAIT TO SELECT

CLEAR

PROCESS PROGCE BASED ON THE USER'S SELECTION.

DO CASE

CASE SELECT = "1"

CALL THE EQUIPM.D.PRG

DO EQUIPM.D

CASE SELECT = "2"

CALL THE EQUIPM.D.REV.PRG

DO EQUIPREV

RETURN TO THE MAIN MENU PROGRAM.

CASE SELECT = "3"

ENDCASE
ENDDO (WHILE SELEKT = "3")
* RETURN TO THE CALLING PROGRAM
* RETURN
*
APPENDIX B: MAINTENANCE MANUAL  Page 163

EQUIPREV.PRG Program Listing

1 | * PROCEDURE EQUIPREV.PRG
2 | *
3 | * AUTHORS : LCDR EDWARD J. CASE, SC, USN
4 | *
5 | * LCDR WINSTON H. BUCKLEY, SC, USN
6 | *
7 | * LCDR ROBERT F. BRADO, SC, USN
8 | *
9 | * PURPOSE : TO ENABLE THE USER TO REVIEW ALL RECORDS IN THE
10 | * EQUIPMENT DATABASE.
11 | *
12 | * INPUT FILES : EQUIP.DBF INDEX EQUIPSIT.NDX
13 | *
14 | * OUTPUT FILES : NONE
15 | *
16 | * CALLED BY : EQUIPCMD.PRG
17 | *
18 | * MODULES CALLED : DELAY.PRG
19 | *
20 | * GLOBAL VARIABLE: HIDATE, HIFNUM, HISITE, LODATE, LOFNUM, LOSITE
21 | *
22 | * LOCAL VARIABLES: CURRENTNO, EOF, ERROR, FIRST_REC, LAST_REC, MCLIN,
23 | * MDESCRIPT, MESSAGE, MSITE, MFLAT, TOF
24 | *
25 | * DATE LAST TIME MODIFIED =========> 23 DECEMBER 1985 =========>
26 | *
27 | * CASE SELECTION = 2 REVIEW EQUIPMENT FILE RECORDS
28 | *
29 | * USE EQUIPMENT DATABASE INDEXED ON SITE NUMBER AND WAIT FOR THE
30 | * USER TO INPUT THE SITE NUMBER, THEN START REVIEWING FROM THAT POINT.
31 | *
32 | SET ESCAPE OFF
33 | SET TALK OFF
34 | SELECT 1
35 | USE EQUIP
36 | GO TOP
37 | CLEAR
38 | IF EOF() = .T. THEN
39 | SET COLOR TO W+/B, W+/B, B
40 | CLEAR
41 | IF EOF() = .T. THEN
42 | SET COLOR TO W+/R, W+/R
43 | 13,24 SAY "The EQUIPMENT Database is EMPTY!"
44 | DO DELAY
45 | RETURN
46 | ENDIF
47 | ?? FLASH + "S.EQUIPREV.SCR/"
48 | ?? 24,0 SAY SPACE (80)
49 | STORE "Enter 00 to start at 'TOF, 99 to start at EOF, or a site number" +
50 | "between 01 and 58 " TO MESSAGE
51 | SET COLOR TO /W, /W
52 | ?? 24,0 SAY MESSAGE
53 | STORE '88' TO MSITE
EQUIPREV.PRG Program Listing

DO WHILE .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
SET COLOR TO /BR, /BR
STORE '00' TO MSITE
@ 9,20 GET MSITE PICT '99'
READ
IF .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W/R, W/R
STORE 'Response must be between ' + LOSITE + ' and ' + HISITE + ', Zero (00) or 99' TO ERROR
@ 24,13 SAY ERROR
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
LOOP
ELSE
IF (MSITE = '00' .OR. MSITE = '99') THEN
USE EQUIP
IF MSITE = '00' THEN
GO BOTTOM
STORE RECNO() TO LAST_REC
GO TOP
STORE RECNO() TO FIRST_REC
ELSE
IF MSITE = '99' THEN
GO TOP
STORE RECNO() TO FIRST_REC
GO BOTTOM
STORE RECNO() TO LAST_REC
ENDIF MSITE = '99'
ENDIF MSITE = '00'
ELSE
USE EQUIP INDEX EQUIPSIT, EQUIPPRFJ, EQUIPPDA, EQUIPSD
GO TOP
FIND &MSITE
IF EOF() = .T. THEN
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
STORE "No records exist for site number " + MSITE + ", try again" TO ERROR
SET COLOR TO W/R, W/R
@ 24,16 SAY ERROR
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
STORE '88' TO MSITE
ENDIF
ENDIF
END
I.

APPENDIX B: MAINTENANCE MANUAL

Page 3

EQUIPREV.PRG Program Listing

101 ENDDO WHILE
102 *
103 STORE SPACE(10) + 'Enter "00 " to start at TOF or a six digit ' +;
104 'feature number'' + SPACE(10) TO MESSAGE
105 IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
106 SET COLOR TO /W, /W
107 @ 24,0 SAY MESSAGE
108 DO WHILE .T.
109 SET COLOR TO /BR, /BR
110 STORE '00 ' TO MFEAT
111 @ 14,45 GET MFEAT PICT '999999'
112 READ
113 IF .NOT. ((MFEAT >= LOPNUM .AND. MFEAT <= HIFNUM) .OR.;
114 MFEAT = '00 ' .OR. MFEAT = '99')
115 SET COLOR TO W/B, W/B
116 @ 24,0 SAY SPACE(80)
117 SET COLOR TO W/R, W/R
118 STORE ' Response must be between ' + LOPNUM + ' and ' +;
119 HIFNUM + ', Zero (00) or 99 ' TO ERROR
120 @ 24,8 SAY ERROR
121 DO DELAY
122 SET COLOR TO /W, /W
123 @ 24,0 SAY MESSAGE
124 LOOP
125 ELSE
126 IF MFEAT = '00 ' THEN
127 EXIT
128 ENDIF
129 IF (MFEAT >= LOPNUM .AND. MFEAT <= HIFNUM)
130 STORE MSITE * MFEAT TO MKEY
131 USE EQUIP INDEX EQUIFDAT
132 GO TOP
133 FIND &MKEY
134 IF EOF() = .T. THEN
135 SET COLOR TO W/B, W/B
136 @ 24,0 SAY SPACE(80)
137 SET COLOR TO W/R, W/R
138 @ 24,12 SAY ' No record exists for feature number ' +;
139 MFEAT + ', try again'
140 DO DELAY
141 SET COLOR TO /W, /W
142 @ 24,0 SAY MESSAGE
143 LOOP
144 ELSE
145 EXIT
146 ENDIF EOF() = .T.
147 ENDIF
148 ENDIF (MFEAT >= LOPNUM .AND. MFEAT <= HIFNUM)
149 ENDDO WHILE
* SET COLOR TO W/B, W/B
24,0 SAY SPACE(80)
STORE "At beginning of records for site number " + MSITE + " TO TOF
DO WHILE .T.
SET COLOR TO R+/B, R+/B
@ 6,47 SAY RECNO() PICT "999"
STORE FEATURENO 110 MFEAT'
SELECT 2
USE DESCRIP INDEX DESCRIP
FIND &MFEAT
STORE CLIN TO MCLIN
STORE DESCRIPT TO MDESCRIP
SELECT 1
SET COLOR /B, /B
@ 9,20 SAY SITENO PICT "99"
@ 9,68 SAY EFFDATE PICT "999999"
@ 13,45 SAY MCLIN PICT "9999"
@ 14,45 SAY FEATURENO PICT "999999"
@ 15,45 SAY MDESCRIP PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!"
@ 16,45 SAY QTY PICT "999"
@ 18,50 SAY UNIT PRICE PICT "99999999.99"
@ 19,50 SAY MAINT PICT "99999999.99"
@ 20,53 SAY UNIT_INSTA PICT "999999.99"
SET COLOR /R, /R
@ 22,68 GET CHOICE PICT ";"
READ
* ENSURE THAT THE USER'S PROMPT IS EITHER "N", "P" OR "X"
DO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X")
IF .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X") THEN
SET COLOR TO /W, /W
@ 24,23 SAY " Response must be either N, P or X "
DO DELAY
STORE "N" TO CHOICE
ENDIF
SET COLOR TO /R, /R
@ 22,68 GET CHOICE PICT ";"
READ
ENDD
* SKIP TO THE NEXT RECORD TO BE REVIEWED
IF CHOICE = "N" THEN
IF (MSITE =: LMSITE .AND. MSITE <= HMSITE) THEN
SKIP
201 IF EOF() = .T. THEN
202 SKIP - 1
203 SET COLOR TO W+/R, W+/R
204 @ 24,21 SAY EOF
205 DO DELAY
206 ELSE
207 IF .NOT. (SITENO = MSITE) THEN
208 SKIP - 1
209 SET COLOR TO W+/R, W+/R
210 @ 24,21 SAY EOF
211 DO DELAY
212 ENDIF
213 ENDIF EOF() = .T.
214 ELSE
215 IF RECNO() = LAST REC THEN
216 GO TOP
217 ELSE
218 SKIP
219 ENDIF
220 ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE)
221 ENDIF CHOICE = "N"
222 *
223 * SKIP TO THE PREVIOUS RECORD
224 *
225 IF CHOICE = "P" THEN
226 STORE RECNO() TO CURRENTNO
227 IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
228 SKIP - 1
229 IF EOF() = .T. THEN
230 GOTO CURRENTNO
231 SET COLOR TO W+/R, W+/R
232 @ 24,16 SAY TOP
233 DO DELAY
234 ELSE
235 IF .NOT. (SITENO = MSITE) THEN
236 SKIP
237 SET COLOR TO W+/R, W+/R
238 @ 24,16 SAY TOP
239 DO DELAY
240 ENDIF
241 ENDIF EOF() = .T.
242 ELSE
243 IF RECNO() = FIRST REC THEN
244 GO BOTTOM
245 ELSE
246 SKIP - 1
247 ENDIF
248 ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE)
249 ENDIF CHOICE = "P"
250 *
APPENDIX B: MAINTENANCE MANUAL

EQUIPREV.FRG Program Listing

251  * USER HAS DECIDED TO EXIT THE REVIEW
252  * IF CHOICE = "X"
253     EXIT
254   ENDIF
255  ENDDO WHILE .T.
257  *
258  * RETURN TO CALLING PROGRAM.
259  *
260  RELEASE ALL LIKE M*, CURRENTNO, EOF, ERROR, FIRST_REC, LAST_REC, TUF
261  CLOSE DATABASES
262  RETURN
263  ***********************************************
* PROCEDURE EQUIPUPD.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
           : LCDR WINSTON H. BUCKLEY, SC, USN
           : LCDR ROBERT F. BRADO, SC, USN
           : LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : TO ENABLE THE USER TO MODIFY ANY DATA ELEMENT IN
            THE EQUIPMENT DATABASE.

* INPUT FILES : EQUIP.DBF, INDICES: EQUIPPRJ.NDX, EQUIPSIT.NDX
                EQUIPDAT.NDX, EQUIPSD.NDX

* OUTPUT FILES : EQUIP.DBF, INDICES: EQUIPPRJ.NDX, EQUIPSIT.NDX
                 EQUIPDAT.NDX, EQUIPSD.NDX

* CALLED BY : EQUIPCMD.PRG

* MODULES CALLED : DELAY.PRG

* GLOBAL VARIABLE: HIDATE, HIFNUM, HISITE, LODATE, LOFNUM, LWSITE

* LOCAL VARIABLES: MEFFDATE, MSITE, MEITE, MFEAT, MPRICE,
                   MAINT, MINSTALL, MQTY, MESSAGE
                   ACCEPT, CHOICE, CURRENTNO, EDF, ERROR, FIRST_REC,
                   INTRO, LAST_REC, TOF

* DATE LAST TIME MODIFIED = 23 DECEMBER 1985

* CASE SELECTION = 1  UPDATE EXISTING RECORDS

* USE EQUIPMENT DATABASE USING THE SITE NUMBER INDEX, BUT UPDATING
* ALL EQUIP FILE RELATED INDICES, ASK THE USER TO INPUT A SITE
* NUMBER THEN START UPDATING FROM THAT POINT.

SET ESCAPE OFF
SET TALK OFF
USE EQUIP
GO TOP
SET COLOR TO W*/B, W*/B, B
CLEAR
IF DEF() = .T. THEN
   SET COLOR TO W*/R, W*/R
   13,24 SAY "The EQUIPMENT Database is EMPTY!"
DO DELAY
RETURN
ENDIF
?? FLASH * "S.EQUIPUPD.SCR"
?? 24,0 SAY SPACE(80)
STORE "Enter 00 to start at TOF, 99 to start at DDF, or a site " +;
EQUIPUD.PRG Program Listing

51 "number between " + LOSITE + " and " + HISITE + "" TO MESSAGE
52 SET COLOR TO /W, /W
53 @ 24,0 SAY MESSAGE
54 STORE '88' TO MSITE
55 DO WHILE .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
56 SET COLOR TO /BR, /BR
57 STORE '00' TO MSITE
58 @ 8,20 GET MSITE PICT '99'
59 READ
60 IF .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
61 SET COLOR TO W/B, W/B
62 @ 24,0 SAY SPACE(80)
63 SET COLOR TO W+/R, W+/R
64 STORE 'Response must be between ' + LOSITE + ' and ' + HISITE + ';
65 Zero (00) or 99 ' TO ERROR
66 @ 24,13 SAY ERROR
67 DO DELAY
68 SET COLOR TO /W, /W
69 @ 24,0 SAY MESSAGE
70 LOOP
71 ELSE
72 IF (MSITE = '00' .OR. MSITE = '99') THEN
73 USE EQUIP
74 IF MSITE = '00' THEN
75 GO BOTTOM
76 STORE RECNO() TO LAST_REC
77 GO TOP
78 STORE RECNO() TO FIRST_REC
79 ELSE
80 IF MSITE = '99' THEN
81 GO TOP
82 STORE RECNO() TO FIRST_REC
83 GO BOTTOM
84 STORE RECNO() TO LAST_REC
85 ENDIF MSITE = '99'
86 ENDIF MSITE = '00'
87 ELSE
88 USE EQUIP INDEX EQUIPSIT, EQUIPPRJ, EQUIPVAT, EQUIPSD
89 GO TOP
90 FIND MSITE
91 IF EOF() = .T. THEN
92 SET COLOR TO W/B, W/B
93 @ 24,0 SAY SPACE(80)
94 STORE "No records exist for site number " + MSITE + " ", try again " TO ERROR
95 SET COLOR TO W+/R, W+/R
96 @ 24,16 SAY ERROR
97 DO DELAY
98 SET COLOR TO /W, /W
99 @ 24,0 SAY MESSAGE
APPENDIX B: MAINTENANCE MANUAL

Page 3

EQUIPUPD.PRG Program Listing

STORE '88' TO MSITE
ENDIF
ENDIF
ENDIF
ENDDO WHILE .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
STORE SPACE(10) + 'Enter "00 " to start at TOF or a six digit ' +;
'feature number' + SPACE(10) TO MESSAGE
IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
DO WHILE .T.
SET COLOR TO /BR, /BR
STORE '00 ' TO MFEAT
@ 11,45 GET MFEAT PICT '999999'
READ
IF .NOT. ((MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM) .OR.;
MFEAT = '00 ' .OR. MFEAT = '99 ')
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
STORE ' Response must be between ' + LOFNUM + ' and ' +;
'HIFNUM + ', Zero (00) or 99 ' TO ERROR
@ 24,8 SAY ERROR
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
LOOP
ELSE
IF MFEAT = '00 ' THEN
EXIT
ENDIF
IF (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
STORE MSITE + MFEAT TO MKEY
USE EQUIP INDEX EQUIPDATA
GO TOP
FIND &MKEY
IF EOF() = .T. THEN
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
STORE ' No record exists for feature number ' +;
MFEAT + ', try again ' TO ERROR
@ 24,12 SAY ERROR
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
EXIT
ELSE
EXIT

308
ENDIF EOF() = .T.
ENDIF (MFEAT => LOWNUM .AND. MFEAT <= HIFNUM)
ENDIF
END DO WHILE .T.
ENDIF (MSITE => LOSITE .OR. MSITE <= HISITE)
* SET COLOR TO W/B, W/B
= 24,0 SAY SPACE(80)
STORE " At beginning of records for site number " +
MSITE + " TO TOF
STORE " At end of records for site number " + MSITE + " TO EOF
STORE SPACE(16) + 'Press "Page Down" key to terminate record update' +
SPACE(16) TO MESSAGE
STORE 1 TO INTRO
DO WHILE .T.
SET COLOR TO /W, /W
24,0 SAY MESSAGE
* STORING THE OLD RECORD TO A WORK RECORD AREA. THE M PREFIX
* INDICATES MEMORY VARIABLES DISTINGUISHING THEM FROM THEIR
* CORRESPONDING DATABASE FIELDS.
* STORE UNIT PRICE TO MPRICE
STORE UNIT_INSTA TO MMINSTA
STORE UNIT_QTY TO MQTY
STORE FEATURENO TO MFEAT
SELECT 2
USE DESCRIPT INDEX DESCRIPT
FIND AMFEAT
STORE DESCRIPT TO MDESCRIP
SELECT 1
* INFORM THE USER OF HOW TO TERMINATE THE UPDATE OF A RECORD
* IF INTRO = 1 THEN
STORE 0 TO INTRO
?? FLASH + "W.EQUIPUPD/"
SET CONSOLE OFF
WAIT TO ANS
SET Console ON
ENDIF
* STORE COLOR TO R+/B, R+/B
4,47 SAY RECORD() PICT "999"
SET COLOR TO /BR, /BR
8,20 SAY STMNO PICT "09"
8,8 SAY EFFDATE PICT "999999"
11,45 SAY MFEAT PICT "999999"
12,45 SAY MDESCRIP PICT ""
APPENDIX B: MAINTENANCE MANUAL

Page 5

EQUIPUFD.PRG Program Listing

013,45 SAY MQTY PICT "999"
015,50 GET MPRICE PICT "99999999.99"
016,50 GET MMAINT PICT "99999999.99"
017,53 GET MINSTALL PICT "99999.99"
READ
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
* IF .NOT. (MQTY=MQTY .AND. UNIT PRICE=MPRICE .AND.
NO_MAINT=MMAINT .AND. UNIT INSTA=MINSTALL) THEN
* ASK THE USER IF HE/SHE DESIRES TO ACCEPT THE CHANGES.
* SET COLOR TO W+/B, W+/B
@ 19,12 SAY "Do you want to accept the changes? (Yes or No): "
SET COLOR TO R+/B, R+/B
@ 19,49 SAY "Y"
@ 19,56 SAY "N"
STORE "N" TO ACCEPT
@ 19,62 GET ACCEPT PICT ";"
READ
ENSURE THAT THE USER'S PROMPT IS EITHER "Y" OR "N"
DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
@ 24,24 SAY "Response must be either N or Y"
DO DELAY
STORE "N" TO ACCEPT
ENDIF
SET COLOR TO R+/B, R+/B
@ 19,62 GET ACCEPT PICT ";"
READ
ENDDO
@ 19,62 SAY " "
IF ACCEPT = "Y" THEN
REPLACE UNIT PRICE WITH MPRICE
REPLACE MAINT WITH MMAINT
REPLACE UNIT INSTA WITH MINSTALL
REPLACE MQTY WITH MQTY
ENDIF
SET COLOR TO W/B, W/B
@ 19,10 SAY SPACE(80)
SET COLOR TO R+/B, R+/B
APPENDIX B: MAINTENANCE MANUAL

Page 6 EQUIPUPD.PRG Program Listing

251 STORE "N" TO CHOICE
252 @ 21,68 GET CHOICE PICT ":1"
253 READ
254 * ENSURE THAT THE USER'S PROMPT IS EITHER "N", "P" OR "X"
255 * DO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X")
256   IF .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X") THEN
257     SET COLOR TO W/B, W/B
258     @ 24,0 SAY SPACE(80)
259     SET COLOR TO W+/R, W+/R
260     @ 24,23 SAY "Response must be either N, P or X"
261     DO DELAY
262     STORE "N" TO CHOICE
263     ENDIF
264     SET COLOR TO W+/R, W+/R
265     @ 24,21 SAY EOF
266     DO DELAY
267     ELSE
268     ENDIF (EOF() = .T.)
269     ELSE (EOF() = .F.)
270     IF CHOICE = "N" THEN
271       IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
272         SKIP
273         IF EOF() = .T. THEN
274           SKIP = 1
275           SET COLOR TO W+/R, W+/R
276           @ 24,21 SAY EOF
277           DO DELAY
278         ELSE
279           IF .NOT. (SITEMO = MSITE) THEN
280             SKIP = 1
281             SET COLOR TO W+/R, W+/R
282             @ 24,21 SAY EOF
283             DO DELAY
284           ENDIF
285           ENDIF EOF() = .T.
286         ELSE
287           IF RECMD() = LAST_REC THEN
288             DO TOP
289           ELSE
290             SKIP
291           ENDIF
292           ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE)
293           IF CHOICE = "N" THEN
294             SKIP TO THE PREVIOUS RECORD
295           IF CHOICE = "P" THEN
APPENDIX B: MAINTENANCE MANUAL

EQUIPUPD.PRG Program Listing

301 | STORE RECN0() TO CURRENTNO
302 | IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
303 |    SKIP - 1
304 |    IF BOF() = .T. THEN
305 |        GOTO CURRENTNO
306 |        SET COLOR TO W/R, W/R
307 |        @ 24,16 SAY TOF
308 |        DO DELAY
309 |    ELSE
310 |        IF .NOT. (SITENO = MSITE) THEN
311 |          . SKI P
312 |          SET COLOR TO W/R, W/R
313 |          DO DELAY
314 |          ENDIF
315 |        ELSE
316 |          ENDIF BOF() = .T.
317 |        ELSE
318 |          IF RECN0() = FIRST_REC THEN
319 |              GO BOTTOM
320 |          ELSE
321 |            SKIP - 1
322 |          ENDIF
323 |        ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE'58')
324 |        ELSE
325 |          IF CHOICE = "P"
326 |            USER HAS DECIDED TO EXIT THE REVIEW
327 |          ENDIF
328 |          IF CHOICE = "X"
329 |            EXIT
330 |          ENDIF
331 |          ENDW HILE .T.
332 |    * RETURN TO CALLING PROGRAM.
333 |    * RELEASE ALL LIKE MS, ACCEPT, CHOICE, CURRENTNO, BOF, ERROR,;
334 |    * FIRST_REC, INTRO, LAST_REC, TOF
335 |    * CLOSE DATABASES
336 |    RETURN
337 |************************************************************
APPENDIX B: MAINTENANCE MANUAL

MAINMENU.PRG Program Listing

* PROCEDURE MAINMENU.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
            LCDR WINSTON H. BUCKLEY, SC, USN
            LCDR ROBERT F. BRADO, USN
            LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : PROVIDE THE USER THE CHOICE OF LOADING A NEW DELIVERY,
            ORDER, MAINTAINING THE EQUIPMENT, MANUAL, AND
            SERIAL NUMBER DATA BASES OR GETTING A SERIES OF
            REPORTS FROM THESE UPDATED DATABASES.

* INPUT FILES : NONE.

* OUTPUT FILES : NONE.

* CALLED BY : SELECTOR.PRG

* MODULES CALLED : NEWLOCMD.PRG, EQUIPMD.PRG, MANULCMD.PRG,
                   SERVOCMD.PRG, REPORTCMD.PRG, DESPMOD.PRG,
                   COMEMOD.PRG, DELAY.PRG, MAINTD.PRG, MKLABELS.PRG

* GLOBAL VARIABLES : HIDATE, HIPNUM, HISITE, LOCATE, LOFNUM, LISITE

* LOCAL VARIABLES : Ans

* DATE LAST TIME MODIFIED =========== 23 DECEMBER 1985 ===========

* DBASE PROGRAM CONFIGURATION VARIABLES:

SET BELL OFF
SET CONSOLE ON
SET INTENSITY OFF
SET SCOREBOARD OFF
SET TALK OFF

PUBLIC HIDATE, HIPNUM, HISITE, LOCATE, LOFNUM, LISITE

* INITIALIZE THE PUBLIC VARIABLES

STORE '0001231' TO HIDATE
STORE '994001' TO HIPNUM
STORE '59' TO HISITE
STORE '840101' TO LOCATE
STORE '000101' TO LOFNUM
STORE '01' TO LISITE

* DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE USER'S CHOICE.

STORE "1" TO Ans
DO WHILE Ans
FLASH = CHR(145)
SET COLOR TO W/B, W/B, B
?? FLASH + "S.MAINMENU.SCR/"
@ 24,0 SAY SPACE (80)
SET COLOR TO R+/B, R+/B
@ 22,53 GET ANS PICT "9"
READ

* PERFORM APPROPRIATE TASK BASED ON THE USER'S CHOICE.

DO CASE
   CALL THE NEW DELIVERY ORDER LOAD COMMAND PROGRAM.
   CASE ANS = "1"
      DO NEWDOCMD
      STORE "1" TO ANS
      DO CASE
   CALL THE EQUIPMENT FILE MAINTENANCE COMMAND PROGRAM.
   CASE ANS = "2"
      DO EQUIPCMMD
      STORE "2" TO ANS
      DO CASE
   CALL THE DESCRIPTION FILE MAINTENANCE COMMAND PROGRAM.
   CASE ANS = "3"
      DO DESPMOD
      STORE "3" TO ANS
      DO CASE
   CALL THE SITE CONFIGURATION FILE MAINTENANCE COMMAND PROGRAM.
   CASE ANS = "4"
      DO CONFMOD
      STORE "4" TO ANS
      DO CASE
   CALL THE MANUAL FILE MAINTENANCE COMMAND PROGRAM.
   CASE ANS = "5"
      DO MANULCMD
      STORE "5" TO ANS
      DO CASE
   CALL THE SERIAL NUMBER MAINTENANCE COMMAND PROGRAM.
   CASE ANS = "6"
      DO SERNUMCMD
      STORE "6" TO ANS
      DO CASE
   CALL THE REPORTS GENERATION COMMAND PROGRAM.
   CASE ANS = "7"
      DO REPORTCMD
      STORE "7" TO ANS
      DO CASE
   CALL THE MAINTENANCE DELIVERY ORDER GENERATION PROGRAM
   CASE ANS = "8"
      DO MAINTLD

STORE "8" TO ANS

CALL THE MAILING LABELS GENERATION PROGRAM

CASE ANS = "9"
DO MKLABELS
STORE "9" TO ANS
ENDCASE

RETURN THE USER TO SELECTOR PROGRAM CONTROL.

CASE ANS = "0"
CLOSE DATABASES
RETURN

ENDCASE

CONTINUE PROCESSING LOOP CONTROL CHECK.

ENDDO WHILE .T.

***************************************************************************
**MAINTDO.PRG Program Listing**

```plaintext
* PROCEDURE MAINTDO.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN  
  LCDR WINSTON H. ECKLEY, SC, USN  
  LCDR ROBERT F. BRADO, USN  
  LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : PROVIDE THE INPUTS FOR A MAINTENANCE DELIVERY ORDER, WHICH WILL BE IMPORTED INTO LOTUS 1-2-3.

* INPUT FILES : EQUIP.DBF, DESCRIP.DBF, DESCRIP.NDX, TEMPONE.DBF  
  EFEAT.NDX, TEMtwor.DBF, TEMPTIRE.DBF, TEMPFour.DBF

* OUTPUT FILE : NEWDO.PRN

* CALLED BY : MAINMENU.PRG

* MODULES CALLED : DELAY.PRG

* GLOBAL VARIABLE: HISITE, LOSITE

* LOCAL VARIABLES: ERROR, MESSAGE, MSITE, NOFIND, RATES, SITES

* DATE LAST TIME MODIFIED ============== 27 DECEMBER 1985 =============

SET ESCAPE OFF
SET TALK OFF
SET COLOR TO W+/B, W+/B, B
CLEAR
?? FLASH + "S.MAINTDO.SCR/"
0 24,0 SAY SPACE(80)
STORE ": Enter the number of the site for which the maintenance is " ,
"to be performed " TO SITES
STORE SPACE(29) + ": Enter the Discount and Escalation Rates ":,  
SPACE(29) TO RATES
SET COLOR TO /BR, /BR
? 20,57 SAY " NEWDO.PRN "

* OBTAIN THE NUMBER OF THE SITE TO RECEIVE THE MAINTENANCE FROM THE USER

* USE EQUIP INDEX EQUIPINDEX.NDX

* DO WHILE .T.

44 SET COLOR TO /W, /W
45 0 24,0 SAY SITES
46 SET COLOR TO R+/R, R+/R  
47 STORE LOSITE TO MSITE
48 0 04,65 GET MSITE PICT ":m"
49 READ
50 IF .NOT. (MSITE >= HISITE .AND. MSITE <= LOSITE) THEN
```
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
STORE ' Response must be between ' + LOSITE +;
' and ' + HISITE + ' TO ERROR
@ 24,22 SAY ERROR
DO DELAY
LOOP
ELSE
GO TOP
FIND &MSITE
IF EOF() = .T. THEN
    SET COLOR TO W/B, W/B
    @ 24,0 SAY SPACE(80)
    SET COLOR TO W+/R, W+/R
    STORE " No records for site number " + MSITE +;
    " exist, try again " TO MESSAGE
    @ 24,16 SAY MESSAGE
    DO DELAY
    LOOP
ELSE
    EXIT
END IF EOF() = .T.
ENDIF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE)
ENDDO WHILE .T.
*
* ENSURE THAT TEMPORARY DATABASES DO NOT EXIST, IF SO ERASE THEM
*
SET CONSOLE OFF
ERASE TEMPONE.DBF
ERASE TEMPONE.NDX
ERASE TEMPTWO.DBF
ERASE TEMPTHRE.DBF
ERASE TEMPFOUR.DBF
SET CONSOLE ON
*
* INFORM THE USER THAT THERE WILL BE A SLIGHT DELAY
*
SET COLOR TO W+/R, W+/R
STORE SPACE(10) + "Creating a temporary database and index. " +;
"PLEASE BE PATIENT " + SPACE(10) TO MESSAGE
@ 24,0 SAY MESSAGE
COPY TEMPONE.DBF WHILE SITENO = "&MSITE"
USE TEMPONE
INDEX ON FEATURENO TO TEMPONE
TOTAL ON FEATURENO TO TEMPTWODBF FIELDS QTY WHILE FEATURENO <> 'XXXXXX'
*
* OBTAIN THE DISCOUNT AND ESCALATION RATES FROM THE USER
*
SET COLOR TO /W, /W
MAINTDO.PRG Program Listing

101 @ 24,0 SAY RATES
102 STORE "0.000" TO LCNHWRATE
103 STORE "0.000" TO LCNWSRATE
104 STORE "0.000" TO SNETSWRATE
105 STORE "0.000" TO UPLIFT
106 SET COLOR TO /BR, /BR
107 @ 14,61 GET LCNHWRATE PICT "9.999"
108 @ 15,61 GET LCNWSRATE PICT "9.999"
109 @ 16,61 GET SNETSWRATE PICT "9.999"
110 @ 17,61 GET UPLIFT PICT "9.999"
111 READ
112 * ASK TO USER TO VERIFY THAT HE/SHE WANTS TO CONTINUE
113 *
114 SET COLOR TO W+/B, W+/B
115 @ 24,0 SAY SPACE(80)
116 @ 22,22 SAY "Do you want to Continue or exit? "
117 SET COLOR TO R+/B, R+/B
118 @ 22,37 SAY "C"
119 @ 22,50 SAY "X"
120 STORE "C" TO CHOICE
121 @ 22,56 GET CHOICE PICT "!
122 READ
123 *
124 * ENSURE THAT THE USER'S RESPONSE IS EITHER "N", "P" OR "X"
125 *
126 DO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
127 IF .NOT. (CHOICE = "C" .OR. CHOICE = "X") THEN
128 SET COLOR TO W+/R,W+/R
129 @ 24,24 SAY "Response must be either C or X"
130 DO DELAY
131 STORE "C" TO CHOICE
132 ENDIF .NOT. (CHOICE = "C" .OR. CHOICE = "X")
133 SET COLOR TO R+/B,R+/B
134 @ 22,56 GET CHOICE PICT "!
135 READ
136 ENDDO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
137 *
138 SET COLOR TO W/B, W/B
139 @ 22,20 SAY SPACE(50)
140 IF CHOICE = "C" THEN
141 STORE 1 + VAL(LCNHWRATE) TO LCNHWRATE
142 STORE 1 + VAL(LCNWSRATE) TO LCNWSRATE
143 STORE 1 + VAL(SNETSWRATE) TO SNETSWRATE
144 STORE 1 + VAL(UPLIFT) TO UPLIFT
145 ELSE
146 SET CONSOLE OFF
147 CLOSE DATABASES
148 ERASE TEMPO. DIF
149 ERASE TEMPO. NDX
150
Page 4 MAINTECLPRG Program Listing

151 | ERASE TEMPLTWO.DBF
152 | ERASE TEMPLTHREE.DBF
153 | ERASE TEMPLFOUR.DBF
154 | SET CONSOLE ON
155 | SET COLOR TO W/B, W/B
156 | @ 24,0 SAY SPACE(80)
157 | RELEASE ERROR, MESSAGE, MSITE, NOFIND, RATES, SITES
158 | RETURN
159 |
160 | ENDIF
161 | *
162 | * INFORM THE USER THAT THERE WILL BE A SLIGHT DELAY
163 | *
164 | SET COLOR TO W+/R, W+/R
165 | STORE " Creating the MAINTENANCE DELIVERY ORDER may take up to 10 " +;
166 | " minutes. PLEASE WAIT " TO MESSAGE
167 | @ 24,0 SAY MESSAGE
168 | SELECT 1
169 | USE TEMPLTWO
170 | USE DESCRIP
171 | SELECT TEMPLTHREE
172 | JOIN WITH DESCRIP TO TEMPLFOUR FOR FEATURENO = DESCRIP->FEATURENO
173 | SELECT 3
174 | USE TEMPLTHREE
175 | GO TOP
176 | REPLACE ALL NO MAINT WITH BASEMAINT*LANOWRAT FOR FEATURENO = "320900" .AND.,
177 | FEATURENO < "420400"
178 | GO TOP
179 | REPLACE ALL NO MAINT WITH BASEMAINT*LANOWRAT FOR FEATURENO = "550901"
180 | REPLACE ALL NO MAINT WITH BASEMAINT*LANOWRAT FOR FEATURENO = "550901"
181 | REPLACE ALL NO MAINT WITH BASEMAINT*LANOWRAT FOR FEATURENO = "551001"
182 | REPLACE ALL NO MAINT WITH BASEMAINT*LANOWRAT FOR FEATURENO = "551101"
183 | REPLACE ALL NO MAINT WITH BASEMAINT*LANOWRAT FOR FEATURENO = "551201"
184 | REPLACE ALL NO MAINT WITH BASEMAINT*LANOWRAT FOR FEATURENO = "551301"
185 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "550710"
186 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "550710"
187 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "550801"
188 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "550901"
189 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551001"
190 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551101"
191 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551201"
192 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551301"
193 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551404"
194 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551304"
195 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551304"
196 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551404"
197 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551404"
198 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551404"
199 | REPLACE ALL NO MAINT WITH BASEMAINT*SNOWRAT FOR FEATURENO = "551404"
200 | SELECT 4
APPENDIX B: MAINTENANCE MANUAL

Page 5

MAINTDO.PRG Program Listing

201 USE TED
202 COPY STRUCTURE TO TEMPFOUR
203 CLOSE DATABASES
204 USE TEMPFOUR
205 APPEND FROM TEMPTIERE
206 GO TOP
207 REPLACE ALL MAINT/MOS WITH 12
208 REPLACE ALL MAINT_FAC WITH UPLIFT
209 REPLACE ALL MAINT_FAC WITH 1 FOR FEATURENO = "550801"
210 REPLACE ALL MAINT_FAC WITH 1 FOR FEATURENO = "550901"
211 REPLACE ALL MAINT_FAC WITH 1 FOR FEATURENO = "551001"
212 REPLACE ALL MAINT_FAC WITH 1 FOR FEATURENO = "551101"
213 REPLACE ALL MAINT_FAC WITH 1 FOR FEATURENO = "551201"
214 REPLACE ALL MAINT_FAC WITH 1 FOR FEATURENO = "551301"
215 REPLACE ALL TOT_MAINT WITH MAINT_FAC*MO * MAINT*MOS
216 REPLACE ALL COMP_DT CR WITH ((UNIT_PRICE + UNIT_INSTA)/48) *;
217 (MO_MAINT * MAINT_FAC) * .005
218 REPLACE ALL SYS DT CR WITH (QTY*MO_MAINT*MAINT_FAC)
219 REPLACE ALL TOT_MAINT WITH TOT_MAINT*QTY FOR FEATURENO > "010200" .AND.;
220 REPLACE ALL UNIT_PRICE WITH 0
221 REPLACE ALL TOT_PRICE WITH 0
222 REPLACE ALL UNIT_INSTA WITH 0
223 REPLACE ALL TOT_INSTAL WITH 0
224 COPY TO NEWDO.PRN DELIMITED
225 ERASE ALL TEMPORARY DATABASES AND INDICES CREATED DURING THE PROGRAM
226 *
227 SET CONSOLE OFF
228 CLOSE DATABASES
229 ERASE TEMPONE.DBF
230 ERASE TEMPNO.DBF
231 ERASE TEMPTIERE.DBF
232 ERASE TEMPFOUR.DBF
233 ERASE TEMPONE.NDX
234 SET CONSOLE ON
235 *
236 RETURN TO CALLING PROGRAM
237 *
238 SET COLOR TO W/B, W/B
239 & 24,0 SAY SPACE(80)
240 RELEASE ERROR, MESSAGE, MSG, DEFIN, HAT, HT, SITES
241 RETURN
242 ********************************************************************************
* PROCEDURE MANULADD.PRG
*
* AUTHORS : LCDR EDWARD J. CASE, SC, USN
* LCDR WINSTON H. BUCKLEY, SC, USN
* LCDR ROBERT F. BRADO, USN
* LCDR ROBERT L. BEARD III, SC, USN
*
* PURPOSE : ADD NEW MANUALS TO THE MANUAL DATABASE FILE.
*
* INPUT FILES : MANUAL.DBF, MANULSIT.NDX
*
* CALLED BY : MANULCMD.PRG
*
* MODULES CALLED : DELAY.PRG
*
* GLOBAL VARIABLE: HIFNUM, HISITE, LOFNUM, LOSITE
*
* LOCAL VARIABLES: ACCEPT, CHOICE, ERROR, FEATURES, GETUIT,
* MCLIN, MANUDESCRPT, MDESCRPT, MESSAGE, MFEATS,
* MSITE, NDFIND, NSITE, SITES
*
* DATE LAST TIME MODIFIED = 23 DECEMBER 1985
*
* CASE SELECTION = 1      ADD A NEW MANUAL DESCRIPTION
*
SET ESCAPE OFF
SET TALK OFF
USE MANUAL
GO TOP
CLEAR
IF EOF() = .T. THEN
SET COLOR TO W+/B, W+/B, B
ENDIF
SELECT 1
USE MANUAL INDEX MANULSIT
*
?? FLASH = "S.MANUALS.SCR/"
@ 24,0 SAY SPACE(80)
@ 22,10 SAY SPACE(60)
SET COLOR TO GR+/B, GR+/B
@ 6,28 SAY " Last "
SET COLOR TO R+/ , R+/ 
@ 1,26 SAY " MANUAL ADDITION FORMAT "
SET COLOR TO W+/B, W+/B
@ 22,23 SAY "Enter C to continue or X to exit: ">
SET COLOR TO R+/B, R+/B

321
APPENDIX B: MAINTENANCE MANUAL

Page 2

MANULADD.PRG Program Listing

@ 22,29 SAY "C"
@ 22,46 SAY "X"
* GENERATE STATUS MESSAGES
STORE ' Enter a Site Number between ' + LOSITE + ' and ' + HISITE + ' TO SITES
STORE ' Enter a Feature Number ( ' + LOFNUM + ' - ' + HIFNUM + ' ) ' TO FEATURES
STORE SPACE(20) + 'Enter the Manual Description to be Added' TO MANDESCRIP
* DO WHILE .T.
  SET COLOR TO R+/B, R+/B
  @ 6,47 SAY RECNO() PICT "9999"
* CLEAR SCREEN AND SET INITIAL VALUES FOR VARIABLES TO BE
  ADDED TO THE FILE. THE M PREFIX INDICATES MEMORY VARIABLES
  DISTINGUISHING THEM FROM THEIR CORRESPONDING DATABASE FIELDS.
  STORE ' TO MFEAT
  STORE ' TO MANDESC
  SET COLOR TO /W, /W
  @ 24,0 SAY SITES
  * ENSURE THAT THE SITE NUMBER IS A VALID SITE
  * DO WHILE .T.
    SET COLOR TO /BR, /BR
    STORE LOSITE TO MSITE
    @ 9,45 GET MSITE PICT '99'
    READ
    IF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
      SET COLOR TO W+/B, W+/B
      @ 24,0 SAY SPACE(80)
      SET COLOR TO W+/R, W+/R
      STORE ' Response must be between ' + LOSITE + ' and ' + HISITE + ' ' TO ERROR
      @ 24,22 SAY ERROR
    ELSE
      GO 'loP
    END IF
    IF EOF() = .T. THEN
      SET COLOR TO W+/B, W+/B
      @ 24,0 SAY SPACE(80)
    END IF
  END DO
  ELSE
    GO 'loP
END DO
APPENDIX B: MAINTENANCE MANUAL

Page 3

MANULADD.PRG Program Listing

101 SET COLOR TO W+/R, W+/R
102 STORE "No records exist for site " + MSITE + "", try another site " to NOSITE
103 @ 24,16 SAY NOSITE
104 DO DELAY
105 SET COLOR TO /W, /W
106 @ 24,0 SAY SITES
107 STORE "99" TO MSITE
108 LOOP
109 
110 ELSE
111 EXIT
112 ENDF (EOF()) = .T.
113 ENDF NOT. (MSITE = LOSITE .AND. MSITE = HISITE)
114 ENDDO WHILE .T.
115 
116 DO BOTTOM
117 SET COLOR TO /W, /W
118 @ 24,0 SAY FEATURES
119 SET COLOR TO /BR, /BR
120 STORE 0 TO NOFIND
121 STORE "N" TO GETIOUT
122 
123 "ENSURE THAT THE FEATURE IS A VALID FEATURE"
124 
125 DO WHILE .NOT. (MFEAT = LOFINUM .AND. MFEAT < HIFNUM)
126 
127 "IF THE USER HAS MADE THREE ATTEMPTS TO SPECIFY A VALID .PRN NAME AND HAS NOT BEEN SUCCESSFUL, ASK HIM/HER IF THEY DESIRE EXIT THIS PROCESS."
128 
129 IF NOFIND = 3 THEN
130 
131 SET COLOR TO W+/B, W+/B
132 @ 19,15 SAY "Do you want to exit this process? (Yes or No): "
133 SET COLOR TO R+/B, R+/B
134 @ 19,51 SAY "Y"
135 @ 19,58 SAY "N"
136 STORE "Y" TO GETOUT
137 @ 19,63 GET GETOUT PICT "!"
138 READ
139 
140 DO WHILE .NOT. (GETOUT = "Y" OR GETOUT = "Y")
141 IF .NOT. (GETOUT = "Y" OR GETOUT = "Y") THEN
142 SET COLOR TO W+/B, W+/B
143 @ 24,24 SAY "Response must be either 'Y' or 'N'
144 DO DELAY
145 STORE "Y" TO GETOUT
146 ENDF
147 SET COLOR TO R+/B, R+/B
148 @ 19,63 GET GETOUT PICT "!"
149 READ

323
APPENDIX B: MAINTENANCE MANUAL

MANULADD.PRG Program Listing

ENDDO

SET COLOR TO W/B, W/B
@ 19,10 SAY SPACE(65)
IF GETOUT = "Y" THEN
EXIT
ELSE
STORE 0 TO NOFIND
SET COLOR TO /W, /W
@ 24,0 SAY FEATURES
LOOP
ENDIF
ENDIF
IF GETOUT = "Y" THEN
EXIT
ENDIF
SET COLOR TO /BR, /BR
STORE LOFNUM 'a) MFEAT
@ 12,45 GET MFEAT PICT '999999'
READ

ENSURE THAT THE FEATURE NUMBER ENTERED BY THE USER IS VALID

IF .NOT. (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM) THEN
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
STORE ' Response must be between ' + LOFNUM + ;
' and ' + HIFNUM + ' TO ERROR
@ 24,18 SAY ERROR
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY FEATURES
ELSE
SELECT 2
USE EQUIP INDEX MFEAT
GO TOP
FIND MFEAT
IF EOF() = .T. THEN
NOFIND = NOFIND + 1
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
STORE ' Feature Number " + MFEAT + ";
"does not exist, try again " TO MESSAGE
IF NOFIND = 3 THEN
@ 24,16 SAY MESSAGE
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY FEATURES

324
APPENDIX B: MAINTENANCE MANUAL

Page 188

Page 5 MANULADD.PRG Program Listing

201 ENDIF
202 STORE "999999" TO MFEAT
203 SELECT 1
204 ENDIF IF() = .T.
205 ENDIF NOT (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
206 ENDDO WHILE NOT (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
207 *
208 SET COLOR TO W/B, W/B
209 " 24,0 SAY SPACE(80)
210 IF GETOUT = "y" THEN
211 EXIT
212 ENDIF
213 SELECT 3
214 USE DESCRIP INDEX DESCRIP
215 GO TOP
216 FIND &MFEAT
217 STORE CLIN TO MCLIN
218 STORE DESCRIP TO MDESCRIP
219 SELECT 1
220 SET COLOR TO /BR, /BR
221 " 13,45 SAY MCLIN PICT "9999"
222 " 14,45 SAY MDESCRIP PICT "!" : !" : !":" : !":" :
223 *
224 SET COLOR TO /W, /W
225 " 24,0 SAY MANDESCRIP
226 SET COLOR TO /BR, /BR
227 " 17,45 GET MMANDESC PICT "!" : !" : !": !":" :
228 READ
229 SET COLOR TO W/B, W/B
230 " 24,0 SAY SPACE(80)
231 *
232 * IF .NOT. (MANDESC = MMANDESC) THEN
233 * ASK THE USER IF HE/SHE DESIRES TO ACCEPT THE CHANGES
234 *
235 SET COLOR TO W/B, W/B
236 " 20,12 SAY "Do you want to accept the change? (Yes or No):"
237 SET COLOR TO R/B, R/B
238 " 20,49 SAY "y"
239 " 20,56 SAY "N"
240 STORE "N" TO ACCEPT
241 " 20,62 GET ACCEPT PICT "i"
242 READ
243 *
244 * ENSURE THAT THE USER'S RESPONSE IS EITHER "y" OR "N"
245 *
246 DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "y")
247 IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "y") THEN
248 SET COLOR TO W/R, W/R
249 " 24,24 SAY "Response must be either N or Y"
APPENDIX B: MAINTENANCE MANUAL

MANULADD.PRG Program Listing

251   |   DO DELAY
252   |   STORE "N" TO ACCEPT
253   |   ENDF
254   |   SET COLOR TO R+/E, R+/B
255   |   @ 20, 62 GET ACCEPT PICT "."
256   |   READ
257   |   ENDDO
258   |   SET COLOR TO W/B, W/B
259   |   @ 20, 10 SAY SPACE(55)
260   |   *
261   |   IF ENTRIES ARE CORRECT, ADD THEM TO DATABASE.
262   |   *
263   |   IF ACCEPT = "Y"
264   |       APPEND BLANK
265   |       REPLACE SITE NO WITH MSITE
266   |       REPLACE FEATURE NO WITH M FEATURE
267   |       REPLACE MAN DESC WITH M MAN DESC
268   |   ENDF
269   |   *
270   |   ENDF
271   |   *
272   |   SET COLOR TO R+/B, R+/B
273   |   STORE "C" TO CHOICE
274   |   @ 22, 58 GET CHOICE PICT "."
275   |   READ
276   |   *
277   |   ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
278   |   *
279   |   DO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
280   |       IF .NOT. (CHOICE = "C" .OR. CHOICE = "X") THEN
281   |           SET COLOR TO W+/R, W+/R
282   |           @ 24, 24 SAY "Response must be either C or X"
283   |       ENDIF
284   |       STORE "C" TO CHOICE
285   |   ENDF
286   |   SET COLOR TO R+/B, R+/B
287   |   @ 22, 58 GET CHOICE PICT "."
288   |   READ
289   |   ENDDO
290   |   *
291   |   SKIP TO THE NEXT RECORD TO BE REVIEWED
292   |   *
293   |   IF CHOICE = "C" THEN
294   |       STORE " " TO M CLIN
295   |       STORE SPACE(30) TO M DESCRIPT
296   |       STORE SPACE(26) TO M MAN DESC
297   |       SET COLOR TO /IR, /IR
298   |       @ 12, 45 SAY ""
299   |       @ 13, 45 SAY M CLIN PICT "9999"
300   |       @ 14, 45 SAY M DESCRIPT PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!"

326
APPENDIX B: MAINTENANCE MANUAL  Page 190

Page 7  MANULADD.PRG Program Listing

301      SAY MMANDESC PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!"
302      SKIP
303      ENDF
304      *
305      *  USER HAS DECIDED TO EXIT THE REVIEW
306      *
307      IF CHOICE = "X"
308      EXIT
309      ENDF
310      *
311      ENDDO WHILE .T.
312      *
313      *  RETURN TO CALLING PROGRAM.
314      *
315      RELEASE ALL LIKE M*, ACCEPT, CHOICE, ERROR, FEATURES, GETOUT,;
316      NOFIND, NOSITE, SITES
317      CLOSE DATABASES
318      RETURN
319      ############################################################################
* PROCEDURE MANULCMD.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
* LCDR WINSTON H. BUCKLEY, SC, USN
* LCDR ROBERT F. Brado, USA
* LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : PROVIDE THE USER THE OPPORTUNITY TO ADD A MANUAL
* RECORD, UPDATE AN EXISTING RECORD, DELETE AN EXISTING
* RECORD OR REVIEW CURRENT RECORDS.

* INPUT FILES : NONE.

* OUTPUT FILES : NONE.

* CALLED BY : MAINMENU.PRG

* MODULES CALLED : MANULADD.PRG, MANULUPD.PRG, MANULDEL.PRG,
* MANULREV.PRG

* LOCAL VARIABLES: SELEKT

* DATE LAST TIME MODIFIED =============== 23 DECEMBER 1985 =============

* DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE SELECTION.

STORE "1" TO SELEKT
DO WHILE SELEKT < "5"
  SET COLOR TO W/B, W/B, B
  CLEAR
  ?? FLASH + "W.MANULCMD/"
  SET CONSOLE OFF
  WAIT TO SELEKT
  SET CONSOLE ON

* PROCESS ROUTINE BASED ON THE USER'S SELECTION.

DO CASE

* CALL THE MANUAL ADD PROGRAM.
  CASE SELEKT = "1"
    DO MANULADD

* CALL THE MANUAL UPDATE PROGRAM.
  CASE SELEKT = "2"
    DO MANULUPD

* CALL MANU IAL DELETION PROGRAM.
  CASE SELEKT = "3"
    DO MANULDEL
APPENDIX B: MAINTENANCE MANUAL

Page 192

MANULCMD.PRG Program Listing

51 * 
52 * CALL MANUAL REVIEW PROGRAM.
53 CASE SELEKT = "4"
54 DO MANULREV
55 *
56 * RETURN TO THE MAIN MENU PROGRAM.
57 CASE SELEKT = "5"
58 *
59 ENDCASE
60 *
61 ENDDO (WHILE SELEKT < "5")
62 *
63 * RETURN TO THE CALLING PROGRAM
64 *
65 RETURN
66 ******************************************
APPENDIX B: MAINTENANCE MANUAL

MANULDEL.PRG Program Listing

1 * PROCEDURE MANULDEL.PRG
2 *
3 * AUTHORS : LCDR EDWARD J. CASE, SC, USN
4 * LCDR WINSTON H. BUCKLEY, SC, USN
5 * LCDR ROBERT F. Brado, USN
6 * LCDR ROBERT L. BEARD III, SC, USN
7 *
8 * PURPOSE : DELETE MANUAL RECORDS FROM THE MANUAL DATABASE FILE.
9 *
10 * INPUT FILES : MANUAL.DBF, MANULSIT.NDX
11 *
12 * CALLED BY : MANULCMD.PRG
13 *
14 * MODULES CALLED : DELAY.PRG
15 *
16 * GLOBAL VARIABLE: HIFNUM, HISITE, LOFNUM, LOSITE
17 *
18 * LOCAL VARIABLES: ACCEPT, CHOICE, ERROR, FEATURES, FIRST_REC,
19 * LAST_REC, MCLIN, MDSCIP, MESSAGE, MFEAT,
20 * MKEY, MMANDESC, MSITE, PACKEM, SITES
21 *
22 * DATE LAST TIME MODIFIED = 24 DECEMBER 1985
23 *
24 * CASE SELECTION = 3 DELETE AN EXISTING MANUAL RECORD
25 *
26 SET DELETED ON
27 SET ESCAPE OFF
28 SET TALK OFF
29 USE MANUAL
30 GO TOP
31 CLEAR
32 IF EOF(1) = .T. THEN
33 SET COLOR TO W+/B, W+/B, B
34 ELSE
35 13,25 SAY "The MANUALS Database is EMPTY!"
36 10 DELAY
37 RETURN
38 ENDIF
39 SELECT 1
40 USE MANUAL INDEX MANULSIT
41 (2) BOTTOM
42 STORE RECN0() TO LAST_REC
43 *
44 ?? FLASH + "S.MANUALS.SCR"
45 @ 24,9 SAY SPACE(80)
46 @ 22,10 SAY SPACE(60)
47 SET COLOR TO R+/+, R+/+
48 & 3,26 SAY "MANUAL DELETION FORMAT"
49 SET COLOR TO W+/B, W+/B
50 & 22,23 SAY "Enter C to continue or X to exit;"
SET COLOR TO R+/B, R+/B
52 @ 22,29 SAY "C"
53 @ 22,46 SAY "X"
54 STORE SPACE(9) + "Enter the Site Number for the Manual " +
55 "Description to be Deleted" + SPACE(9) TO SITES
56 STORE SPACE(10) + "Enter the Feature Number for the Manual " +
57 "Description Deletion" + SPACE(10) TO FEATURES
58 STORE "Records marked for deletion have been deleted and " +
59 "CAN NOT" + SPACE(2) + "be recovered" TO PACKEM
60 STORE "Are you sure you want to delete this description? " +
61 "(Yes or No):" TO MESSAGE
62 *
63 SET COLOR TO /W, /W
64 @ 24,0 SAY SITES
65 *
66 ENSURE THAT THE SITE NUMBER IS A VALID SITE
67 *
68 STORE ' ' TO MSITE
69 DO WHILE .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE)
70 SET COLOR TO /BR, /BR
71 STORE LOSITE TO MSITE
72 @ 9,45 GET MSITE PICT '99'
73 READ
74 IF .NOT. (MSITE = LOSITE .AND. MSITE <= HISITE)
75 SET COLOR TO /W/B, /W/B
76 @ 24,0 SAY SPACE(80)
77 SET COLOR TO W+/R, W+/R
78 STORE ' Response must be between ' + LOSITE +
79 ' and ' + HISITE + ' TO ERROR
80 @ 24,22 SAY ERROR
81 DO DELAY
82 SET COLOR TO /W, /W
83 @ 24,0 SAY SITES
84 LOOP
85 ELSE
86 GO TOP
87 FIND &MSITE
88 IF EOF() = .T. THEN
89 SET COLOR TO W/B, W/B
90 @ 24,0 SAY SPACE(80)
91 SET COLOR TO W+/R, W+/R
92 STORE ' No record for site number ' + MSITE +
93 ' exists, try again ' TO ERROR
94 @ 24,16 SAY ERROR
95 DO DELAY
96 SET COLOR TO /W, /W
97 @ 24,0 SAY SITES
98 STORE '99' TO MSITE
99 ENDIF EOF() = .T.
100 ENDIF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE)
APPENDIX B: MAINTENANCE MANUAL

MANUDEL.PRG Program Listing

101 END DO WHILE .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE)
102 *
103 SET COLOR TO W/B, W/B
104 @ 24,0 SAY SPACE(80)
105 STORE " " TO MFEAT
106 SET COLOR TO /W, /W
107 @ 24,0 SAY FEATURES
108 *
109 * ENSURE THAT THE FEATURE IS A VALID FEATURE
110 *
111 DO WHILE .NOT. (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
112 SET COLOR TO /BR, /BR
113 STORE LOFNUM TO MFEAT
114 @ 12,45 GET MFEAT PICT '999999'
115 READ
116 IF .NOT. (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
117 SET COLOR TO W/B, W/B
118 @ 24,0 SAY SPACE(80)
119 SET COLOR TO W+/R, W+/R
120 STORE 'Response must be between ' + LOFNUM + ' and ' + HIFNUM + ' TO ERROR
121 @ 24,18 SAY ERROR
122 DO DELAY
123 SET COLOR TO /W, /W
124 @ 24,0 SAY FEATURES
125 LOOP
126 ELSE
127 STORE MSITE + MFEAT TO MKEY
128 GO TO
129 FIND &MKEY
130 IF EOF() = .T. THEN
131 SET COLOR TO W/B, W/B
132 @ 24,0 SAY SPACE(80)
133 SET COLOR TO W+/R, W+/R
134 STORE 'No record exists for feature number ' + MFEAT + ' , try again ' TO ERROR
135 @ 24,12 SAY ERROR
136 DO DELAY
137 SET COLOR TO /W, /W
138 @ 24,0 SAY FEATURES
139 STORE '999999' TO MFEAT
140 ENDIF EOF() = .T.
141 ENDIF .NOT. (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
142 END DO WHILE .NOT. (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
143 *
144 SET COLOR TO W/B, W/B
145 @ 24,0 SAY SPACE(80)
146 *
147 DO WHILE .T.
148 SET COLOR TO R+/B, R+/B

332
APPENDIX B: MAINTENANCE MANUAL

Page 4

MANULDEL.PRG Program Listing

151 @ 6,47 SAY RECNO() PICT "9999"
152 STORE FEATURENO TO MFEAT
153 SELECT 2
154 USE DESCRIP INDEX DESCRIP.NDX
155 FIND &MFEAT
156 STORE CLIN TO MCLIN
157 STORE DESCRIPT TO MDESCIPT
158 SELECT 1
159 SET COLOR TO /BR, /BR
160 @ 9,45 SAY SITEJO PICT '99'
161 @ 12,45 SAY FEATURENO PICT '99999'
162 @ 13,45 SAY MCLIN PICT "9999"
163 @ 14,45 SAY MDESCIPT PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
164 @ 17,45 SAY MANULDESC PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
165 SET COLOR TO W/B, W/B
166 @ 24,0 SAY SPACE(80)
167 *
168 * ASK THE USER IF HE/SHE IS SURE ABOUT THE DELETION
169 *
170 * SET COLOR TO W+/B, W+/B
171 @ 20,06 SAY MESSAGE
172 SET COLOR TO R+/B, R+/B
173 @ 20,58 SAY "Y"
174 @ 20,65 SAY "N"
175 STORE "N" TO ACCEPT
176 @ 20,70 GET ACCEPT PICT ","
177 READ
178 *
179 * ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
180 *
181 * DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
182 * IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
183 * SET COLOR TO W+/R, W+/R
184 @ 24,24 SAY "Response must be either N or Y"
185 * DO DELAY
186 * STORE "N" TO ACCEPT
187 ENDIF
188 SET COLOR TO R+/B, R+/B
189 @ 20,70 GET ACCEPT PICT ","
190 READ
191 ENDDO
192 SET COLOR TO W/B, W/B
193 @ 20,05 SAY SPACE(70)
194 *
195 * IF ENTRIES ARE CORRECT, DELETE THEM FROM THE DATABASE.
196 * IF NOT RECOVER THEM
197 *
198 IF ACCEPT = "Y"
199 DELETE
200 ENDIF
SET COLOR TO R+/B, R+/B
STORE "C" TO CHOICE
@ 22,58 GET CHOICE PICT ":";
READ
*
ENSURE THAT THE USER'S RESPONSE IS EITHER "C" OR "X"
*
DO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
  IF .NOT. (CHOICE = "C" .OR. CHOICE = "X") THEN
    @ 24,24 SAY " Response must be either C or X "
    DO DELAY
    STORE "C" TO CHOICE
  ENDIF
  SET COLOR TO R+/B, R+/B
  @ 22,58 GET CHOICE PICT ":";
  READ
ENDO
*
SKIP TO THE NEXT RECORD TO BE REVIEWED
*
IF CHOICE = "C" THEN
  IF RECOD() = LAST_REC THEN
    GO TOP
  ELSE
    SKIP
  ENDIF
ENDIF
*
USER HAS DECIDED TO EXIT THE REVIEW
*
IF CHOICE = "X"
  SET COLOR TO W+/R, W+/R
  @ 24,0
  @ 24,6 SAY PACKEM
  SET COLOR TO W/B, W/B
  PACK
  EXIT
ENDIF
*
ENDO WHILE .T.
*
RETURN TO CALLING PROGRAM.
*
RELEASE ALL LIKE M*, ACCEPT, CHOICE, ERROR, FEATURES, FIRST_REC;
LAST_REC, PACKEM, SITES
CLOSE DATABASES
RETURN
*******************************************************************************
**APPENDIX B: MAINTENANCE MANUAL**

Page 1

**MANULREV.PRG Program Listing**

---

1 | * PROCEDURE MANULREV.PRG
2 |
3 | * AUTHORS : LCDR EDWARD J. CASE, SC, USN
4 | *          : LCDR WINSTON H. BUCKLEY, SC, USN
5 | *          : LCDR ROBERT F. BRADO, USN
6 | *          : LCDR ROBERT L. BEARD III, SC, USN
7 | *
8 | * PURPOSE : TO ENABLE THE USER TO REVIEW ALL THE RECORDS IN
9 | *          : THE MANUAL DATABASE
10 |
11 | * INPUT FILES : MANUAL.DBF, MANULSIT.NDX
12 |
13 | * CALLED BY : MANULCMD.PRG
14 |
15 | * MODULES CALLED : DELAY.PRG
16 |
17 | * GLOBAL VARIABLE: HIITNUM, HIISITE, LOFNUM, LOSITE
18 |
19 | * LOCAL VARIABLES: ACCEPT, CHOICE, CURRENTNO, EOF, ERROR, FIRSTREC,
20 | *          : LAST_REC, NCLIN, MDESCIPT, MFEAT, MSITE, TOF
21 |
22 | * DATE LAST TIME MODIFIED = 26 DECEMBER 1985
23 |
24 | * CASE SELECTION = 4 REVIEW EXISTING MANUAL RECORDS
25 |
26 | SET ESCAPE OFF
27 | SET TALK OFF
28 | USE MANUAL
29 | GO TOP
30 | SET COLOR TO W+/B, W+/B, B
31 | CLEAR
32 | IF EOF() = .T. THEN
33 | SET COLOR TO W+/R, W+/R
34 | @ 13,25 SAY "The MANUALS Database is EMPTY!"
35 | DO DELAY
36 | RETURN
37 | ENDIF
38 | ?? FLASH + "S.MANUALS.SCR/
39 | @ 24,0 SAY SPACE(80)
40 | SET COLOR TO R+/R, R+/R
41 | @ 3,26 SAY "MANUAL REVIEW FORMAT"
42 | SELECT 1
43 | STORE "Enter 00 to start at TOF, 99 to start at EOF or a site number" +;
44 | "between " + LOGSITE + " and " + HISITE + " to message"
45 | SET COLOR TO /W, /W
46 | @ 24,0 SAY MESSAGE
47 | STORE '80' TO MSITE
48 | DO WHILE .NOT. ((MSITE = '00') .AND. MSITE <= HISITE) .OR. MSITE = '00')
49 | SET COLOR TO /BR, /BR
50 | STORE '00' TO MSITE

---

335
09,45 GET MSITE PICT '99'
READ
53 IF .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99') THEN
54 SET COLOR TO W/B, W/B
55 @ 24,0 SAY SPACE(80)
56 SET COLOR TO W+/R, W+/R
57 STORE ' Response must be between ' + LOST ' + ' and ' + ;
58 HISITE + ', Zero (00) or 99 ' TO ERROR
59 @ 24,13 SAY ERROR
60 DO DELAY
61 SET COLOR TO /W, /W
62 @ 24,0 SAY MESSAGE
63 LOOP
ELSE
65 IF (MSITE = '00' .OR. MSITE = '99') THEN
66 USE MANUAL
67 IF MSITE = '00' THEN
68 GO BOTTOM
69 STORE RECONO() TO LAST_REC
70 GO TOP
71 STORE RECONO() TO FIRST_REC
72 ELSE
73 GO TOP
74 STORE RECONO() TO FIRST_REC
75 GO BOTTOM
76 STORE RECONO() TO LAST_REC
77 ENDF MSITE = '00'
78 EXIT
ELSE USE MANUAL INDEX MANULSIT
GO TOP
FIND &MSITE
83 IF EOF() = .T. THEN
84 SET COLOR TO W/B, W/B
85 @ 24,0 SAY SPACE(80)
86 SET COLOR TO W+/R, W+/R
87 STORE " No records exist for site number " + MSITE + ;
88 " , try again " TO ERROR
89 @ 24,16 SAY ERROR
90 DO DELAY
91 SET COLOR TO /W, /W
92 @ 24,0 SAY MESSAGE
93 STORE '88' TO MSITE
94 LOOP
ELSE
96 EXIT
ELSE ENDF
98 ENDF
99 ENDF
100 ENDDO WHILE
APPENDIX B: MAINTENANCE MANUAL

Page 3

MANULREV.PRG Program Listing

101*  
102 STORE SPACE(10) + 'Enter "00 " to start at TOF or a six digit ' +;
103 'feature number' + SPACE(10) TO MESSAGE
104 IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
105 SET COLOR TO /W, /W
106 $ 24,0 SAY MESSAGE
107 DO WHILE .T.
108    SET COLOR TO /BR, /BR
109    STORE '00 ' TO MFEAT
110 $ 12,45 GET MFEAT PICT '999999'
111 READ
112 IF .NOT. ((MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM) .OR.;
113    MFEAT = '00 ') THEN
114    SET COLOR TO W/B, W/B
115 $ 24,0 SAY SPACE(80)
116    SET COLOR TO W+/R, W+/R
117    STORE ' Response must be between ' + LOFNUM + ' and ' +;
118    HIFNUM + ' or Zero (00) ' TO ERROR
119 $ 24,9 SAY ERROR
120 DO DELAY
121 SET COLOR TO /W, /W
122 $ 24,0 SAY MESSAGE
123 LOOP
124 ELSE
125 IF (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM) THEN
126    IF MFEAT = '99 ' THEN
127        SET COLOR TO W/B, W/B
128 $ 24,0 SAY SPACE(80)
129        SET COLOR TO W+/R, W+/R
130        STORE ' Response must be between ' + LOFNUM + ' and ' +;
131        HIFNUM + ' or Zero (00) ' TO ERROR
132 $ 24,9 SAY ERROR
133 DO DELAY
134 SET COLOR TO /W, /W
135 $ 24,0 SAY MESSAGE
136 LOOP
137 ENDIF MFEAT = '99 '
138 STORE MSITE + MFEAT TO MIKEY
139 USE MANUAL INDEX MANUSIT
140 GO TOP
141 FIND MIKEY
142 IF EOF() = .T. THEN
143 SET COLOR TO W/B, W/B
144 $ 24,0 SAY SPACE(80)
145 SET COLOR TO W+/R, W+/R
146 STORE " No record with feature number " + MIKEY +;
147    " exists, try again " TO ERROR
148 $ 24,12 SAY ERROR
149 DO DELAY
150 SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
LOOP
ELSE
EXIT
ENDIF EOF() = .T.
ELSE
GO TOP
EXIT
ENDIF (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
ENDIF
ENDIF .NOT. (MSITE >= LOSITE .OR. MSITE <= HISITE)
* SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
STORE "At beginning of records for site number " + MSITE + " TO TOP
STORE "At end of records for site number " + MSITE + " TO EOF
DO WHILE .T.,
SET COLOR TO R+/B, R+/B
@ 6,47 SAY RECNO() PICT "9999"
* CLEAR SCREEN AND SET INITIAL VALUES FOR VARIABLES TO BE
* ADDED TO THE FILE. THE M PREFIX INDICATES MEMORY VARIABLES
* DISTINGUISHING THEM FROM THEIR CORRESPONDING DATABASE FIELDS.
STORE FEATURENO TO MFEAT
SELECT 2
USE DESCRIP INDEX DESCRIP
FIND &MFEAT
STORE CLIN TO MCLIN
STORE DESCRIPT TO MDESCIP
SELECT 1
SET COLOR TO /BR, /BR
@ 09,45 SAY SITENO PICT "99"
@ 12,45 SAY FEATURENO PICT "999999"
@ 13,45 SAY MCLIN PICT "9999"
@ 14,45 SAY MDESCIP PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@ 17,45 SAY MANDESC PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
SET COLOR TO /BR, /BR
STORE "N" TO CHOICE
@ 22,67 GET CHOICE PICT "]"
READ
* ENSURE THAT THE USER'S RESPONSE IS EITHER "N", "P" OR "X"
DO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X")
IF .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X") THEN
APPENDIX B: MAINTENANCE MANUAL

Page 5

MANULREV.PRG Program Listing

201 | SET COLOR TO W+/R,W+/R
202 | @ 24,22 SAY " Response must be either N, P or X "
203 | DO DELAY
204 | STORE "N" TO CHOICE
205 | ENDIF
206 | SET COLOR TO R+/B,R+/B
207 | @ 22,67 GET CHOICE PICT ":"
208 | READ
209 | ENDDO
210 | * SKIP TO THE NEXT RECORD TO BE REVIEWED
211 | *
212 | IF CHOICE = "N" THEN
213 | IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
214 | SKIP
215 | IF EOF() = .T. THEN
216 | SKIP - 1
217 | SET COLOR TO W+/R, W+/R
218 | @ 24,21 SAY EOF
219 | DO DELAY
220 | ELSE
221 | IF .NOT. (SITENO = MSITE) THEN
222 | SKIP - 1
223 | SET COLOR TO W+/R, W+/R
224 | @ 24,21 SAY EOF
225 | DO DELAY
226 | ENDIF
227 | ENDIF EOF() = .T.
228 | ELSE
229 | IF RECNO() = LAST_REC THEN
230 | GO TOP
231 | ELSE
232 | SKIP
233 | ENDIF
234 | ENDIF
235 | ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE)
236 | ENDIF CHOICE = "N"
237 | *
238 | SKIP TO THE PREVIOUS RECORD
239 | *
240 | IF CHOICE = "P" THEN
241 | STORE RECNO() TO CURRENTNO
242 | IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
243 | SKIP - 1
244 | IF EOF() = .T. THEN
245 | GO TO CURRENTNO
246 | SET COLOR TO W+/R, W+/R
247 | @ 24,16 SAY TOP
248 | DO DELAY
249 | ELSE
250 | IF .NOT. (SITENO = MSITE) THEN
APPENDIX B: MAINTENANCE MANUAL

Page 6

MANULREV.PRG Program Listing

251 | SKIP
252 | SET COLOR TO W+/R, W+/R
253 | @ 24,16 SAY TOF
254 | DO DELAY
255 | ENDIF
256 | ENDIF BOF() = .T.
257 | ELSE
258 | IF RECNO() = FIRST_REC THEN
259 | GO BOTTOM
260 | ELSE
261 | SKIP - 1
262 | ENDIF
263 | ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE)
264 | ENDIF CHOICE = "p"
265 | *
266 | * USER HAS DECIDED TO EXIT THE REVIEW
267 | *
268 | IF CHOICE = "X"
269 | EXIT
270 | ENDIF
271 | *
272 | ENDDO WHILE .T.
273 | *
274 | * RETURN TO CALLING PROGRAM.
275 | *
276 | RELEASE ALL LIKE M*, ACCEPT, CHOICE, CURRENTNO, BOF, FIRST_REC, LAST_REC, TOF
277 | CLOSE DATABASES
278 | RETURN
279 | ******************************************************************************
* PROCEDURE MANULUPD.PRG

* AUTHORS
  LCDR EDWARD J. CASE, SC, USN
  LCDR WINSTON H. BUCKLEY, SC, USN
  LCDR ROBERT F. BRADO, USN
  LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE
  TO ENABLE THE USER TO UPDATE SELECTED RECORDS IN
  THE MANUAL DATABASE

* INPUT FILES
  MANUAL.DBF, MANULSIT.NDX

* CALLED BY
  MANULCMD.PRG

* MODULES CALLED
  DELAY.PRG

* GLOBAL VARIABLE:
  IIFNUM, HISITE, W+F, W+B, BSITE

* LOCAL VARIABLES:
  ACCEPT, ANS, CHOICE, EOF, ERROR, MCLIN, MDATE,
  MDESCIPT, MFEXT, MSITE, TOP

* DATE LAST TIME MODIFIED
  26 December 1985

* CASE SELECTION = 2
  UPDATE AN EXISTING MANUAL DESCRIPTION

SET ESCAPE OFF
SET TALK OFF
USE MANUAL
GO TOP
CLEAR
IF EOF() = .T. THEN
  SET COLOR TO W+/B, W+/B, B
  IF 13,25 SAY "The MANUALS database is empty!"
  DO DELAY
  RETURN
ENDIF
?? FLASH + "S.MANUALS.SCR"
& 24,0 SAY SPACE(80)
SET COLOR TO R+/R, R+/R
& 3,26 SAY "MANUAL UPDATE FORMAT"
SELECT 1
STORE "Enter 00 to start at TOF, 99 to start at EOF or a site number:"
"between " + MSITE + " and " + MSITE + " TO MESSAGE"
SET COLOR TO /W, /W
& 24,0 SAY MESSAGE
STORE '88' TO MSITE
DO WHILE .NOT. ((MSITE == '00') .AND. MSITE == MSITE) .OR. MSITE == '99'
SET COLOR TO /BR, /BR
STORE '00' TO MSITE
APPENDIX B: MAINTENANCE MANUAL

MANULUPD.PRG Program Listing

Page 2

@ 09.45 GET MSITE PICT '99'
READ
IF .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
STORE ' Response must be between ' + LOSITE + ' and ' + HISITE + ', Zero (00) or 99 ' TO ERROR
@ 24,13 SAY ERROR
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
LOOP
ELSE
IF (MSITE = '00' .OR. MSITE = '99') THEN
USE MANUAL
IF MSITE = '00' THEN
GO BOTTOM
STORE RECNO() TO LAST_REC
GO TOP
STORE RECNO() TO FIRST_REC
ELSE
GO TOP
STORE RECNO() TO FIRST_REC
GO BOTTOM
STORE RECNO() TO LAST_REC
ENDIF MSITE = '00'
EXIT
ELSE
USE MANUAL INDEX MANULSIT
GO TOP
FIND &MSITE
IF EOF() = .T. THEN
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
STORE " No records exist for site number " + MSITE + " try again " TO ERROR
@ 24,16 SAY ERROR
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
STORE '88' TO MSITE
LOOP
ENDIF EOF() = .T.
ENDIF (MSITE = '00' .OR. MSITE = '99')
ENDIF .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
ENDIF WHILE .NOT. ((MSITE >= '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
STORE " At beginning of records for site number " +;

342
APPENDIX B: MAINTENANCE MANUAL

MANULUPD.PRG Program Listing

101 MSITE + " " TO TOF
102 STORE " At end of records for site number " + MSITE + " " TO EOF
103 STORE ' Enter "00 " to start at TOF or a six digit feature ' +
104 ' number (' + LOFNUM + ' - ' + HIFNUM + ') ' TO MESSAGE
105 IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
106 SET COLOR TO /W, /W
107 @ 24,0 SAY MESSAGE
108 DO WHILE .T.
109 SET COLOR TO /BR, /BR
110 STORE '00 ' TO MFEAT
111 @ 12,45 GET MFEAT PICT '999999'
112 READ
113 IF .NOT. ((MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM) .OR.;
114 MFEAT = '00 ') THEN
115 SET COLOR TO W/B, W/B
116 @ 24,0 SAY SPACE(80)
117 SET COLOR TO W+/R, W+/R
118 STORE ' Response must be between ' + LOFNUM + ' and ' +;
119 'HIFNUM + ' or Zero (00) ' TO ERROR
120 @ 24,9 SAY ERROR
121 DO DELAY
122 SET COLOR TO /W, /W
123 @ 24,0 SAY MESSAGE
124 LOOP
125 ELSE
126 IF (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM) THEN
127 IF MFEAT = '99 ' THEN
128 SET COLOR TO W/B, W/B
129 @ 24,0 SAY SPACE(80)
130 SET COLOR TO W+/R, W+/R
131 STORE ' Response must be between ' + LOFNUM + ' and ' +;
132 'HIFNUM + ' or Zero (00) ' TO ERROR
133 @ 24,9 SAY ERROR
134 DO DELAY
135 SET COLOR TO /W, /W
136 @ 24,0 SAY MESSAGE
137 LOOP
138 ENDIF MFEAT = '99 '
139 STORE MSITE + MFEAT TO MKEY
140 USE MANUAL INDEX MANUSIT
141 GO TOP
142 FIND &MKEY
143 IF EOF() = .T. THEN
144 SET COLOR TO W/B, W/B
145 @ 24,0 SAY SPACE(80)
146 SET COLOR TO W+/R, W+/R
147 STORE " No record with feature number " + MFEAT +;
148 " exists, try again " TO ERROR
149 @ 24,12 SAY ERROR
150 DO DELAY

343
MANULUPD.PRG Program Listing

151 SET COLOR TO /w, /w
152 @ 24,0 SAY MESSAGE
153 LOOP
154 ELSE
155 EXIT
156 ENDIF BOF() = .T.
157 ELSE
158 GO TOP
159 EXIT
160 ENDIF (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
161 ENDIF
162 ENDDO WHILE .T.
163 ENDIF ,NOT. (MSITE = LOSITE .OR. MSITE <= HISITE)
164 *
165 STORE SPACE(16) + 'Press "Page Down" key to terminate record update' +;
166 SPACE(16) TO MESSAGE
167 STORE 1 TO INTRO
168 DO WHILE .T.
169 *
170 * INFORM THE USER OF HOW TO TERMINATE THE UPDATE OF A RECORD
171 *
172 IF INTRO = 1 THEN
173 STORE 0 TO INTRO
174 ?? FLASH + 'W.MANULUPD/"
175 SET CONSOLE OFF
176 WAIT TO ANS
177 SET CONSOLE ON
178 ENDF
179 *
180 SET COLOR TO R+/B, R+/B
181 @ 6,47 SAY RECNO() PICT "9999"
182 SET COLOR TO /w, /w
183 @ 24,0 SAY MESSAGE
184 *
185 * STORING THE OLD RECORD TO A WORK RECORD AREA. THE M PREFIX
186 * INDICATES MEMORY VARIABLES DISTINGUISHING THEM FROM THEIR
187 * CORRESPONDING DATABASE FIELDS.
188 *
189 *
190 STORE FEATURENO TO MFEAT
191 STORE MANULDESC TO MNULDESC
192 *
193 SELECT 2
194 USE DESCRIP INDEX DESCRIP
195 FIND &MFEAT
196 STORE CLIN TO MCLIN
197 STORE DESCRIP TO MDESCRIP
198 SELECT 1
199 SET COLOR TO /Br, /Br
200 @ 09,45 SAY STRING PICT "99"
201 @ 12,45 SAY FEATURENO PICT "999999"
202 @ 13,45 SAY MCLIN PICT "9999"
203 @ 14,45 SAY MDESCPT PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
204 @ 17,45 GET MMANDESC PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
205 READ
206 SET COLOR TO W/B, W/B
207 @ 24,0 SAY SPACE(80)
208 *
209 IF .NOT. (MANLDESC = MMANDESC) THEN
210 *
211 * ASK THE USER IF HE/SHE DESIRES TO ACCEPT THE CHANGES.
212 *
213 SET COLOR TO W+/B, W+/B
214 @20,12 SAY "Do you want to accept the changes? (Yes or No):"
215 SET COLOR TO R+/R, R+/R
216 @20,49 SAY "Y"
217 @20,56 SAY "N"
218 STORE "N" TO ACCEPT
219 @20,62 GET ACCEPT PICT ":!"
220 READ
221 *
222 * ENSURE THAT THE USER'S PROMPT IS EITHER "Y" OR "N"
223 *
224 DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
225 IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
226 SET COLOR TO W+/R, W+/R
227 @ 24,24 SAY "Response must be either N or Y"
228 DO DELAY
229 STORE "N" TO ACCEPT
230 ENDIF
231 SET COLOR TO R+/R, R+/R
232 @20,62 GET ACCEPT PICT ":!"
233 READ
234 ENDDO
235 SET COLOR TO W/B, W/B
236 @ 20,10 SAY SPACE(60)
237 *
238 IF ACCEPT = "Y" THEN
239 REPLACE MANLDESC WITH MMANDESC
240 ELSE
241 SET COLOR TO /BR, /BR
242 @ 17,45 SAY MANLDESC PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
243 ENDIF
244 ENDIF
245 *
246 *
247 SET COLOR TO R+/B, R+/B
248 STORE "N" TO CHOICE
249 @ 22,67 GET CHOICE PICT ":!"
250 READ
ENSURE THAT THE USER'S RESPONSE IS EITHER "N", "P" OR "X"

DO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X")
    IF .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X") THEN
        SET COLOR TO W+/R, W+/R
        @ 24,22 SAY " Response must be either N, P or X "
        DO DELAY
        STORE "N" TO CHOICE
    ENDIF
    SET COLOR TO R+/B, R+/B
    @ 22,67 GET CHOICE PICT ":'
    READ
ENDDO

* SKIP TO THE NEXT RECORD TO BE REVIEWED

* IF CHOICE = "N" THEN
    IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
        SKIP
    ENDIF
    IF EOF() = .T. THEN
        SKIP - 1
        ENDIF EOF() = .T.
    ELSE
        IF (MSITE = DSITE .AND. MSITE <= HISITE) THEN
            SKIP 1
            IF BOF() = .T. THEN
                GO TO CURRENTNO
            ELSE
                ENDIF
        ELSE IF (LSITE >= MSITE AND MSITE <= HSITE) THEN
            ENDIF CHOICE = "N"
        ENDIF
        * SKIP TO THE PREVIOUS RECORD
    * IF CHOICE = "P" THEN
        STORE RECNO() TO CURRENTNO
        IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
            SKIP - 1
            IF EOF() = .T. THEN
                GO TO CURRENTNO
            ELSE
                ENDIF
        ENDIF

346
APPENDIX B: MAINTENANCE MANUAL  Page 210

MANULUPD.PRG Program Listing

301 | SET COLOR TO W+/R, W+/R
302 | @ 24,16 SAY TOF
303 | DO DELAY
304 | ELSE
305 | IF .NOT. (SITENO = MSITE) THEN
306 |   SKIP
307 |   SET COLOR TO W+/R, W+/R
308 |   @ 24,16 SAY TOF
309 |   DO DELAY
310 | ENDIF
311 | ENDIF BOF() = .T.
312 | ELSE
313 | IF RECNO() = FIRST_REC THEN
314 |   GO BOTTOM
315 | ELSE
316 |   SKIP - 1
317 | ENDIF
318 | ENDIF (MSITE >= I0SITE AND MSITE <= IIISITE)
319 | ENDIF CHOICE = "P"
320 | *
321 | * USER HAS DECIDED TO EXIT THE REVIEW
322 | *
323 | IF CHOICE = "X"
324 |   EXIT
325 |   ENDIF
326 | *
327 | ENDDO WHILE .T.
328 | *
329 | * RETURN TO CALLING PROGRAM.
330 | *
331 | RELEASE ALL LIKE M*, ACCEPT, ANS, CHOICE, ERROR
332 | CLOSE DATABASES
333 | RETURN
334 | *******************************************************************************
* PROCEDURE MKLABELS.PRG

* AUTHORS
  LCDR EDWARD J. CASE, SC, USN
  LCDR WINSTON H. BUCKLEY, SC, USN
  LCDR ROBERT F. BRADO, USN
  LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE
  PROVIDE THE USER WITH THE CAPABILITY OF RAPIDLY GENERATING MAILING LABELS FOR ALL OF THE SPLICE SITES.

* INPUT FILES
  : NONE.

* OUTPUT FILES
  : NONE.

* CALLED BY
  : MAINMENU.PRG

* MODULES CALLED
  : DELAY.PRG

* LOCAL VARIABLES
  : COPIES, IMAGE, INTRO, LABELS, LAST_LINE, LINECNT, MESSAGE, SKIPONE

* DATE LAST TIME MODIFIED
  : 27 DECEMBER 1985

* GENERATE MAILING LABELS FOR ALL OF THE SPLICE SITES.

SET ESCAPE OFF
SET EXACT ON
SET TALK OFF
CLEAR
?? FLASH + "S.MKLABELS.SCR/
@ 24,0 SAY SPACE(80)
SET COLOR TO R+/B, R+/B

* OBTAIN THE NUMBER OF SETS OF LABELS TO PRINT FROM THE USER
STORE SPACE(5) + "Input the number of sets of labels desired" +;
  "(Range 1 - 10) or 00 TO EXIT" + SPACE(5) TO MESSAGE

STORE "99" TO COPIES
DO WHILE .NOT. (COPIES >= "00" .AND. COPIES <= "10")
  SET COLOR TO /W, /W
  @ 24,0 SAY MESSAGE
  STORE "00" TO COPIES
  SET COLOR TO R+/B, R+/B
  @ 6,55 GET COPIES PICT "99"
  READ
  IF .NOT. (COPIES == '00' .AND. COPIES == '10')
    SET COLOR TO W/B, W/B

348
APPENDIX B: MAINTENANCE MANUAL

Page 2

MKLABELS.PRG Program Listing

51  @ 24,0 SAY SPACE(80)
52  SET COLOR TO W+/R, W+/R
53  @ 24,22 SAY "Response must be between 00 and 10"
54  DO DELAY
55  SET COLOR TO /W, /W
56  @ 24,0 SAY MESSAGE
57  LOOP
58  ENDIF
59  ENDDO
60  SET COLOR TO W/B, W/B
61  @ 24,0 SAY SPACE(80)
62  *
63  IF COPIES = "00" THEN
64      SET EXACT OFF
65      RELEASE COPIES, MESSAGE
66      RETURN
67  ENDIF
68  *
69  * START PRINTING LABELS
70  *
71  USE CONFIG INDEX CONFIG
72  GO 'UP
73  STORE SPACE(15) + " Performing printer alignment test for label forms" +;
74  SPACE(15) TO MESSAGE
75  STORE "Running label forms alignment print test" TO IMAGE
76  STORE "Y" TO CHOICE
77  STORE 1 TO INTRO
78  STORE 1 TO LINECNT
79  STORE LINECNT + 8 TO SKIPONE
80  *
81  * ASK THE USER IF A PRINTER ALIGNMENT TEST IS DESIRED
82  *
83  SET COLOR TO W+/B, W+/B
84  @ 22,9 SAY "Do you desire to run a printer alignment test? (Yes or No): "
85  SET COLOR TO R+/B, R+/B
86  @ 22,57 SAY "Y"
87  @ 22,64 SAY "N"
88  DO WHILE CHOICE = "Y"
89      SET COLOR TO R+/B, R+/B
90      @ 22,70 GET CHOICE PICT ","
91      READ
92  *
93  * ENSURE THAT THE USER'S PROMPT IS EITHER "Y" OR "N"
94  *
95  DO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "Y")
96      IF .NOT. (CHOICE = "N" .OR. CHOICE = "Y") THEN
97          SET COLOR TO W+/R, W+/R
98          @ 24,24 SAY "Response must be either N or Y"
99          DO DELAY
100     STORE "Y" TO CHOICE

349
APPENDIX B: MAINTENANCE MANUAL

Page 3

MKLABELS.PRG Program Listing

101 ENDIF .NOT. (CHOICE = "N" .OR. CHOICE = "Y")
102 SET COLOR TO R+/B ,R+/B
103 @ 22,70 GET CHOICE PICT ":!
104 READ
105 ENDDO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "Y")
106 *
107 IF CHOICE = "Y"
108 @ 22,70 SAY " "
109 *
110 * DISPLAY PRINTER INFORMATION WINDOW TO USER
111 *
112 IF INTRO = 1 THEN
113 STORE 0 TO INTRO
114 ?? FLASH + "W.LABELS/"
115 SET CONSOLE OFF
116 WAIT TO ANS
117 SET CONSOLE ON
118 ENDIF INTRO = 1
119 SET COLOR TO /W, /W
120 @ 24,0 SAY MESSAGE
121 SET COLOR TO /BR, /BR
122 @ 14,19 SAY IMAGE
123 @ 15,19 SAY IMAGE
124 @ 16,19 SAY IMAGE
125 @ 17,19 SAY IMAGE
126 @ 18,19 SAY IMAGE
127 @ 19,19 SAY IMAGE
128 SET DEVICE TO PRINT
129 DO WHILE LINECNT < SKIPONE
130 @ LINECNT,1 SAY IMAGE
131 LINECNT = LINECNT + 1
132 ENDDO WHILE LINECNT < SKIPONE
133 SKIPONE = LINECNT + 8
134 SET DEVICE TO SCREEN
135 SET COLOR TO W/B, W/B
136 @ 24,0 SAY SPACE(80)
137 ELSE
138 SET COLOR TO /BR, /BR
139 @ 14,19 SAY SPACE(40)
140 @ 15,19 SAY SPACE(40)
141 @ 16,19 SAY SPACE(40)
142 @ 17,19 SAY SPACE(40)
143 @ 18,19 SAY SPACE(40)
144 @ 19,19 SAY SPACE(40)
145 LOOP
146 ENDIF CHOICE = "Y"
147 ENDDO WHILE CHOICE = "Y"
148 *
149 * SKIP ONE BLANK LABEL PRIOR TO PRINTING SITE LABELS
150
APPENDIX B: MAINTENANCE MANUAL

Page 4

MKLABELS.PRG Program Listing

151 SET DEVICE TO PRINT
152 *
153 DO WHILE LINECNT < SKIPONE
154   @ LINECNT,1 SAY ""
155   LINECNT = LINECNT + 1
156 ENDDO WHILE LINECNT < SKIPONE
157 *
158 SET DEVICE TO SCREEN
159 *
160 SET COLOR TO W+/B, W+/B
161 @ 21,10 SAY SPACE (60)
162 *
163 DISPLAY PRINTER INFORMATION WINDOW TO USER
164 *
165 IF INTRO = 1 THEN
166   STORE 0 TO INTRO
167   ?? FLASH + "W.LABELS/"
168   SET CONSOLE OFF
169   WAIT TO ANS
170   SET CONSOLE ON
171 ENDF INTRO = 1
172 *
173 STORE SPACE(28) + "Printing Mailing Labels " + SPACE(28) TO MESSAGE
174 SET COLOR TO /W, /W
175 @ 24,0 SAY MESSAGE
176 *
177 DO WHILE .NOT. EOF()
178   STORE TRIM(SITECITY) + "", " + TRIM(SITESTATE) + " " +;
179    TRIM(SITEZIP) TO LAST_LINE
180   IF SITECO = " " THEN
181    LOOP
182    SKIP
183    ENDF SITECO = " "
184 @ 10,46 SAY SITECO PICT "99"
185 SET COLOR TO /R+, /R+
186 @ 15,19 SAY SITECO PICT "............................................."
187 @ 16,19 SAY SITENAMEFL PICT "............................................."
188 IF SITEADD1 > " " THEN
189   @ 17,19 SAY SITEADD1 PICT "............................................."
190   IF SITEADD2 > " " THEN
191      @ 18,19 SAY SITEADD2 PICT "............................................."
192      @ 19,19 SAY LAST_LINE
193 ENDIF SITEADD2 > " "
194 ELSE
195   @ 18,19 SAY LAST_LINE
196 ENDF SITEADD2 > " "
197 ELSE
198   @ 17,19 SAY LAST_LINE
199 ENDF SITEADD1 > " "
200 SET DEVICE TO PRINT
APPENDIX B: MAINTENANCE MANUAL

Page 5

MKLABELS.PRG Program Listing

201| STORE 0 TO LABELS
202| DO WHILE LABELS < VAL(COPIES)
203| @ LINECNT,1 SAY SPACE(40)
204| @ LINECNT+1,1 SAY SPACE(40)
205| @ LINECNT+2,1 SAY SITEADD PICT "!!!!!!!!!!!!!!!!!!!!!!"
206| @ LINECNT+3,1 SAY SITENAMEFL
207| PICT "!!!-----------------------!!"  
208| IF SITEADD1 > " " THEN
209| @ LINECNT+4,1 SAY SITEADD1
210| PICT "!!!!!!!!!!!!!!!!!!!!!!"  
211| IF SITEADD2 > " " THEN
212| @ LINECNT+5,1 SAY SITEADD2
213| PICT "!!!!!!!!!!!!!!!!!!!!!!"
214| @ LINECNT+6,1 SAY LAST_LINE
215| @ LINECNT+7,1 SAY SPACE(40)
216| ELSE
217| @ LINECNT+5,1 SAY LAST_LINE
218| @ LINECNT+6,1 SAY SPACE(40)
219| @ LINECNT+7,1 SAY SPACE(40)
220| ENDIF SITEADD2 > " \\
221| ELSE
222| @ LINECNT+4,1 SAY LAST_LINE
223| @ LINECNT+5,1 SAY SPACE(40)
224| @ LINECNT+6,1 SAY SPACE(40)
225| @ LINECNT+7,1 SAY SPACE(40)
226| ENDIF SITEADD1 > " \\
227| LINECNT = LINECNT+8
228| IF LINECNT > 81 THEN
229| LINECNT = 1
230| ENDIF LINECNT > 81
231| LABELS = LABELS + 1
232| ENDDO WHILE LABELS < COPIES
233| SET DEVICE TO SCREEN
234| @ 14,19 SAY SPACE(40)
235| @ 15,19 SAY SPACE(40)
236| @ 16,19 SAY SPACE(40)
237| @ 17,19 SAY SPACE(40)
238| @ 18,19 SAY SPACE(40)
239| @ 19,19 SAY SPACE(40)
240| SKIP
241| ENDDO WHILE .NOT. BUF()
242| * RETURN TO THE CALLING PROGRAM
243| *  
245| SET EXACT OFF
246| RELEASE COPIES, IMAGE, INTRO, LABELS, LAST_LINE, LINECNT,;
247| MESSAGE, SKIPONE
248| CLOSE DATABASES
249| RETURN
250| ******************************************************
* PROCEDURE MNLSTRPT.PRG

* AUTHORS
  : LCDR EDWARD J. CASE, SC, USN
  : LCDR WINSTON H. BUCKLEY, SC, USN
  : LCDR ROBERT F. BRADO, USN
  : LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE
  : PROVIDE THE USER A SPLICE MANUAL SITE LEVEL REPORT.

* INPUT FILES
  : MANUAL.DBF, TEMPONE.DBF, DESCRIPT.DBF,
  : DESCRIP.NDX

* OUTPUT FILES
  : NONE.

* CALLED BY
  : SITERPTS.PRG

* MODULES CALLED
  : DELAY.PRG

* GLOBAL VARIABLE
  : HISITE, IOSITE

* LOCAL VARIABLES
  : ACCEPT, CHOICE, ERROR, LINECT, MSITE, PAGENO,
  : TODAY, TODATE

* DATE LAST TIME MODIFIED
  : 27 DECEMBER 1985

* CASE SELECTION = 2 MANUAL SITE LEVEL REPORT

* CREATE THE SPLICE MANUAL SITE REPORT AND CHECK IF THE REPORT IS TO BE PRINTED OR DISPLAYED ON THE SCREEN.

SET ESCAPE OFF
SET TALK OFF
SET COLOR TO W+/B, W+/B, B
CLEAR
USE MANUAL
GO TOP
IF EOF() = .T. THEN
  SET COLOR TO W+/R, W+/R
  @ 13,25 SAY "The MANUALS Database is EMPTY!"
  DO DELAY
  RETURN
ENDIF
?? FLASH + "S.REPORTS.SCR/"
@ 24,0 SAY SPACE(80)
@ 2,27 SAY "SITE LEVEL MANUAL REPORT"
* ENSURE THAT TEMPORARY DATABASE AND INDEX DO NOT EXIST, IF SO ERASE THEM
SET CONSOLE OFF
ERASE TEMPONE.DBF
ERASE TEMPONE.NDX
SET CONSOLE ON
* SET COLOR TO W+/BR, W+/BR
@ 13,15 SAY "Enter site number for which the report is desired:"
USE MANUAL INDEX MANUSIT
* DO WHILE .T.
SET COLOR TO /BR, /BR
STORE LOSITE TO MSITE
@ 13,66 GET MSITE PICT '99'
READ
IF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
  SET COLOR TO W+/R, W+/R
  STORE ' Response must be between ' + LOSITE +;
    ' and ' + HISITE + ' TO ERROR
@ 24,22 SAY ERROR
DO DELAY
DO LOOP
ELSE
GO TOP
FIND &MSITE
IF EOF() = .T. THEN
  STORE " No manuals exist for site " + MSITE +;
    " try another site " TO MESSAGE
  SET COLOR TO W+/R, W+/R
  @ 24,16 SAY MESSAGE
  DO DELAY
  DO LOOP
ELSE
  EXIT
ENDIF EOF() = .T.
ENDIF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE)
ENDDO WHILE .T.
* SET COLOR TO /BR, /BR
@ 13,15 SAY SPACE(55)
* SET COLOR TO R+/ R+/R
@ 13,13 SAY " CREATING THE TEMPORARY DATABASE AND ASSOCIATED INDEX."
* CREATE THE TEMPORARY DATABASE TO BE USED
SET CONSOLE OFF
COPY STRUCTURE TO TEMPONE
USE TEMPONE
APPEND FROM MANUAL FOR SITE=MSITE
INDEX ON FEATURENO TO TEMPONE
APPENDIX B: MAINTENANCE MANUAL

Page 3

MNLSTRPT.PRG Program Listing

101 SET CONSOLE ON
102 * CREATE THE SPLICE EQUIPMENT PROJECT REPORT AND CHECK IF THE REPORT
103 * IS TO BE PRINTED OR DISPLAYED ON THE SCREEN.
104 * SET COLOR TO /BR, /BR
105 @ 13,12 SAY SPACE(65)
106 SET COLOR TO W+/BR, W+/BR
107 @ 13,16 SAY " Do you want a printed report? (Yes or No): 
108 SET COLOR TO /BR, /BR
109 @ 13,49 SAY "Y"
110 @ 13,56 SAY "N"
111 STORE "N" TO ACCEPT
112 @ 13,62 GET ACCEPT PICT "!
113 READ
114 * ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
115 * DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
116 IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
117 SET COLOR TO W+/R, W+/R
118 @ 24,24 SAY " Response must be either N or Y "
119 DO DELAY
120 STORE "N" TO ACCEPT
121 ENDIF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
122 SET COLOR TO /BR, /BR
123 @ 13,62 GET ACCEPT PICT "!
124 READ
125 ENDDO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
126 * SET COLOR TO /BR, /BR
127 @ 13,12 SAY SPACE(65)
128 * SELECT 1
129 USE TEMPONE
130 SELECT 2
131 USE DESCRIPT INDEX DESCRIP
132 SELECT TEMPONE
133 SET RELATION TO FEATURENO INTO DESCRIP
134 GO TOP
135 *
136 IF ACCEPT = "Y" THEN
137 ?? FLASH "W.PRINTER/"
138 SET CONSOLE OFF
139 WAIT TO CHOICE
140 SET CONSOLE ON
141 SET COLOR TO W/B, W/B
APPENDIX B: MAINTENANCE MANUAL

Page 219

MNLSTRPT.PRG Program Listing

201 | SET COLOR TO GR+/B, GR+/B
202 | @ 5,2 SAY "SITE CLIN FEATURE# DESCRIPTION"
203 | @ 5,57 SAY "MANUAL DESCRIPTION"
204 | SET COLOR TO /BR, /BR
205 | STORE 0 TO LINECT
206 | *
207 | DO WHILE .NOT. EOF() 
208 | DO WHILE LINECT < 15
209 |   @ LINECT+7,3 SAY SITENO
210 |   @ LINECT+7,8 SAY DESCRIP->CLIN
211 |   @ LINECT+7,16 SAY FEATURENO
212 |   @ LINECT+7,25 SAY DESCRIP->DESCRIPT
213 |   @ LINECT+7,54 SAY MANLDESC
214 |   LINECT = LINECT + 1
215 |   SKIP
216 |   IF EOF() = .T. THEN
217 |     SET COLOR TO W+/R, W+/R
218 |     @ 24,18 SAY " End of File reached, Press any key to EXIT "
219 |     SET CONSOLE OFF
220 |     WAIT TO ACCEPT
221 |     SET CONSOLE ON
222 |     EXIT
223 |   ENDIF EOF() = .T.
224 | ENDDO WHILE LINECT < 15
225 | *
226 | IF EOF() = .T. THEN
227 |   EXIT
228 | ENDIF EOF() = .T.
229 | SET COLOR TO R+/B, R+/B
230 | STORE "C" TO CHOICE
231 | @ 22,57 GET CHOICE PIC ""!
232 | READ
233 | *
234 | ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
235 | *
236 | DO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
237 |   IF .NOT. (CHOICE = "C" .OR. CHOICE = "X") THEN
238 |     SET COLOR TO W+/R, W+/R
239 |     @ 24,24 SAY " Response must be either C or X "
240 |     DO DELAY
241 |     STORE "C" TO CHOICE
242 |   ENDIF .NOT. (CHOICE = "C" .OR. CHOICE = "X")
243 | SET COLOR TO R+/B, R+/B
244 | @ 22,57 GET CHOICE PIC "!"
245 | READ
246 | ENDDO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
247 | *
248 | DETERMINE IF THE USER WANTS TO QUIT OR CONTINUE
249 | *
250 | IF CHOICE = "C"
APPENDIX B: MAINTENANCE MANUAL

Page 220

Page 6

MNLSTRPT.PRG Program Listing

251      SET COLOR TO /BR, /BR
252      @ 07,2 SAY SPACE(76)
253      @ 08,2 SAY SPACE(76)
254      @ 09,2 SAY SPACE(76)
255      @ 10,2 SAY SPACE(76)
256      @ 11,2 SAY SPACE(76)
257      @ 12,2 SAY SPACE(76)
258      @ 13,2 SAY SPACE(76)
259      @ 14,2 SAY SPACE(76)
260      @ 15,2 SAY SPACE(76)
261      @ 16,2 SAY SPACE(76)
262      @ 17,2 SAY SPACE(76)
263      @ 18,2 SAY SPACE(76)
264      @ 19,2 SAY SPACE(76)
265      @ 20,2 SAY SPACE(76)
266      @ 21,2 SAY SPACE(76)
267      STORE 0 TO LINECT
268      ELSE EXIT
269      ENDF IF CHOICE = "C"
270      * ENDOO WHILE .NOT. EOF()
271      * ENDF ACCEPT = "Y"
272      * ERASE THE TEMPORARY DATABASE AND ASSOCIATED INDEX USED FOR TOTALS
273      * CLOSE DATABASES
274      SET CONSOLE OFF
275      ERASE TEMPONE.DBF
276      ERASE TEMPONE.NDX
277      SET CONSOLE ON
278      SET PRINT OFF
279      * RETURN TO CALLING PROGRAM
280      * RELEASE ACCEPT, CHOICE, ERROR, LINECT, MSITE, PAGENO, TODAY, TOTDATE
281      RETURN
282
283 *******************************
**NEWDOADD.PRG Program Listing**

1. **PROCEDURE NEWDOADD.PRG**

2. **AUTHORS**
   - LCDR EDWARD J. CASE, SC, USN
   - LCDR WINSTON H. BUCKLEY, SC, USN
   - LCDR ROBERT F. BRADO, USN
   - LCDR ROBERT L. BEARD III, SC, USN

3. **PURPOSE**
   - To add a new delivery order to the existing equipment, manual, and serial number databases while updating all indexes. No TEMP.DBF line item with a blank or "XXXXXX" feature number will be added to the file.

4. **INPUT FILES**
   - EQUIP.DBF, MANUAL.DBF, SERIALNO.DBF, TEMP.DBF
   - NEWDOMOD.DBF, SERNOTP.DBF

5. **OUTPUT FILES**
   - EQUIP.DBF, MANUAL.DBF, SERIALNO.DBF, TEMP.DBF
   - NEWDOMOD.DBF, SERNOTP.DBF

6. **CALLED BY**
   - NEWDOCVT.PRG

7. **MODULES CALLED**
   - SERNOBLD.PRG

8. **LOCAL VARIABLES**: MESSAGE, MFENT, MSITE, MINDEX

9. **DATE LAST TIME MODIFIED**
   - 22 DECEMBER 1985

10. **CLEAR SCREEN. COPY AND MODIFY INPUT FILE TO DATABASE FORMAT FOR ALL RECORDS THAT DON'T HAVE A BLANK OR "XXXXXX" IN THE FEATURE NUMBER.**

11. **SET COLOR TO R+/R+, R+/2**
12. # 15,24 SAY "UPDATING THE EQUIPMENT DATABASE"
13. "Adding new records to the EQUIPMENT database" +;
14. ", PLEASE BE PATIENT" TO MESSAGE
15. **SET COLOR TO /W, /W**
16. # 24,0 SAY MESSAGE
17. **USE EQUIP**
18. **COPY STRUCTURE TO NEWDOMOD**
19. **USE NEWDOMOD**
20. **APPEND FROM TEMP.DBF FOR FEATURENO <> ' ' .AND. FEATURENO <> 'XXXXXX'**
21. **FILL-IN THE EFFECTIVE DELIVERY ORDER DATE FIELD WITH THE DATE SUPPLIED**
22. **BY THE USER AND ADD THE DELIVERY ORDER TO THE EQUIPMENT DATABASE.**
23. **REPLACE ALL ENDATE WITH MEFFDATE**
24. **USE EQUIP INDEX EQUIPDAT, EQUIPSIT, EQUIPPRJ, EQUIPSD, MEFF**
25. **APPEND FROM NEWDOMOD**
26. **ADD THE NEW RECORDS FOR THE MANUAL DATABASE.**
27. **SET COLOR TO R+/R+/R+**
APPENDIX B: MAINTENANCE MANUAL

NEWDOADD.PRG Program Listing

51 @ 15,24 SAY " UPDATING THE MANUALS DATABASE "
52 STORE " Adding new records to the MANUAL database" +;
53 ", PLEASE BE PATIENT " TO MESSAGE
54 SET COLOR TO /W, /W
55 @ 24,0 SAY MESSAGE
56 CLOSE DATABASES
57 SELECT 1
58 USE MANUAL INDEX MANULSIT
59 SELECT 2
60 USE NEWDOMOD
61 *
62 DO WHILE .NOT. EOF()
63 STORE SITENO TO MSITE
64 STORE FEATURENO TO MFEAT
65 STORE SITENO + FEATURENO TO MINDEX
66 SELECT 1
67 GO TOP
68 FIND &MINDEX
69 IF BOF()
70 GO BOTTOM
71 INSERT BLANK
72 REPLACE FEATURENO WITH "&MFEAT"
73 REPLACE SITENO WITH "&MSITE"
74 ENDF
75 SELECT 2
76 SKIP
77 ENDDO WHILE .NOT. EOF()
78 *
79 * BUILDING A DUMMY SERIAL NUMBER FILE WHICH WILL BE MODIFIED AND
80 * EXPANDED WHEN ALL DELIVERY ORDERS HAVE BEEN LOADED.
81 *
82 SET COLOR TO R+/ , R+/,
83 @ 15,24 SAY " BUILDING THE SERIAL NUMBER FILE ";
84 STORE " Adding new records to the SERIAL NUMBER database," +;
85 " PLEASE BE PATIENT " TO MESSAGE
86 SET COLOR TO /W, /W
87 @ 24,0 SAY MESSAGE
88 USE SERIALNO
89 COPY STRUCTURE TO SERNOIMP
90 USE SERNOIMP
91 APPEND FROM NEWDOMOD
92 *
93 * CALL THE PROGRAM TO BUILD THE BLANK SERIAL NUMBER RECORDS
94 *
95 DO SERNOBDL
96 STORE " Appending new records to the database may be a long process," +;
97 " PLEASE BE PATIENT " TO MESSAGE
98 SET COLOR TO /W, /W
99 @ 24,0 SAY MESSAGE
100 SET COLOR TO R+, R+/

359
APPENDIX B: MAINTENANCE MANUAL

Page 3

NEWDOADD.PRG Program Listing

101    @ 15,12 SAY " APPENDING NEW RECORDS TO THE SERIAL NUMBER DATABASE "
102    USE SERIALNO INDEX SERNOPRJ, SERNOSIT, SERNODAT, SERNOFEA
103    APPEND FROM SERNOINP
104    SET COLOR TO W/B, W/B
105    @ 15,10 SAY SPACE(65)
106    @ 24,0 SAY SPACE(80)
107    *       RETURNING TO THE CALLING PROGRAM.
108    *       
109    CLOSE DATABASES
110    RETURN
111    ***********************************************************************************
NEWDOCMD.PRG Program Listing

* PROCEDURE NEWDOCMD.PRG

* AUTHORS
  LCDR EDWARD J. CASE, SC, USN
  LCDR WINSTON H. BUCKLEY, SC, USN
  LCDR ROBERT F. BRADO, USN
  LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE: LOAD NEW DELIVERY ORDERS TO THE DATABASE FILES.

* INPUT FILES: NONE.

* OUTPUT FILES: NONE.

* CALLED BY: MAINMENU.PRG.

* MODULES CALLED: NEWDOCVT.PRG

* LOCAL VARIABLES: SELEKT

* DATE LAST TIME MODIFIED: 22 DECEMBER 1985

DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR SELECTION

STORE "1" TO SELEKT
DO WHILE SELEKT < "2"
  SET COLOR TO W/B, W/B
  CLEAR
  ?? FLASH + "W.NEWDOCMD/"
  SET CONSOLE OFF
  WAIT TO SELEKT
  SET CONSOLE ON

PROCESS ROUTINE BASED ON THE USER'S SELECTION.

DO CASE
  CALL THE NEW DELIVERY ORDER CONVERT AND LOAD PROGRAM.
  CASE SELEKT = "1"
    DO NEWDOCVT
  RETURN TO THE MAINMENU PROGRAM.
  CASE SELEKT = "2"
  END CASE
ENDDO (WHILE SELEKT = "2")
RETURN TO THE CALLING PROGRAM
RETURN
APPENDIX B: MAINTENANCE MANUAL

Page 2

NEWDOCMD.PRG Program Listing

51| ******************************************

362
* PROCEDURE NEWDOCVT.PRG

* AUTORS : LCDR EDWARD J. CASE, SC, USN
* LCDR WINSTON H. BUCKLEY, SC, USN
* LCDR ROBERT F. BRADO, USN
* LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : TO COMPARE AN INCOMING NEW DELIVERY ORDER TO THE
* EXISTING EQUIPMENT DATABASE AND CHECK FOR DUPLICATE
* SITE NUMBER AND DELIVERY ORDER DATE. IF THE SITE
* NUMBER AND DELIVERY DATE ARE UNIQUE OR THE USER
* DECIDES TO LOAD THE DUPLICATE SITE NUMBER/DELIVERY
* ORDER ANYWAY THEN THE NEW DELIVERY ORDER ADD
* PROGRAM IS CALLED. WHEN ALL DELIVERY ORDERS ARE
* ADDED THEN SPECIFIED INDEXES ARE UPDATED.

* INPUT FILES : TED.DBF, EQUIP.DBF, MANUAL.DBF, SERIALNO.DBF, NEW
* DELIVERY ORDER .PRN FILE, EFFDATE.NDX, EQUIPSIT.NDX,
* EQUIPPRJ.NDX, MANUJSIT.NDX, SERNOPRJ.NDX, SERNOSIT.NDX,
* SERNODAT.NDX, NEWDOCMD.DBF, TEMP.DBF, SERNOUP.DBF,
* EFEAT.NDX

* OUTPUT FILES : EQUIP.DBF, MANUAL.DBF, SERIAL.DBF, EFFDATE.NDX,
* EQUIPSIT.NDX, EQUIPPRJ.NDX, MANUJSIT.NDX, EFEAT.NDX
* SERNOPRJ.NDX, SERNOSIT.NDX, SERNODAT.NDX.

* CALLED BY : NEWDOCMO.DRG

* MODULES CALLED : NEWDOCMO.DRG, DELAY.DRG

* GLOBAL VARIABLE: H1IDATE, HISITE, LODATE, LOSITE

* LOCAL VARIABLES: ACCEPT, CHOICE, DBNAME, ERASIT, ERROR, MDAY, MEFFDATE,
* MESSAGE, MKEY, MONWHI, MULDATE, MSITE, MYEAR, MOFILE

* DATE LAST TIME MODIFIED =========> 22 DECEMBER 1985 <========

* SET UP INITIAL STRUCTURE AND RECEIVE INPUT INFORMATION.
* AND START LOOP PROCESS.

SET ESCAPE OFF
SET TALK OFF
SET COLOR TO W+/B, W+/B, B
?? FLASH "S.NEWDOCVT.SCR/
24,0 SAY SPACE(80)
STORE "Are all input entries correct? (Yes or No):" TO CORRECT
DO WHILE .T.
* GET THE INPUT VALUES FROM THE USER

363
DO WHILE .T.
    STORE SPACE(18) + "Enter the name of the .PRN file to be loaded" +;
    SPACE(18) TO MESSAGE
    SET COLOR TO /W, /W
    @ 24,0 SAY MESSAGE
    STORE "SPICE " TO DBNAME
    STORE DTOC(DATE()) TO SYSDATE
    STORE SUBSTR(SYSDATE,7,2) + SUBSTR(SYSDATE,1,2) +;
    SUBSTR(SYSDATE,4,2) TO MEFFDATE
    STORE "01" TO MSITE
    SET COLOR TO /BR, /BR
    a 24,54 SAY "File does not exist, try again"
    DO DELAY
    NOFILE = NOFILE + 1
    IF NOFILE = 3 THEN
        SET COLOR TO W+/BG, W+/BG
        @ 17,15 SAY " Do you want to exit this process? (Yes or No): "
        SET COLOR TO /BG, /BG
        a 17,51 SAY "Y"
        a 17,58 SAY "N"
        STORE "Y" TO ACCEPT
        a 17,3 GET ACCEPT PICT "!
        READ
        DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
            IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
                SET COLOR TO W+/R, W+/R
                @ 24,24 SAY " Response must be either N or Y "
                DO DELAY
                STORE "Y" TO ACCEPT
            ENDIF
        SET COLOR TO /BG, /BG
        a 17,63 GET ACCEPT PICT "!
        READ
        ENDDO
    IF ACCEPT = "Y" THEN
        SET CONSOLE OFF
        CLOSE DATABASES
        ERASE TEMP.DBF
        ERASE NEWLMOV.DBF
        ERASE SERNOTMP.DBF
        SET CONSOLE ON
        RELEASE ALL LIKE M*, ACCEPT, CHOOSE, CORRECT, DEADMIL
RETURN
ELSE
NOFILE = 0
ENDIF
SET COLOR TO W+/B, W+/B
@ 17,10 SAY SPACE(55)
ENDIF
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
STORE "SPLICE " TO DBNAME
SET COLOR TO /BR, /BR
@ 6,54 GET DBNAME PICT "!!!!!!!"
READ
ENDDO
STORE TRIM(DBNAME) + ".PRN" TO DBNAME
USE TED
COPY TO TEMP.DBF
USE TEMP.DBF
APPEND FROM &DBNAME SDF
GO TOP
*
HAVE THE USER SPECIFY THE EFFECTIVE DATE OF THE DELIVERY ORDER
*
STORE SPACE(17) + "Input Effective Date (Range " + LODEATE +;"
" to " + HIDEATE + ")" + SPACE(17) TO MESSAGE
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
STORE "000000" TO MOLDATE
DO WHILE .NOT. (MOLDATE >= LODEATE .AND. MOLDATE <= HIDEATE)
STORE MEFFDATE TO MOLDATE
SET COLOR TO /BR, /BR
@ 8,54 GET MOLDATE PICT "999999"
READ
DO WHILE .T.
IF .NOT. (SUBSTR(MOLDATE,1,2) > "83" .AND.;
SUBSTR(MOLDATE,1,2) <= "99") THEN
SET COLOR TO W/B, W/B
@ 24,0 SAY SPACE(80)
SET COLOR TO W+/R, W+/R
@ 24,16 SAY " Year portion of date must be between 84 and 99 ">
DO DELAY
SET COLOR TO /W, /W
@ 24,0 SAY MESSAGE
STORE SUBSTR(MEFFDATE,1,2) TO MYEAR
SET COLOR TO /BR, /BR
@ 8,54 GET MYEAR PICT "99"
READ
STORE MYEAR + SUBSTR(MOLDATE,3,4) TO MOLDATE
LOOP

365
APPENDIX B: MAINTENANCE MANUAL

NEWDOCVT.PRG Program Listing

151 | ELSE
152 | EXIT
153 | ENDIF
154 | ENDDO WHILE .T.
155 |
156 | DO WHILE .T.
157 | IF NOT. (SUBSTR(MOLDATE,3,2) >= "01" .AND.;
158 | SUBSTR(MOLDATE,3,2) <= "12") THEN
159 | SET COLOR TO W/B, W/B
160 | @ 24,0 SAY SPACE(80)
161 | SET COLOR TO W+/R, W+/R
162 | @ 24,16 SAY " Month portion of date must be between 01 and 12 "
163 | DO DELAY
164 | SET COLOR TO /W, /W
165 | @ 24,0 SAY MESSAGE
166 | STORE SUBSTR(MEFFDATE,3,2) TO MMNTH
167 | SET COLOR TO /BR, /BR
168 | @ 8,56 GET MMNTH PICT "99"
169 | READ
170 | STORE SUBSTR(MOLDATE,1,2) + MMNTH +;
171 | SUBSTR(MOLDATE,5,2) TO MOLDATE
172 | LOOP
173 |
174 | ELSE
175 | EXIT
176 | ENDIF
177 | ENDDO WHILE .T.
178 |
179 | DO WHILE .T.
180 | IF ((SUBSTR(MOLDATE,3,2) = "04" .OR. SUBSTR(MOLDATE,3,2) = "06" .OR.;
181 | SUBSTR(MOLDATE,3,2) = "09" .OR. SUBSTR(MOLDATE,3,2) = "11") .AND.;
182 | .NOT. (SUBSTR(MOLDATE,5,2) >= "01" .AND. SUBSTR(MOLDATE,5,2) <= "30")) THEN
183 | SET COLOR TO W/B, W/B
184 | @ 24,0 SAY SPACE(80)
185 | SET COLOR TO W+/R, W+/R
186 | @ 24,16 SAY " Day portion of date must be between 01 and 30 "
187 | DO DELAY
188 | SET COLOR TO /W, /W
189 | @ 24,0 SAY MESSAGE
190 | STORE SUBSTR(MEFFDATE,5,2) TO MDAY
191 | SET COLOR TO /BR, /BR
192 | @ 8,56 GET MDAY PICT "99"
193 | READ
194 | STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
195 | LOOP
196 |
197 | ELSE
198 | IF (SUBSTR(MOLDATE,3,2) = "02" .AND. .NOT.;
199 | (SUBSTR(MOLDATE,5,2) >= "01" .AND.;
200 | SUBSTR(MOLDATE,5,2) <= "28")) THEN
201 | SET COLOR TO W/B, W/B
APPENDIX B: MAINTENANCE MANUAL

NEWDOCVT.PRG Program Listing

201 @ 24,0 SAY SPACE(80)
202 SET COLOR TO W/R, W/R
203 @ 24,16 SAY " Day portion of date must be between 01 and 28 "
204 DO DELAY
205 SET COLOR TO /W, /W
206 @ 24,0 SAY MESSAGE
207 STORE SUBSTR(MEFDATE,5,2) TO MDAY
208 SET COLOR TO /BR, /BR
209 @ 8,58 GET MDAY PICT "99"
210 READ
211 STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
212 LOOP
213 ELSE
214
215 IF .NOT. (SUBSTR(MOLDATE,5,2) >= "01") .AND.;
216 SUBSTR(MOLDATE,5,2) <= "31") THEN
217 SET COLOR TO W/B, W/B
218 @ 24,0 SAY SPACE(80)
219 SET COLOR TO W/R, W/R
220 @ 24,16 SAY " Day portion of date must be between 01 and 31 "
221 DO DELAY
222 SET COLOR TO /W, /W
223 @ 24,0 SAY MESSAGE
224 STORE SUBSTR(MEFDATE,5,2) TO MDAY
225 SET COLOR TO /BR, /BR
226 @ 8,58 GET MDAY PICT "99"
227 READ
228 STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
229 LOOP
230 ELSE
231 EXIT
232 ENDF
233 ENDF
234 ENDF
235 ENDDO WHILE .T.
236 ENDDO WHILE .NOT. (MOLDATE >= LODATE .AND. MOLDATE <= HIDATE)
237
238 STORE MOLDATE TO MEFDATE
239 SET COLOR TO W/B, W/B
240 @ 24,0 SAY SPACE(80)
241 STORE SPACE(8) + "Enter site number of Delivery Order to be " +
242 "loaded to the database" + SPACE(8) TO MESSAGE
243 SET COLOR TO /W, /W
244 @ 24,0 SAY MESSAGE
245 SET COLOR TO /BR, /BR
246 @ 11,54 SAY SITENO PICT "99"
247 @ 13,54 GET MSITE PICT "99"
248 READ
249 DO WHILE .NOT. (MSITE >= LOSITE .AND. MSITE <= HISTIE)
APPENDIX B: MAINTENANCE MANUAL

NEWDCVT.PRG Program Listing

Page 6

IF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE)
    SET COLOR TO W/B, W/B
    @ 24,0 SAY SPACE(80)
    SET COLOR TO W+/R, W+/R
    STORE 'Response must be between ' + LOSITE + ' and ' + HISITE + ' TO ERROR
    @ 24,21 SAY ERROR
    DO DELAY
    SET COLOR TO /W, /W
    @ 24,0 SAY MESSAGE
    SET COLOR TO /BR, /BR
    STORE '01' TO MSITE
    @ 13,54 GET MSITE PICT "99"
    READ
    ENDIF
ENDDO

ASK THE USER IF THE INPUTS ARE VALID OR NOT

SET COLOR TO W+/B, W+/B
@ 24,0 SAY SPACE(80)
@ 16,17 SAY CORRECT
SET COLOR TO R+/B, R+/B
@ 16,49 SAY "Y"
@ 16,56 SAY "N"
STORE "N" TO ACCEPT
@ 16,62 GET ACCEPT PICT "!"
READ

ENSURE THAT THE RESPONSE IS EITHER "Y" OR "N"

DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
    IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
        SET COLOR TO W+/R, W+/R
        @ 24,24 SAY "Response must be either N or Y"
        DO DELAY
        STORE "N" TO ACCEPT
    ENDIF
    SET COLOR TO R+/B, R+/B
    @ 16,62 GET ACCEPT PICT "!"
    READ
ENDDO

SET COLOR TO W+/B, W+/B
@ 16,15 SAY SPACE(55)

IF ACCEPT = "Y" THEN

ASK THE USER IF THE INPUT "PRN" FILE IS TO BE ERASED

SET COLOR TO W+/B, W+/B
APPENDIX B: MAINTENANCE MANUAL

Page 7

NEWDOCVT.PRG Program Listing

301 STORE "Do you want to erase the input file " + DBNAME + ;
302 @ 16,10 SAY MESSAGE
303 STORE "N" TO ERASIT
304 SET COLOR TO R+/B, R+/B
305 @ 16,46 SAY DBNAME
306 @ 16,45+LEN(DBNAME)+5 SAY "Y"
307 @ 16,45+LEN(DBNAME)+12 SAY "N"
308 @ 16,45+LEN(DBNAME)+17 GET ERASIT PICT "!"
309 READ
310 *
311 * ENSURE THAT THE RESPONSE IS EITHER "Y" OR "N"
312 *
313 DO WHILE .NOT. (ERASIT = "N" .OR. ERASIT = "Y")
314 IF .NOT. (ERASIT = "N" .OR. ERASIT = "Y") THEN
315 SET COLOR TO W+/R, W+/R
316 @ 24,24 SAY " Response must be either N or Y "
317 DO DELAY
318 STORE "N" TO ERASIT
319 ENDIF
320 ENDDO
321 SET COLOR TO R+/B, R+/B
322 @ 16,45+LEN(DBNAME)+17 GET ERASIT PICT "!"
323 READ
324 ENDDO
325 SET COLOR TO W+/B, W+/B
326 @ 16,10 SAY SPACE(65)
327 *
328 IF ERASIT = "Y" THEN
329 ERASE &DBNAME
330 ENDIF
331 EXIT
332 ELSE
333 SET COLOR TO /BR, /BR
334 @ 8,54 SAY " "
335 @ 11,54 SAY " "
336 @ 13,54 SAY " "
337 LOOP
338 ENDIF
339 ENDDO WHILE .T.
340 *
341 SET COLOR TO W+/B, W+/B
342 @ 16,10 SAY SPACE(65)
343 REPLACE ALL SLTEDO WITH "&MSITE"
344 USE EQUIP INDEX EQUIPSD
345 STORE MEFFDATE + &MSITE TO MKEY
346 GO TO P
347 IF EOF() = .T. THEN
348 "U" "NEWLYAD
349 ELSE
350 "FIND &MKEY

369
APPENDIX B: MAINTENANCE MANUAL

NEWDOCVT.PRG Program Listing

351 IF EOF() = .T. THEN
352 DO NEWDOADD
353 ELSE
354 SET COLOR TO R*+/ , R*+/;
355 @ 16,21 SAY " THIS IS A DUPLICATE DELIVERY ORDER! "
356 SET COLOR TO W+/B, W+/B
357 @ 17,17 SAY " Do you still desire to load it? (Yes or No): "
358 SET COLOR TO R+/B, R+/B
359 @ 17,51 SAY "Y"
360 @ 17,58 SAY "N"
361 STORE "N" TO ACCEPT
362 @ 17,63 GET ACCEPT PICT ")!
363 READ
364 DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
365 IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
366 SET COLOR TO W+/R, W+/R
367 @ 24,24 SAY " Response must be either N or Y "
368 DO DELAY
369 STORE "N" TO ACCEPT
370 ENDF
371 SET COLOR TO R+/B, R+/B
372 @ 17,63 GET ACCEPT PICT ")!
373 READ
374 ENDDO
375 SET COLOR TO W/B, W/B
376 @ 16,20 SAY SPACE(50)
377 @ 17,15 SAY SPACE(55)
378 IF ACCEPT = "Y" THEN
379 DO NEWDOADD
380 ENDF
381 ENDF
382 * CHECK TO SEE IF THERE ARE MORE DELIVERY ORDERS TO BE ADDED.
383 *
384 SET COLOR TO R+/B, R+/B
385 STORE "N" TO CHOICE
386 @ 21,68 GET CHOICE PICT ")!
387 READ
388 * ENSURE THAT THE RESPONSE IS EITHER "Y" OR "N"
389 *
390 DO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "Y")
391 IF .NOT. (CHOICE = "N" .OR. CHOICE = "Y") THEN
392 SET COLOR TO W+/R, W+/R
393 @ 24,24 SAY " Response must be either N or Y "
394 DO DELAY
395 STORE ")" TO CHOICE
396 ENDF
397 SET COLOR TO R+/B, R+/B
NEWDOCVT.PRG Program Listing

APPENDIX B: MAINTENANCE MANUAL

Page 9

1. NEWDOCVT.PRG Program Listing

401 | @ 21,68 GET CHOICE PICT "!"
402 | READ
403 | ENDDO
404 | IF CHOICE = "N" THEN
405 | EXIT
406 | ELSE
407 | SET COLOR TO W/B, W/B
408 | @ 19,10 SAY SPACE(65)
409 | @ 21,68 SAY " "
410 | SET COLOR TO /BR, /BR
411 | @ 8,54 SAY " "
412 | @ 11,54 SAY " "
413 | @ 13,54 SAY " "
414 | ENDIF
415 | ENDDO
416 | WHILE .T.
417 | ERASE ALL TEMPORARY DBF FILES CREATED DURING THE LOAD
418 | SET COLOR TO R+, R+
419 | @ 15,26 SAY " ERASING TEMPORARY DATABASES "
420 | CLOSE DATABASES
421 | SET CONSOLE OFF
422 | ERASE TEMP.DBF
423 | ERASE NEWDOMOD.DBF
424 | ERASE SERNYTIP.DBF
425 | SET CONSOLE ON
426 | RELEASE ALL LIKE M*, ACCEPT, CHOICE, CORRECT, DNUMNAME, ERASIT, ERROR;
427 | NOFILE, SYSDATE
428 | RETURN
PROCEDURE PRJRTS.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
* LCDR WINSTON H. BUCKLEY, SC, USN
* LCDR ROBERT F. BRADO, USN
* LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : PROVIDE THE USER A SELECTION OF PROJECT LEVEL REPORTS.

* INPUT FILES : NONE.

* OUTPUT FILES : NONE.

* CALLED BY : REPORTCMD.PRG

* MODULES CALLED : EQPPJRPT.PRG, SNOPJRPT.PRG

* LOCAL VARIABLES : PRJRTS

* DATE LAST TIME MODIFIED ============== 18 DECEMBER 1985 =============

* DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE SELECTION.

STORE "1" TO PRJRPTS
DO WHILE PRJRTS < "3"
   CLEAR
   ?? FLASH + "W.PROJRPTS/"
   SET CONSOLE OFF
   WAIT TO PRJRPTS
   SET CONSOLE ON
   DO CASE
     * PROCESS ROUTINE BASED ON THE USER"S SELECTION.
     * DO CASE
     * CALL THE EQUIPMENT PROJECT LEVEL REPORT.
       CASE PRJRTS = "1"
         DO EQPJRTP
       * CALL THE SERIAL NUMBER PROJECT LEVEL REPORT.
       CASE PRJRTS = "2"
         DO SNPJRTP
       * RETURN TO THE SPICE REPORTING LEVEL MENU.
       CASE PRJRTS = "3"
     END_CASE
     * END DO CASE
   END DO CASE
END DO WHILE PRJRTS < "3"
51 * RETURN TO THE CALLING PROGRAM
52 *
53 *
54 RETURN
55 ************************************************************************************
APPENDIX B: MAINTENANCE MANUAL

Page 1

REPORCMD.PRG Program Listing

1* PROCEDURE REPORCMD.PRG
2*
3* AUTHORS : LCDR EDWARD J. CASE, SC, USN
4* LCDR WINSTON H. BUCKLEY, SC, USN
5* LCDR ROBERT F. BRAVO, USN
6* LCDR ROBERT L. BEARD III, SC, USN
7*
8* PURPOSE : PROVIDE THE USER AN OPPORTUNITY TO SELECT A REPORT
9* LEVEL - PROJECT LEVEL, SITE LEVEL, OR DELIVERY ORDER
10* DATE LEVEL.
11*
12* INPUT FILES : NONE.
13*
14* OUTPUT FILES : NONE.
15*
16* MODULES CALLED : PROJRPTS.PRG, SITERPTS.PRG, LATZPTS.PRG
17*
18* LOCAL VARIABLES: SELEKT
19*
20* DATE LAST TIME MODIFIED = DECEMBER 1985 ==
21*
22* DISPLAY THE REPORT LEVEL MENU TO THE USER AND WAIT FOR THE SELECTION.
23*
24STORE "1" TO SELEKT
25DO WHILE SELEKT < "4"
26 SET COLOR TO W/B, W/B, B
27 CLEAR
28 ?? FLASH + "W.REPORCMD/"
29 SET CONSOLE OFF
30 WAIT TO SELECT
31 SET CONSOLE ON
32
33* PROCESS PICTURE BASED ON THE USER'S SELECTION.
34*
35DO CASE
36
37CALL THE PROJECT LEVEL REPORTS PROGRAM.
38CASE SELECT = "1"
39DO PROJRPTS
40
41CALL THE SITE LEVEL REPORTS PROGRAM.
42CASE SELECT = "2"
43DO SITERPTS
44
45CALL THE EFFECTIVE DELIVERY ORDER DATE LEVEL REPORTS PROGRAM.
46CASE SELECT = "3"
47DO DATERPTS
48
49RETURN TO THE MAIN MENU PROGRAM.
50CASE SELECT = "4"
 Page 2

REPORCMD.PRG Program Listing

51 | *
52 | ENDCASE
53 | *
54 | ENDDO (WHILE SELEKT < "4")
55 | *
56 | RETURN TO THE CALLING PROGRAM
57 | *
58 | RETURN
59 | **********************************************
PROCEDURE NAME : SELEC'OR. PRG

AUTHORS : LCDR EDWARD J. CASE, SC, USN
Lcdr WINSTON H. BUCKLEY, SC, USN
Lcdr ROBERT F. BRADO, USN
Lcdr ROBERT L. BEARD III, SC, USN

PURPOSE : TO PERMIT THE USER TO SELECT THE DESIRED PROCESSING ACTION. CHOICES INCLUDE: THE SPLICE CONFIGURER, LOTUS 1-2-3 FOR "WHAT-IF" ANALYSIS, AND THE dBASE III CONFIGURATION MANAGEMENT SYSTEM. CHANGES TO ACTIVE DIRECTORIES AND CALLS TO dBASE EXTERNAL PROGRAMS ARE EFFECTED WITH THE dBASE "RUN" COMMAND.

INPUT FILES : NONE.

OUTPUT FILES : NONE.

MODULES CALLED : SPLICE.COM; 123.EXE, MAINMENU.PRG, DELAY.PRG, WS.COM

GLOBAL VARIABLE: FLASH

LOCAL VARIABLES: ANS

DATE LAST TIME MODIFIED = 18 DECEMBER 1985

DBASE PROGRAM CONFIGURATION VARIABLES:

DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE USER'S CHOICE.

PUBLIC FLASH

STORE "1" TO ANS

DO WHILE .T.
SET COLOR TO W+/B, W+/B, B
CLEAR
FLASH = CH(145)
?? FLASH + "S.SELEC'OR.SCR/"
@ 24,0 SAY SPACE (80)
SET COLOR TO R+/B,R+/B
@ 21,53 GET ANS PICT "9"
READ
DO WHILE (ANS < "1" .OR. ANS > "6")
IF (ANS < "1" .OR. ANS > "6") THEN
APPENDIX B: MAINTENANCE MANUAL

Page 2

SELECIOR.PRG Program Listing

51 SET COLOR TO W+/R,W+/R
52 @ 24,23 SAY "Response must be between 1 and 6"
53 DO DELAY
54 STORE "1" TO ANS
55 ENDIF
56 SET COLOR TO R+/B,R+/B
57 @ 21,53 GET ANS PICT "9"
58 READ
59 ENDDO
60 * PERFORM APPROPRIATE TASK BASED ON THE USER'S CHOICE.
61 * DO CASE
62 *
63 * CHANGE THE ACTIVE DIRECTORY TO TURBO AND CALL SPLICE.COM.
64 * COPY THE OUTPUT .PRN FILE TO THE dBASE III SUBDIRECTORY.
65 *
66 CASE ANS = "1"
67 RUN CD\TURBO
68 RUN SPLICE.COM
69 RUN COPY *.PRN \DBASEIII\*.PRN
70 RUN CD\DBASEIII
71 STORE "1" TO ANS
72 *
73 * CHANGE THE ACTIVE SUBDIRECTORY TO LOTUS AND CALL 123.EXE. THE USER
74 * SUBDIRECTORY WHILE IN LOTUS MUST BE dBASE III.
75 *
76 CASE ANS = "2"
77 RUN CLS
78 RUN ECHO WHEN IN 123, CHANGE THE DEFAULT DIRECTORY TO DBASEIII
79 RUN PAUSE
80 RUN CD\LOTUS
81 RUN 123
82 RUN CD\DBASEIII
83 STORE "2" TO ANS
84 *
85 * CALL THE CONFIGURATION MANAGEMENT SYSTEM dBASE III PROGRAM
86 *
87 CASE ANS = "3"
88 DO MAINMENU
89 STORE "3" TO ANS
90 *
91 * CHANGE THE ACTIVE DIRECTORY TO WORDSTAR AND EDIT THE USER'S MANUAL.
92 *
93 CASE ANS = "4"
94 RUN CLS
95 RUN CD\WORDSTAR
96 RUN COPY USERS.MAN SPLICE.MAN
97 RUN WS.COM SPLICE.MAN
98 RUN DEL SPLICE.MAN
99 100
APPENDIX B: MAINTENANCE MANUAL

Page 3

SELECTOR.PRG Program Listing

101 | RUN CD\DBASEIII
102 | STORE "4" TO ANS
103 | *
104 | *
105 | * RETURN THE USER TO dBASE SYSTEM CONTROL.
106 | *
107 | CASE ANS = "5"
108 | CLEAR
109 | CLEAR ALL
110 | EXIT
111 | *
112 | * RETURN THE USER TO OPERATING SYSTEM CONTROL.
113 | *
114 | CASE ANS = "6"
115 | CLEAR
116 | CLEAR ALL
117 | STORE 0 TO CONTINUE
118 | QUIT
119 | *
120 | ENDCASE
121 | *
122 | * CONTINUE PROCESSING LOOP CONTROL CHECK.
123 | *
124 | ENDDO WHILE .T.
125 | ******************************************
PROCEDURE SERNOBLD.PRG

AUTHORS: LCDR EDWARD J. CASE, SC, USN
          LCDR WINSTON H. BUCKLEY, SC, USN
          LCDR ROBERT F. BRADO, USN
          LCDR ROBERT L. BEARD III, SC, USN

PURPOSE: TO BUILD BLANK SERIAL NUMBER RECORDS.

INPUT FILES: SERNOTMP.DBF

OUTPUT FILES: SERNOTMP.DBF

MODULES CALLED: NONE

CALLED BY: NEWDOADD.PRG

LOCAL VARIABLES: CIXJI7I'Y, INITIAL, MEFFDATE, MFEATURE, M/V/Y,
                  MSERIALN, MSITE, REC_COUNT

DATE LAST TIME MODIFIED: 23 DECEMBER 1985

IF NOT MEF, DETERMINE THE COMPONENT QUANTITY. WHILE THE QUANTITY IS GREATER THAN 1, BUILD AND EXPAND A BLANK SERIAL NUMBER RECORD.

STORE 1 TO INITIAL
USE SERNOTMP
GO TOP
DO WHILE .T.
  IF EOF() = .T. THEN
    EXIT
  ELSE
    IF INITIAL = 1 THEN
      SET COLOR TO GR+/B, GR+/B
      \ 17,21 SAY "FEATURE:"
      \ 17,40 SAY "RECORD NUMBER:"
      \ 19,18 SAY "Building and expanding sub-record "
      \ 19,56 SAY "of"
      STORE 0 TO INITIAL
    ENDIF
    SET COLOR TO /BR, /BR
    \ 17,31 SAY FEATURENO PICT "999999"
    SET COLOR TO R+/B, R+/B
    \ 17,55 SAY RECNO() PICT "9999"
    SET COLOR TO W+/BG, W+/BG
    STORE 1 TO REC_COUNT
    \ 19,52 SAY REC_COUNT PICT "999"
SERNOBLD.PRG Program Listing

51  @ 19,59 SAY QTY PICT "999"
52  STORE EFFDATE TO MEFFDATE
53  STORE SITENO TO MSITE
54  STORE FEATURENO TO MFEATURE
55  STORE QTY TO MQTY, CTOTQTY
56  REPLACE TOTQTY WITH MQTY
57  STORE ' ' TO MSERIALN
58  DO WHILE MQTY > 1
59  REC_COUNT = REC_COUNT + 1
60  @ 19,52 SAY REC_COUNT PICT "999"
61  INSERT BLANK
62  REPLACE TOTQTY WITH CTOTQTY
63  REPLACE QTY WITH MQTY - 1
64  REPLACE EFFDATE WITH MEFFDATE
65  REPLACE SITENO WITH MSITE
66  REPLACE FEATURENO WITH MFEATURE
67  REPLACE SERIALNO WITH MSERIALN
68  MQTY = MQTY - 1
69  ENDDO WHILE MQTY > 1
70  SKIP
71  ENDF(D/) = .T.
72  ENDDO WHILE .T.
73  CLEAR OUT THE STATUS FIELD LINES
74  SET COLOR TO W+/B, W+/B
75  @ 15,10 SAY SPACE(60)
76  @ 17,10 SAY SPACE(60)
78  RETURN ALL LIKE M*, CTOTQTY, INITIAL, REC_COUNT
79  CLOSE DATABASES
80  RETURN
81 ****************************

380
**APPENDIX B: MAINTENANCE MANUAL**

SERNOCMD.PRG Program Listing

1 | * PROCEDURE SERNOCMD.PRG
2 | *
3 | * AUTHORS: LCDR EDWARD J. CASE, SC, USN
4 | * LCDR WINSTON H. BUCKLEY, SC, USN
5 | * LCDR ROBERT F. BRADO, USN
6 | * LCDR ROBERT L. BEARD III, SC, USN
7 | *
8 | * PURPOSE: PROVIDE THE USER THE OPPORTUNITY TO INPUT
9 | * THE COMPONENT SERIAL NUMBER OR REVIEW THE
10 | * SERIAL NUMBER DATABASE RECORDS.
11 | *
12 | * INPUT FILES: NONE
13 | *
14 | * OUTPUT FILES: NONE
15 | *
16 | * CALLED BY: MAINMENU.PRG
17 | *
18 | * MODULES CALLED: SERNOUPD.PRG, SERNOREV.PRG
19 | *
20 | * LOCAL VARIABLES: SELECT
21 | *
22 | * DATE LAST TIME MODIFIED: 26 DECEMBER 1985
23 | *
24 | * DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE SELECTION.
25 | *
26 | STORE "1" TO SELECT
27 | DO WHILE SELECT < "3"
28 | SET COLOR TO W/B, W/B, B
29 | CLEAR
30 | ?? FLASH + "W.SERNOCMD/
31 | SET CONSOLE OFF
32 | WAIT TO SELECT
33 | SET CONSOLE ON
34 | *
35 | * PROCESS ROUTINE BASED ON THE USER'S SELECTION.
36 | *
37 | DO CASE
38 | *
39 | * CALL THE SERIAL NUMBER UPDATE PROGRAM.
40 | CASE SELECT = "1"
41 | DO SERNOUPD
42 | *
43 | * CALL SERIAL NUMBER REVIEW PROGRAM.
44 | CASE SELECT = "2"
45 | DO SERNOREV
46 | *
47 | * RETURN TO THE MAIN MENU PROGRAM.
48 | CASE SELECT = "3"
49 | *
50 | END CASE
APPENDIX B: MAINTENANCE MANUAL

Page 2

SERNOCMD.PRG Program Listing

51 *
52 ENDDO (WHILE SELEKT < "3")
53 *
54 * RETURN TO THE CALLING PROGRAM
55 *
56 RETURN
57 ************************************************************
* PROCEDURE SERNOREV.PRG

* AUTHORS : LCDR EDWARD J. CASE, SC, USN
            LCDR WINSTON H. BUCKLEY, SC, USN
            LCDR ROBERT F. BRADO, USN
            LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE : TO ENABLE THE USER TO REVIEW ALL RECORDS IN THE SERIAL NUMBER DATABASE.

* INPUT FILES : SERIALNO.DBF, SERNOSIT.NDX

* OUTPUT FILES : SERIALNO.DBF, SERNOSIT.NDX

* CALLED BY : SERNOCMD.PRG

* MODULES CALLED : DELAY.PRG

* GLOBAL VARIABLE: HIFNUM, HISITE, LOFNUM, LOSITE

* LOCAL VARIABLES: ACCEPT, CHOICE, CURRENTNO, EOF, FIRST_REC, LAST_REC,
                  MFILE, MFILEATOR, MSITE, TOF

* DATE LAST TIME MODIFIED = 26 DECEMBER 1985

* CASE SELECTION = 2    REVIEW SERIAL NUMBER FILE RECORDS

* USE SERIAL NUMBER DATABASE INDEXED ON SITE NUMBER AND WAIT FOR THE
* USER TO INPUT THE SITE NUMBER, THEN START REVIEWING FROM THAT POINT.

SET ESCAPE OFF
SET TALK OFF
SELECT 1
USE SERIALNO
GO TOP
SET COLOR TO W+/B, W+/B, B
CLEAR
IF EOF() = .T. THEN
    SET COLOR TO W+/R, W+/R
    # 13,22 SAY " The SERIAL NUMBER Database is EMPTY! "
    DO DELAY
    RETURN
END IF
FLASH + "S_SERIALNO.SCR"
SET COLOR TO W+/B, W+/B
# 24,0 SAY SPACE(80)
SET COLOR TO R+/R, R+/R
# 3,26 SAY " SERIAL NUMBER REVIEW PENDING "
? " Enter 00 to start at EOF, or to start at EOF or a site number ";
       " between " + MSITE + " and " + HISITE + " " TO SERIALNO

383
APPENDIX B: MAINTENANCE MANUAL

SERNOREV.PRG Program Listing

51 * 52 DO WHILE .T. 53 SET COLOR TO /W, /W 54 @ 24,0 SAY MESSAGE 55 SET COLOR TO /BR, /BR 56 STORE '00' TO MSITE 57 @ 09,20 GET MSITE PICT '99' 58 READ 59 IF .NOT. ((MSITE = '00' .AND. MSITE <= HISITE) .OR.; MSITE = '99') THEN 60 SET COLOR TO W/B, W/B 61 @ 24,0 SAY SPACE(80) 62 SET COLOR TO W+/R,W+/R 63 STORE 'Response must be between ' + LOSITE + ' and ' +; HISITE + ', Zero (00) or 99 ' TO ERROR 64 @ 24,13 SAY ERROR 65 DO DELAY 66 LOOP 67 ELSE 68 IF (MSITE = '00' .OR. MSITE = '99') THEN 69 IF MSITE = '00' THEN 70 GO BOTTOM 71 STORE RECN0() TO LAST_REC 72 GO TOP 73 STORE RECN0() TO FIRST_REC 74 ELSE 75 GO TOP 76 STORE RECN0() TO FIRST_REC 77 GO BOTTOM 78 STORE RECN0() TO LAST_REC 79 ENDIF MSITE = '00' 80 EXIT 81 ELSE 82 USE SERIALNO INDEX SERNO 83 GO TOP 84 FIND MSITE 85 IF EOF() = .T. THEN 86 IF EOF() = .T. THEN 87 SET COLOR TO W/B, W/B 88 @ 24,0 SAY SPACE(80) 89 SET COLOR TO W+/R, W+/R 90 STORE "No records exist for site number " + MSITE + "", try again " TO ERROR 91 @ 24,16 SAY ERROR 92 DO DELAY 93 LOOP 94 ELSE 95 EXIT 96 ELSE 97 ENDIF EOF() = .T. 98 ENDIF (MSITE = '00' .OR. MSITE = '99') 99 ENDIF .NOT. ((MSITE = '00' .AND. MSITE <= HISITE) .OR. MSITE = '99')
ENDDO WHILE .T.
* 
STORE SPACE(10) + 'Enter "00 " to start at TOF or a six digit ' +
'feature number' + SPACE(10) TO MESSAGE
IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
DO WHILE .T.
  SET COLOR TO /W, /W
  $ 24,0 SAY MESSAGE
  SET COLOR TO /BR, /BR
  STORE '00 ' TO MFEAT
  $ 13,45 GET MFEAT PIC '999999'
  READ 
  IF .NOT. ((MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM) .OR.;
    MFEAT = '00 ') THEN
    SET COLOR TO W/B, W/B
    $ 24,0 SAY SPACE(80)
    SET COLOR TO W+/R, W+/R
    STORE ' Response must be between ' + LOFNUM + ' + and ' +;
    HIFNUM + ' or Zero (00) ' TO ERROR
    $ 24,9 SAY ERROR
    DO DELAY
    LOOP
  ELSE
    IF (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
      IF MFEAT = '99 ' THEN
        SET COLOR TO W/B, W/B
        $ 24,0 SAY SPACE(80)
        SET COLOR TO W+/R, W+/R
        STORE ' Response must be between ' + LOFNUM +;
        ' + HIFNUM + ' or Zero (00) ' TO ERROR
        $ 24,9 SAY ERROR
        DO DELAY
        LOOP
      ENDIF MFEAT = '99 '
      STORE MSITE + MFEAT TO MKEY
      USE SERIALNO INDEX SERNOREV
      GO TOP
    ELSE
      FIND &MKEY
      IF EOF() = .T. THEN
        SET COLOR TO W/B, W/B
        $ 24,0 SAY SPACE(80)
        SET COLOR TO W+/R, W+/R
        STORE " No record with feature number " + MFEAT +;
        " exists, try again " TO ERROR
        $ 24,12 SAY ERROR
        DO DELAY
        LOOP
      ELSE
        EXIT
      ENDIF EOF() = .T.
151    ELSE
152       EXIT
153    ENDF (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
154    ENDIF
155 ENDDO WHILE .T.
156 ENDF .NOT. (MSITE = LOSITE .OR. MSITE <= HISITE)
157 *
158 STORE " At beginning of records for site number " +;
159 MSITE + " " TO TOF
160 STORE " At end of records for site number " + MSITE + " " TO EOF
161 SET COLOR TO W/B, W/B
162 @ 24, 0 SAY SPACE(80)
163 *
164 DO WHILE .T.
165 *
166 * USING THE SERIAL NUMBER REVIEW FORMAT FILE TO PRODUCE THE SCREEN
167 * DISPLAY, IF NOT AT THE END OF FILE.
168 *
169 STORE FEATURENO TO MFEATURE
170 SELECT 2
171 USE DESCRIP INDEX DESCRIP
172 FIND &MFEATURE
173 STORE CLIN TO MCLIN
174 STORE DESCRIPT TO MDESCRIPT
175 SELECT 1
176 SET COLOR TO R+/B, R+/B
177 @ 6, 45 SAY RECNO( ) PICT "9999"
178 SET COLOR TO /BR, /BR
179 @ 9, 20 SAY SITENO PICT "99"
180 @ 9, 68 SAY EFFDATE PICT "999999"
181 @ 12, 45 SAY MCLIN PICT "9999"
182 @ 13, 45 SAY FEATURENO PICT "999999"
183 @ 14, 45 SAY MDESCRIPT PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!"
184 @ 15, 45 SAY TOTXY PICT "999"
185 SET COLOR TO W+/BG, W+/BG
186 @ 17, 45 SAY OTY PICT "999"
187 @ 17, 52 SAY TOTXY PICT "999"
188 SET COLOR TO /BR, /BR
189 @ 19, 45 SAY SERIALNO PICT "!!!!!!!"
190 ENDF
191 *
192 SET COLOR TO R+/B, R+/B
193 STORE "N" TO CHOICE
194 @ 22, 68 GET CHOICE PICT "!
195 READ
196 *
197 * ENSURE THAT THE USER'S PROMPT IS EITHER "N", "P" OR "X"
198 *
199 DO WHILE .NOT. (CHOICE = " " .OR. CHOICE = "P" .OR. CHOICE = "N")
200 IF .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X") THEN
201 SET COLOR TO W+/R, W+/R
202 @ 24,23 SAY "Response must be either N, P or X"
203 DO DELAY
204 STORE "N" TO CHOICE
205 ENDIF
206 SET COLOR TO R+/B, R+/B
207 @ 22,68 GET CHOICE PICT ":"
208 READ
209 ENDDO
210 *
211 SET COLOR TO W+/R, W+/R
212 *
213 * SKIP TO THE NEXT RECORD TO BE REVIEWED
214 *
215 IF CHOICE = "N" THEN
216 IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
217 SKIP
218 IF EOF() = .T. THEN
219 SKIP - 1
220 @ 24,21 SAY EOF
221 DO DELAY
222 LOOP
223 ELSE
224 IF NOT. (SITENO = MSITE) THEN
225 SKIP - 1
226 @ 24,21 SAY EOF
227 DO DELAY
228 LOOP
229 ENDIF
230 ENDIF EOF() = .T.
231 ELSE
232 IF RECNO() = LAST_REC THEN
233 GO TOP
234 ELSE
235 SKIP
236 ENDIF
237 ENDIF MSITE >= LOSITE .AND. MSITE <= HISITE
238 ENDIF CHOICE = "N"
239 *
240 * SKIP TO THE PREVIOUS RECORD
241 *
242 IF CHOICE = "P" THEN
243 STORE RECNO() TO CURRENTN
244 IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
245 SKIP - 1
246 IF EOF() = .T. THEN
247 GO TO (CURRENTN)
248 @ 24,16 SAY EOF
249 DO DELAY
250 LOOP
APPENDIX B: MAINTENANCE MANUAL

SERNOREV.PRG Program Listing

ELSE
   IF .NOT. (SITENO = MSITE) THEN
      SKIP
      @ 24,16 SAY TOF
      DO DELAY
      LOOP
      ENDIF
      ENDIF BOF() = .T.
   ELSE
      IF RECNO() = FIRST_REC THEN
         GO BOTTOM
      ELSE
         SKIP - 1
      ENDIF
   ENDIF
   ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE')
   ENDIF CHOICE = "P"
   *
   * USER HAS DECIDED TO EXIT THE REVIEW
   *
   IF CHOICE = "x"
   EXIT
   ENDIF
   ENDDO WHILE .T.
   *
   * RETURN TO CALLING PROGRAM.
   *
   RELEASE ALL LIKE M*, ACCEPT, CHOICE, CURRENTNO, BOF, FIRST_REC,;
   LAST_REC, TOF
   CLOSE DATABASES
   RETURN

******************************************************************************
APPENDIX B: MAINTENANCE MANUAL  Page 252

SERNOUPD.PRG Program Listing

* PROCEDURE SERNOUPD.PRG

* AUTHORS
: LCDR EDWARD J. CASE, SC, USN
: LCDR WINSTON H. BUCKLEY, SC, USN
: LCDR ROBERT F. BRADO, USN
: LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE
: TO ENABLE THE USER TO INPUT THE SERIAL NUMBERS FOR
: THE SERIAL NUMBER DATABASE.

* INPUT FILES
: SERIALNO.DBF, SERNOSIT.NDX

* OUTPUT FILES
: SERIALNO.DBF, SERNOSIT.NDX

* CALLED BY
: SERNOCMD.PRG

* MODULES CALLED : DELAY.PRG

* GLOBAL VARIABLE: IIIDATE, HIFNUM, HISITE, LIDATE, IJFNUM, LJSITE

* LOCAL VARIABLES: ACCEPT, ANS, CHOICE, CURRENTNO, EOF, INTRO, MDAY,;
: MESSAGE, MMONTH, MOLDATE, MYEAR, NOFIND, SYSDATE, TOF

* DATE LAST TIME MODIFIED ===========> 26 DECEMBER 1985 <==========

* CASE SELECTION = 2           REVIEW SERIAL NUMBER FILE RECORDS

* USE SERIAL NUMBER DATABASE INDEXED ON SITE NUMBER AND WAIT FOR THE
* USER TO INPUT THE SITE NUMBER, THEN START REVIEWING FROM THAT POINT.

SET ESCAPE OFF
SET TALK OFF
SELECT 1
USE SERIALNO
GO TOP
SET COLOR TO W+/B, W+/B, B
CLEAR

IF EOF() = .T. THEN
  SET COLOR TO W+/R, W+/R
  "13,22 SAY "THE SERIAL NUMBER DATABASE IS EMPTY!"
  "0) DELAY
  RETURN
ENDIF

?? FLASH + "S.SERIALNO.SCR/"
SET COLOR TO W+/B, W+/B
"24,0 SAY SPACE(80)
SET COLOR TO R+/R, R+/R
"3,26 SAY "SERIAL NUMBER UPDATE FORMAT"
STORE SPACE(22) + "Enter a Site Number between " + LJSITE + ;
" and " * HISITE + SPACE(21) TO MESSAGE
52 USE SERIALNO INDEX SERNUMIT
53 *
54 DO WHILE .T.
55   SET COLOR TO /W, /W
56     \$24,0 SAY MESSAGE
57   SET COLOR TO /BR, /BR
58   STORE HISITE TO MSITE
59     \$ (0,2) SET WHITE PICT '99'
60   READ
61   IF NOT. (MSITE > LOSITE .AND. MSITE <= HISITE)
62     SET COLOR TO W/B, W/B
63       \$ 24,0 SAY SPACE(80)
64     SET COLOR TO W+R, W/R
65     STORE "Response must be between ' + LOSITE + ;
66       ' and ' + HISITE + ' TO ERROR
67     \$ 24,16 SAY ERROR
68     DO DELAY
69     LOOP
70 ELSE
71   GO TO IF
72   FIND MSITE
73   IF EOF() = .T. THEN
74     SET COLOR TO W/B, W/B
75       \$ 24,0 SAY SPACE(80)
76     SET COLOR TO W+R, W/R
77     STORE "No records exist for site number " + MSITE + ;
78       " , try again " TO ERROR
79     \$ 24,16 SAY ERROR
80   DO DELAY
81   LOOP
82 ELSE
83   EXIT
84   ENDIF EOF() = .T.
85 ENDIF NOT. (MSITE > LOSITE .AND. MSITE <= HISITE)
86 ENDDO WHILE .NOT. (MSITE > LOSITE .AND. MSITE <= HISITE)
87 *
88 STORE SPACE(17) + 'Input Effective Date (Range ' + LIDATE + ;
89       ' to ' + HIDATE + ')' + SPACE(17) TO MESSAGE
90 STORE DTOC(LIDATE()) TO SYSDATE
91 STORE SUBSTR(SYSDATE,1,2) + SUBSTR(SYSDATE,1,2) +
92       SUBSTR(SYSDATE,4,2) TO MDATE
93 STORE 0 TO NOFIND
94 STORE "000000" TO MDATE
95 USE SERIALNO INDEX SERNUMIT
96 *
97 DO WHILE .NOT. (MDATE > LIDATE .AND. MDATE <= HIDATE)
98   SET COLOR TO /W, /W
99     \$ 24,0 SAY MESSAGE
100 STORE MDATE TO MDATE
SERNOUPD.PRG Program Listing

101 SET COLOR TO /BR, /BR
102 @ 9,68 GET MOLDATE PICT "999999"
103 READ
104 DO WHILE .T.
105 IF .NOT. (SUBSTR(MOLDATE,1,2) > "83" .AND.)
106 SUBSTR(MOLDATE,1,2) <= "99") THEN
107 SET COLOR TO W/B, W/B
108 @ 24,0 SAY SPACE(80)
109 SET COLOR TO W+/R, W+/R
110 STORE " Year portion of date must be between 84 and 99 ";
111 TO ERROR
112 @ 24,16 SAY ERROR
113 DO DELAY
114 SET COLOR TO /W, /W
115 @ 24,0 SAY MESSAGE
116 STORE SUBSTR(MDATE,1,2) TO MYEAR
117 SET COLOR TO /BR, /BR
118 @ 9,68 GET MYEAR PICT "99"
119 READ
120 STORE MYEAR + SUBSTR(MOLDATE,3,4) TO MOLDATE
121 LOOP
122 ELSE
123 EXIT
124 ENDDO WHILE .T.
125 ENDIF
126 DO WHILE .T.
127 IF .NOT. (SUBSTR(MOLDATE,1,2) > "01" .AND.)
128 SUBSTR(MOLDATE,1,2) <= "12") THEN
129 SET COLOR TO W/B, W/B
130 @ 24,0 SAY SPACE(80)
131 SET COLOR TO W+/R, W+/R
132 @ 24,16 SAY " Month portion of date must be between 01 and 12 ";
133 DO DELAY
134 SET COLOR TO /W, /W
135 @ 24,0 SAY MESSAGE
136 STORE SUBSTR(MDATE,3,2) TO MMNTH
137 SET COLOR TO /BR, /BR
138 @ 9,70 GET MMNTH PICT ":00"
139 READ
140 STORE SUBSTR(MOLDATE,1,2) + MMNTH ;
141 SUBSTR(MOLDATE,5,2) TO MOLDATE
142 LOOP
143 ELSE
144 EXIT
145 ENDIF
146 ENDDO WHILE .T.
147 ENDIF
148 DO WHILE .T.
149 IF (SUBSTR(MOLDATE,1,2) = "04" .OR. SUBSTR(MOLDATE,3,2) = "04") THEN

391
APPENDIX B: MAINTENANCE MANUAL

SERNoupd.prg Program Listing

Page 4

151 | SUBSTR(MOLDATE,3,2)="09" .OR. SUBSTR(MOLDATE,3,2)="11") .AND.;
152 | .NOT. (SUBSTR(MOLDATE,5,2)="01") .AND.;
153 | SUBSTR(MOLDATE,5,2)="30") THEN
154 | SET COLOR TO W/B, W/B
155 | @ 24,0 SAY SPACE(80)
156 | SET COLOR TO W+/R, W+/R
157 | @ 24,16 SAY "Day portion of date must be between 01 and 30"
158 | DO DELAY
159 | SET COLOR TO /W, /W
160 | @ 24,0 SAY MESSAGE
161 | STORE SUBSTR(MDATE,5,2) TO MDAY
162 | SET COLOR TO /BR, /BR
163 | @ 9,72 GET MDAY PICT "99"
164 | READ
165 | STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
166 | LOOP
167 | ELSE
168 | IF (SUBSTR(MOLDATE,3,2) = "02") .AND. ".NOT.
169 | (SUBSTR(MOLDATE,5,2) = "01") .AND.
170 | SUBSTR(MOLDATE,5,2) = "28") THEN
171 | SET COLOR TO W/B, W/B
172 | @ 24,0 SAY SPACE(80)
173 | SET COLOR TO W+/R, W+/R
174 | @ 24,16 SAY "Day portion of date must be between 01 and 28"
175 | DO DELAY
176 | SET COLOR TO /W, /W
177 | @ 24,0 SAY MESSAGE
178 | STORE SUBSTR(MDATE,5,2) TO MDAY
179 | SET COLOR TO /BR, /BR
180 | @ 9,72 GET MDAY PICT "99"
181 | READ
182 | STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
183 | LOOP
184 | ELSE
185 | IF .NOT. (SUBSTR(MOLDATE,5,2) = "01") .AND.
186 | SUBSTR(MOLDATE,5,2) = "31") THEN
187 | SET COLOR TO W/B, W/B
188 | @ 24,0 SAY SPACE(80)
189 | SET COLOR TO W+/R, W+/R
190 | @ 24,16 SAY "Day portion of date must be between 01 and 31"
191 | DO DELAY
192 | SET COLOR TO /W, /W
193 | @ 24,0 SAY MESSAGE
194 | STORE SUBSTR(MDATE,5,2) TO MDAY
195 | SET COLOR TO /BR, /BR
196 | @ 9,72 GET MDAY PICT "99"
197 | READ
198 | STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE

392
APPENDIX B: MAINTENANCE MANUAL

SERNOUPD.PRG Program Listing

```
201     LOOP
202     ELSE
203         EXIT
204     ENDIF
205     ENDIF
206     ENDIF
207     ENDDO WHILE .T.
208
209     * SEE IF THE USER'S DATE IS A VALID DATE FOR THE SITE SELECTED
210
211     STORE MSITE + YOLDATE TO &KEY
212     GO TOP
213     FIND &KEY
214     IF EOF() = .T. THEN
215         NOFIND = NOFIND + 1
216     IF NOFIND = 3 THEN
217         SET COLOR TO W+/B, W+/B
218         @ 24,0 SAY SPACE(80)
219         ? FLASH + "W.SERNOFND/"
220         SET CONSOLE OFF
221         WAIT TO ANS
222         SET CONSOLE ON
223         IF ANS = "2" THEN
224             RELEASE ALL LIKE M*, ACCEPT, ANS, CHOICE,
225             CURRENTNO, EOF, INTRO, NODEDATE,
226             NOFIND, SYSDATE, TOF
227             CLOSE DATABASES
228             RETURN
229     ELSE
230         SET COLOR TO /W, /W
231         @ 24,0 SAY MESSAGE
232         STORE 0 TO NOFIND
233         STORE '000000' TO YOLDATE
234         LOOP
235         ENDIF ANS = "2"
236     ELSE
237         SET COLOR TO W/B, W/B
238         @ 24,0 SAY MESSAGE
239         STORE " EFFECTIVE DATE " + YOLDATE + " does not exist for site " + MSITE + ", try another " TO NODEDATE
240         SET COLOR TO W+/R, W+/R
241         @ 24,10 SAY NODEDATE
242         DO DELAY
243         SET COLOR TO /W, /W
244         @ 24,0 SAY MESSAGE
245         STORE "000000" TO NODEDATE
246         LOOP
247         ENDIF NOFIND = 3
248         ENDIF EOF() = .T.
249     ENDDO WHILE .NOT. (YOLDATE > DDATE .AND. YOLDATE <= HILDATE)
```

393
APPENDIX B: MAINTENANCE MANUAL  Page 257

Page 6  SERNUPDATE.PRG Program Listing

251 *
252 STORE SPACE(10) + 'Enter a six digit feature number between ' + LOFNUM + ' and ' + HIFNUM + SPACE(11) TO MESSAGE
253 @ 24,0 SAY MESSAGE
254 STORE '999999' TO MFEAT
255 STORE 0 TO NOFIND
256 *
257 DO WHILE .T.
258   DO WHILE .NOT. (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
259     SET COLOR TO /BR, /BR
260     STORE '010201' TO MFEAT
261     SET COLOR TO W/B, W/B
262     STORE 13,45 GET MFEAT PICT '999999'
263     READ
264     IF .NOT. (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM) THEN
265       SET COLOR TO W/B, W/B
266       @ 24,0 SAY SPACE(80)
267       SET COLOR TO W+/R, W+/R
268       STORE 'Response must be between ' + LOFNUM + ' and ' + HIFNUM + ' to ERROR'
269       @ 24,17 SAY ERROR
270       DO DELAY
271       SET COLOR TO /W, /W
272       @ 24,0 SAY MESSAGE
273     ELSE
274       IF NOFIND = NOFIND + 1 THEN
275         SET COLOR TO W+/B, W+/B
276         SAY SPACE(80)
277         FLASH 'SERNOUPD.PRG'
278         SET CONSOLE OFF
279         WAIT '1'
280         IF ANS = "2" THEN
281           RETURN
282         IF ANS = "2" THEN
283           STORE 0 TO NOFIND
284           STORE '999999' TO MFEAT
285           USE DESCRIPT INDEX DESCRIPT
286           GO TOP
287           FIND MFEAT
288           IF LOF() = .T. THEN
289             NOFIND = NOFIND + 1
290           IF NOFIND = 3 THEN
291             SET COLOR TO W+/R, W+/R
292             @ 24,0 SAY SPACE(80)
293             ?? FLASH 'SERNOUPD.PRG'
294             SET CONSOLE OFF
295             WAIT TO ANS
296             SET CONSOLE ON
297             IF ANS = "2" THEN
298               RELEASE ALL LIKE M*, ACCEPT, ANS, WORK, CURRENT, DESCRIPT INDEX, NOFIND, SYSDATE, TOP
299               CLOSE DATABASES
300               RETURN
301           ELSE
302             SET COLOR TO /W, /W
303             @ 24,0 SAY MESSAGE
304             STORE 0 TO NOFIND
305             STORE '999999' TO MFEAT
306           END IF
307         END IF
308       ELSE
309         USE DESCRIPT INDEX DESCRIPT
310         GO TOP
311         FIND MFEAT
312         IF NOFIND = NOFIND + 1 THEN
313           SET COLOR TO W+/B, W+/B
314           SAY SPACE(80)
315           FLASH 'SERNOUPD.PRG'
316           SET CONSOLE OFF
317           WAIT '1'
318           IF ANS = "2" THEN
319             RETURN
320           IF ANS = "2" THEN
321             STORE 0 TO NOFIND
322             STORE '999999' TO MFEAT
323           END IF
324           ELSE
325             SET COLOR TO /W, /W
326             @ 24,0 SAY MESSAGE
327             STORE 0 TO NOFIND
328             STORE '999999' TO MFEAT
329           END IF
330         END IF
331       END IF
332     END IF
333   END IF
334 END DO
APPENDIX B: MAINTENANCE MANUAL

SERNoupd.prg Program Listing

301 LOOP
302 ENDIF ANS = "2"
303 ELSE
304 SET COLOR TO W/B, W/B
305 $ 24,0 SAY SPACE(80)
306 SET COLOR TO W+/R, W+/R
307 STORE "No record exists for feature number ": MFEAT "+", try again " TO ERROR
308 $ 24,12 SAY ERROR
309 DO DELAY
310 SET COLOR TO /W, /W
311 $ 24,0 SAY MESSAGE
312 STORE '999999' TO MFEAT
313 LOOP
314 ENDIF NOFIND = 3
315 ENDIF EOF() = .T.
316 END (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
317 ENDIF .NOT. (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
318 ENDDD WHILE .NOT. (MFEAT >= LOFNUM .AND. MFEAT <= HIFNUM)
319 *
320 STORE MOLDATE + WSITE + MFEAT TO MKEY
321 USE SERIALNO INDEX SERNORID
322 GO TOP
323 FIND &MKEY
324 IF EOF() = .T. THEN
325 SET COLOR TO W/B, W/B
326 $ 24,0 SAY SPACE(80)
327 SET COLOR TO W+/R, W+/R
328 STORE " Feature number " + MFEAT + " for site " + WSITE + " on date " + MOLDATE + " does not exist, try again " TO ERROR
329 $ 24,0 SAY ERROR
330 DO DELAY
331 SET COLOR TO W+B, W+B
332 ?? FLASH + "W.SERIALNO/"
333 SET CONSOLE OFF
334 WAIT TO EOF
335 SET CONSOLE ON
336 IF EOF = "2" THEN
337 RELEASE ALL LIKE M*, ACCEPT, AS*, CHOSE, CURRENT, DEF,;
338 INPUT, INSERT, MODIF, SYST, TOP
339 CLOSE DATABASES;
340 RETURN
341 ELSE
342 SET COLOR TO W/B, W/B
343 $ 24,10 SAY SPACE(60)
344 SET COLOR TO /W, /W
345 $ 24,0 SAY MESSAGE
346 STORE '999999' TO MFEAT
347 LOOP
APPENDIX B: MAINTENANCE MANUAL

Page 259

SERNOUPD.PRG Program Listing

351 | ENDF ANS = "2"
352 | ELSE
353 |   EXIT
354 | ENDF EOF() = .T.
355 | ENDDO WHILE .T.
356 | *
357 | STORE " At beginning of records for site number " +
358 |   MSITE + " " TO TOF
359 | STORE " At end of records for site number " + MSITE + " " TO EOF
360 | SET COLOR TO W/B, W/B
361 | @ 24,0 SAY SPACE(80)
362 | *
363 | STORE SPACE(16) + 'Press "Page Down" key to terminate record update' +
364 |   SPACE(16) TO MESSAGE
365 | STORE 1 TO INTRO
366 | DO WHILE .T.
367 |   SET COLOR TO /W, /W
368 |   @ 24,0 SAY MESSAGE
369 | *
370 | * USING THE SERIAL NUMBER UPDATE FORMAT FILE TO PRODUCE THE SCREEN
371 | * DISPLAY, IF NOT AT THE END OF FILE.
372 | *
373 | STORE SERIALNO TO MSERIAL
374 | STORE FEATURENO TO MFEAT
375 | *
376 | * INFORM THE USER OF HOW TO TERMINATE THE UPDATE OF A RECORD
377 | *
378 | IF INTRO = 1 THEN
379 |   STORE 0 TO INTRO
380 |   ?? FLASH + "W.SERNOUPD/"
381 |   SET CONSOLE OFF
382 |   WAIT TO ANS
383 |   SET CONSOLE ON
384 | ENDIF
385 | *
386 | SELECT 2
387 | USE DESCRIP INDEX DESCRIP
388 | FIND MFEAT
389 | STORE CLIN TO MCLIN
390 | STORE DESCRIP TO MDESCIP
391 | SELECT 1
392 | SET COLOR TO R+/B, R+/B
393 | @ 6,45 SAY RECNO() PICT "9999"
394 | SET COLOR TO /BR, /BR
395 | @ 9,20 SAY SERIAL PICT "99"
396 | @ 9,68 SAY EFFDATE PICT "999999"
397 | @ 12,45 SAY MCLIN PICT "9999"
398 | @ 13,45 SAY MFEAT PICT "999999"
399 | @ 14,45 SAY MDESCIP PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!"
400 | @ 15,45 SAY TOTQTY PICT "999"
SERNOUPD.PRG Program Listing

* 401 SET COLOR TO W+/BG, W+/BG
* 402 @ 17,45 SAY QTY PICT "999"
* 403 @ 17,52 SAY TOTQTY PICT "999"
* 404 SET COLOR TO /BR, /BR
* 405 @ 19,45 GET MSERIAL PICT "!!!!!!!"
* 406 READ
* 407 SET COLOR TO W/B, W/B
* 408 @ 24,0 SAY SPACE(80)
* 409 * IF .NOT. (SERIALNO = MSERIAL) THEN
* 410 * ASK THE USER IF HE/SHE DESIRES TO ACCEPT THE CHANGES
* 411 * SET COLOR TO W/B, W/B
* 412 @ 21,12 SAY "Do you want to accept the change? (Yes or No):"
* 413 * SET COLOR TO R+/B, R+/B
* 414 @ 21,48 SAY "Y"
* 415 @ 21,55 SAY "N"
* 416 STORE "N" TO ACCEPT
* 417 @ 21,61 GET ACCEPT PICT ";"
* 418 READ
* 419 * ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
* 420 * DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
* 421 * IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
* 422 * SET COLOR TO W+/R, W+/R
* 423 @ 24,24 SAY "Response must be either N or Y"
* 424 * DO DELAY
* 425 * ENDIF ACCEPT
* 426 SET COLOR TO R+/B, R+/B
* 427 @ 21,61 GET ACCEPT PICT ";"
* 428 READ
* 429 * STORE THE CHANGED EDIT FIELD FROM THE WORK AREA INTO THE
* 430 * DATABASE VARIABLE
* 431 * IF ACCEPT = "Y" THEN
* 432 REPLACE SERIALNO WITH MSERIAL
* 433 ELSE
* 434 SET COLOR TO /BR, /BR
* 435 @ 19,45 SAY SERIALNO PICT "!!!!!!!"
* 436 ENDIF ACCEPT = "Y"
* 437 ENDIF .NOT. (SERIALNO = MSERIAL)
* 438 * SET COLOR TO R+/B, R+/B

397
APPENDIX B: MAINTENANCE MANUAL

Page 10

SERNOUPD.PRG Program Listing

STORE "N" TO CHOICE
@ 22,68 GET CHOICE PICT ".1"
READ
*
ENSURE THAT THE USER'S RESPONSE IS EITHER "N", "P" OR "X"
*
DO WHILE .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X")
  IF .NOT. (CHOICE = "N" .OR. CHOICE = "P" .OR. CHOICE = "X") THEN
    SET COLOR TO W+/R, W+/R
    @ 24,23 SAY " Response must be either N, P or X "
    DO DELAY
    STORE "N" TO CHOICE
  ENDIF
  SET COLOR TO R+/B, R+/B
  @ 22,68 GET CHOICE PICT ".!"
  READ
ENDDO
*
SKIP TO THE NEXT RECORD TO BE REVIEWED
*
IF CHOICE = "N" THEN
  IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
    SKIP
    IF EOF() = .T. THEN
      SKIP - 1
      SET COLOR TO W+/R, W+/R
      @ 24,21 SAY EOF
      DO DELAY
    ELSE
      IF .NOT. (SITENO = MSITE) THEN
        SKIP - 1
        SET COLOR TO W+/R, W+/R
        @ 24,21 SAY EOF
        DO DELAY
      ENDIF
    ENDIF EOF() = .T.
  ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE)
  ENDIF CHOICE = "N"
*
SKIP TO THE PREVIOUS RECORD
*
IF CHOICE = "P" THEN
  STORE RECNO() TO CURRENTNO
  IF (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
    SKIP - 1
    IF EOF() = .T. THEN
      G013 CURRENTNO
      SET COLOR TO W+/R, W+/R
      @ 24,16 SAY "W"
      DO DELAY
    ENDIF
  ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE)
  ENDIF CHOICE = "P"
*
398
ELSE
    IF .NOT. (SITENO = MSITE) THEN
        SKIP
        SET COLOR TO W+/R, W+/R
        @ 24,16 SAY TOF
        DO DELAY
    ENDIF
    ENDIF BOF() = .T.
ENDIF (MSITE >= LOSITE .AND. MSITE <= HISITE)
    ENDIF CHOICE = "P"
    * USER HAS DECIDED TO EXIT THE REVIEW
    * IF CHOICE = "X"
    EXIT
    ENDDO WHILE .T.
    *
    * RETURN TO CALLING PROGRAM.
    *
    * RELEASE ALL LIKE M*, ACCEPT, ANS, CHOICE, CURRENTNO, BOF, INTRO,;
    NODATE, NOFIND, SYSDATE, TOF
    CLOSE DATABASES
    RETURN

*******************************************************************************
/* PROCEDURE SITERPTS.PRG */

/* AUTHORS */
LCDR EDWARD J. CASE, SC, USN
LCDR WINSTON H. BUCKLEY, SC, USN
LCDR ROBERT F. BRADO, USN
LCDR ROBERT L. BEARD III, SC, USN

/* PURPOSE */
PROVIDE THE USER A SELECTION OF SITE LEVEL REPORTS.

/* INPUT FILES */
NONE.

/* OUTPUT FILES */
NONE.

/* CALLED BY */
REPORCMD.PRG

/* MODULES CALLED */
EQPSTRPT.PRG, MNLSTRPT.PRG, SNOSTRPT.PRG

/* LOCAL VARIABLES: */
SELEKT

/* DATE LAST TIME MODIFIED */
18 DECEMBER 1985

/* DISPLAY THE PROCESS MENU TO THE USER AND WAIT FOR THE SELECTION. */
STORE "$1" TO SITERPTS
DO WHILE SITERPTS < "$4"
SET COLOR TO W/B, W/B, B
CLEAR
?? FLASH + "$SITERPTS/"
SET CONSOLE OFF
WAIT TO SITERPTS
SET CONSOLE ON

/* PROCESS ROUTINE BASED ON THE USER'S SELECTION. */
DO CASE
CASE SITERPTS = "$1"

CALL THE EQUIPMENT SITE LEVEL REPORT.
CASE SITERPTS = "$2"

CALL THE MANUAL SITE LEVEL REPORT.
CASE SITERPTS = "$3"

CALL THE SERIAL NUMBER SITE LEVEL REPORT.
CASE SITERPTS = "$4"

RETURN TO THE SPICE REPORTING LEVEL MENU.

RETURN TO CASE.
SITERPTS.PRG Program Listing

51 *
52  ENDCASE
53 *
54 ENDDO (WHILE SITERPTS = "4")
55 *
56 * RETURN TO THE CALLING PROGRAM
57 *
58 RETURN
59 *********************************************************************
* PROCEDURE SNOIRPT.PRG

* AUTHORS
  : LCDR EDWARD J. CASE, SC, USN
  : LCDR WINSTON H. BUCKLEY, SC, USN
  : LCDR ROBERT F. BRADO, USN
  : LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE
  : PROVIDE THE USER A SPLICE SERIAL NUMBER EFFECTIVE DELIVERY ORDER LEVEL REPORT.

* INPUT FILES
  : SERIALNO.DBF, SERNODAT.NDX, DESCRIP.DBF,
  : DESCRIP.NDX, EQUIP.DBF, EQUIPSIT.NDX

* CALLED BY
  : DATERPTS.PRG

* MODULES CALLED : NONE.

* GLOBAL VARIABLE: HIDATE, HISITE, LDATED, LDISTE

* LOCAL VARIABLES:
  : ACCEPT, CHOICE, COLCNT, ERROR, LINEDID, MDAY, MDATE,
  : MDAY, MNDAT, MDATE, MDISTE, MDAY, PAGEO

* SYSDATE, 'IDATE, 'IDATE

* DATE LAST TIME MODIFIED ============= 27 DECEMBER 1985 ===========

* CASE SELECTION = 3 SERIAL NUMBER EFFECTIVE DELIVERY ORDER LEVEL REPORT

* CALL THE SERIAL NUMBER DATABASE INDEXED ON SITE NUMBER. DISPLAY
* THE EFFECTIVE DELIVERY ORDER DATES FOR THE USER TO SELECT FROM.
* CALL SERIAL NUMBER DATABASE INDEXED ON EFFECTIVE DELIVERY ORDER DATE
* AND SITE NUMBER. COPY TO TEMPONE, INDEXED ON FEATURE NUMBER. RELATE
* TO THE DESCRIPTION FILE AND PRODUCE REPORT.

SET ESCAPE OFF
SET TALK OFF
SET COLOR TO W+R, W+R, B
CLEAR
USE SERIALNO
GO TOP
IF EOF() = .T. THEN
  SET COLOR TO W+R, W+R
  "11,2,2, SAY " The SERIAL NUMBER Database is EMPTY! 
DO DELAY
RETURN
ENDIF
12 FLASH + "S.REPORTS.SCR"
24,0 SAY SPACE(80)
SET COLOR TO R+/ , R+/R
2,26 SAY " SITE SERIAL NUMBER REPORT 
SET COLOR TO W+/BR, W+/BR

402
@ 13,15 SAY "Enter site number for which the report is desired:"

* ENSURE THAT TEMPORARY DATABASE AND INDEX DO NOT EXIST,

* IF SO ERASE THEN

SET CONSOLE OFF
ERASE TEMPONE.DBF
ERASE TEMPONE.NDX
SET CONSOLE ON
USE SERIALNO INDEX SERNOSIT

DO WHILE .T.
  SET COLOR TO /BR, /BR
  STORE LOSITE TO MSITE
  @ 13,66 GET MSITE PICT '99'
  READ IF .NOT. (MSITE >= LOSITE .AND. MSITE <= HISITE) THEN
  SET COLOR TO W+/R, W+/R
  IF 24,22 SAY ERROR
  DO DELAY
  LOOP

ELSE
  GO TOP
  FIND &MSITE
  IF EOF() = .T. THEN
  STORE "No serial numbers exist for site " + MSITE +;
  "try another site" TO MESSAGE
  SET COLOR TO W+/R, W+/R
  IF 24,13 SAY MESSAGE
  DO DELAY
  LOOP
  ELSE
  EXIT
  ENDIF
  ENDIF

END DO WHILE .T.

SET COLOR TO W+/BR, W+/BR
@ 13,15 SAY SPACE(60)

SET COLOR TO W+/B, W+/B
@ 05,09 SAY "The following Delivery Order Effective Dates exist for Site"
@ 05,09 SAY MSITE
SET COLOR TO /BR, /BR
@ 13,05 SAY SPACE(70)
STORE 1 TO COUNT
STORE 1,000 TO COUNT

END
APPENDIX B: MAINTENANCE MANUAL

Page 3

SNODTRPT.PRG Program Listing

101 DO WHILE SITENO = MSITE
102 IF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00) THEN
103 *LINECT+6,57 SAY EFFDATE
104 ELSE
105 IF (COLCNT - (COLCNT * (COLCNT/2)) = 0.00) THEN
106 *LINECT+6,38 SAY EFFDATE
107 ELSE
108 ENDIF
109 *LINECT+6,19 SAY EFFDATE
110 ENDIF
111 ELSE
112 IF (COLCNT - (COLCNT * (COLCNT/3)) = 0.00) THEN
113 *LINECT = 1 + *LINECT
114 COLCNT = 1.00
115 ELSE
116 COLCNT = COLCNT + 1.00
117 ENDIF
118 STORE EFFDATE TO MOLDATE
119 DO WHILE ((EFFDATE = MOLDATE).AND. .NOT. EOF())
120 *LINECT+1,00 SAY EFFDATE
121 ELSE
122 *LINECT+1,00 SAY EFFDATE
123 ELSE
124 IF EOF() THEN
125 EXIT
126 ELSE
127 *LINECT+1,00 SAY EFFDATE
128 ENDIF
129 ENDDO WHILE SITENO = MSITE
130 *LINECT+1,00 SAY EFFDATE
131 STORE DTOC(DATE()) TO SYSDATE
132 STORE SUBSTR(SYSDATE,7,2) + SUBSTR(SYSDATE,1,2) +;
133 STORE (SYSDATE,4,2) TO MDATE
134 STORE SPACE(17) + 'Input Effective Date (Range ' + MDATE +
135 ' to ' + HIDATE + ') ' + SPACE(17) TO MESSAGE
136 SET COLOR TO /W, /W
137 @ 24,0 SAY MESSAGE
138 SET COLOR TO W+/B, W+/B
139 @ 3,29 SAY "EFFECTIVE DATE: "
140 USE SERIALNO INDEX SERNODAT
141 STORE "999999" TO MOLDATE
142 STORE "000000" TO MOLDATE
143 DO WHILE NOT. (MOLDATE <= LDATE).AND. MOLDATE <= HIDATE)
144 STORE MDATE TO MOLDATE
145 SET COLOR TO R+/B, R+/B
146 @ 3,45 GET MOLDATE PICT "999999"
147 READ
148 L) WHILE .T.
149 IF .NOT. (SUBSTR(MOLDATE,1,2) = "8") .AND.;
APPENDIX B: MAINTENANCE MANUAL

Page 4

SNODTRPT.PRG Program Listing

151 SUBSTR(MOLDATE,1,2) <= "99" THEN
152 SET COLOR TO W/B, W/B
153 9 24,0 SAY SPACE(80)
154 SET COLOR TO W+/R, W+/R
155 9 24,16 SAY " Year portion of date must be between 84 and 99 ")
156 DO DELAY
157 SET COLOR TO /W, /W
158 9 24,0 SAY MESSAGE
159 STORE SUBSTR(MDATE,1,2) TO MYEAR
160 SET COLOR TO R+/B, R+/B
161 9 3,47 GET MYEAR PICT "99"
162 READ
163 STORE MYEAR + SUBSTR(MOLDATE,3,4) TO MOLDATE
164 ELSE
165 EXIT
166 ENDIF
167 ENDDO WHILE .T.
168 *
169 DO WHILE .T.
170 IF .NOT. (SUBSTR(MOLDATE,3,2) >= "01" .AND.;
171 SUBSTR(MOLDATE,3,2) <= "12") THEN
172 SET COLOR TO W/B, W/B
173 9 24,0 SAY SPACE(80)
174 SET COLOR TO W+/R, W+/R
175 9 24,16 SAY " Month portion of date must be between 01 and 12 ")
176 DO DELAY
177 SET COLOR TO /W, /W
178 9 24,0 SAY MESSAGE
179 STORE SUBSTR(MDATE,3,2) TO MMONTH
180 SET COLOR TO R+/B, R+/B
181 9 3,47 GET MMONTH PICT "99"
182 READ
183 STORE SUBSTR(MOLDATE,1,2) + MMONTH +;
184 SUBSTR(MOLDATE,5,2) TO MOLDATE
185 ELSE
186 EXIT
187 ENDIF
188 ENDDO WHILE .T.
189 *
190 DO WHILE .T.
191 IF (SUBSTR(MOLDATE,3,2) = "04" .OR. SUBSTR(MOLDATE,3,2) = "09") .AND.;
192 SUBSTR(MOLDATE,5,2) = "01") .AND. .NOT.;
193 (SUBSTR(MOLDATE,5,2) = "01") .AND. SUBSTR(MOLDATE,5,2) = "30") THEN
194 SET COLOR TO W/B, /W
195 9 24,0 SAY SPACE(80)
196 SET COLOR TO W+/R, W+/R
197 9 24,16 SAY " Day portion of date must be between 01 and 30 ")
198 DO DELAY
199 SET COLOR TO /W, /W
200 9 24,0 SAY MESSAGE
APPENDIX B: MAINTENANCE MANUAL

Page 5

SNODTRPT.PRG Program Listing

201    STORE SUBSTR(MDATE,5,2) TO MDAY
202    SET COLOR TO R+/B, R+B
203    $ 3,49 GET MDAY PICT "99"
204    READ
205    STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
206    LOOP
207    ELSE
208 *     IF (SUBSTR(MOLDATE,3,2) = "02" .AND. .NOT.;
209 (SUBSTR(MOLDATE,5,2) >= "01" .AND.;
210     SUBSTR(MOLDATE,5,2) <= "28") THEN
211     SET COLOR TO W/B, W/B
212 $ 24,0 SAY SPACE(80)
213     SET COLOR TO W+/R, W+/R
214 $ 24,16 SAY " Day portion of date must be between 01 and 28 "
215 DO DELAY
216     SET COLOR TO /W, /W
217 $ 24,0 SAY MESSAGE
218     $ 24,0 SAY MESSAGE
219     STORE SUBSTR(MDATE,5,2) TO MDAY
220     SET COLOR TO R+/B, R+B
221 $ 3,49 GET MDAY PICT "99"
222     READ
223     STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
224     LOOP
225    ELSE
226 *     IF .NOT. (SUBSTR(MOLDATE,3,2) = "01" .AND.;
227     SUBSTR(MOLDATE,5,2) <= "31") THEN
228     SET COLOR TO W/B, W/B
229 $ 24,0 SAY SPACE(80)
230     SET COLOR TO W+/R, W+/R
231 $ 24,16 SAY " Day portion of date must be between 01 and 31 "
232 DO DELAY
233     SET COLOR TO /W, /W
234 $ 24,0 SAY MESSAGE
235     $ 24,0 SAY MESSAGE
236     STORE SUBSTR(MDATE,5,2) TO MDAY
237     SET COLOR TO R+/B, R+B
238 $ 3,49 GET MDAY PICT "99"
239     READ
240     STORE SUBSTR(MOLDATE,1,4) + MDAY TO MOLDATE
241     LOOP
242    ELSE
243     EXIT
244 ENDIF
245     ENDIF
246     ENDIF
247     ENDDO WHILE .T.
248 *     DO TOP
249     STORE MSITE + MOLDATE TO MSKEY

406
APPENDIX B: MAINTENANCE MANUAL

Page 6

SNODIRPT.PRG Program Listing

251 FIND &MKEY
252 IF EOF() = .T. THEN
253 SET COLOR TO W/B, W/B
254 @ 24.0 SAY SPACE(80)
255 STORE " EFFECTIVE DATE " + MOLDATE + " does not exist for site " + MSITE + ", try another " TO NODATE
256 SET COLOR TO W+/R, W+/R
257 @ 24.10 SAY NODATE
258 DO DELAY
259 SET COLOR TO /W, /W
260 @ 24.0 SAY MESSAGE
261 STORE "000000" TO MOLDATE
262 LOOP
263 ELSE
264 EXIT
265 ENDDO
266 ENDWILE .NOT. (MOLDATE >= LODATE .AND. MOLDATE <= HIDATE)
267 *
268 SET COLOR TO W+/B, W+/B
269 @ 05.05 SAY SPACE(70)
270 @ 24.0 SAY SPACE(80)
271 *
272 CLEAR LISTING OF EFFECTIVE DATES FROM SCREEN
273 *
274 SET COLOR TO /BR, /BR
275 @ 07.2 SAY SPACE(76)
276 @ 08.2 SAY SPACE(76)
277 @ 09.2 SAY SPACE(76)
278 @ 10.2 SAY SPACE(76)
279 @ 11.2 SAY SPACE(76)
280 @ 12.2 SAY SPACE(76)
281 @ 13.2 SAY SPACE(76)
282 @ 14.2 SAY SPACE(76)
283 @ 15.2 SAY SPACE(76)
284 @ 16.2 SAY SPACE(76)
285 @ 17.2 SAY SPACE(76)
286 @ 18.2 SAY SPACE(76)
287 @ 19.2 SAY SPACE(76)
288 @ 20.2 SAY SPACE(76)
289 @ 21.2 SAY SPACE(76)
290 *
291 SET COLOR TO R+/ , R+/
292 @ 13.18 SAY " CREATING TEMPORARY DATABASE AND INDEX FILE "
293 *
294 COPY TO TEMPONE FOR SITEENO = "MSITE" .AND. EFDATE = "MOLDATE"
295 SELECT 1
296 USE TEMPONE
297 INDEX ON FEATURENO TO TEMPONE
298 SELECT 2
299 USE DESCRIPT INDEX DESCRIPT
APPENDIX B: MAINTENANCE MANUAL

Page 7

SNODTRPT.PRG Program Listing

301 SELECT TEMPONE
302 SET RELATION TO FEATURENO INTO DESCRIP
303 GO TOP
304 *
305 * CREATE THE SPLICE EQUIPMENT PROJECT REPORT AND CHECK IF THE REPORT
306 * IS TO BE PRINTED OR DISPLAYED ON THE SCREEN.
307 *
308 SET COLOR TO W/BR, W/BR
309 @ 13,15 SAY SPACE(60)
310 @ 13,16 SAY " Do you want a printed report? (Yes or No): "
311 SET COLOR TO /BR, /BR
312 @ 13,49 SAY ",Y"
313 @ 13,56 SAY ",N"
314 STORE ",N" TO ACCEPT
315 @ 13,62 GET ACCEPT PICT ",!"
316 READ
317 *
318 * ENSURE THAT THE USER'S RESPONSE IS EITHER ",Y" OR ",N"
319 *
320 DO WHILE .NOT. (ACCEPT = ",N" OR ACCEPT = ",Y")
321 IF .NOT. (ACCEPT = ",N" OR ACCEPT = ",Y") THEN
322 SET COLOR TO W/R, W/R
323 @ 24,24 SAY " Response must be either N or Y "
324 DO DELAY
325 STORE ",N" TO ACCEPT
326 ENDIF .NOT. (ACCEPT = ",N" OR ACCEPT = ",Y")
327 SET COLOR TO /BR, /BR
328 @ 13,62 GET ACCEPT PICT ",!"
329 READ
330 ENDDO WHILE .NOT. (ACCEPT = ",N" OR ACCEPT = ",Y")
331 *
332 SET COLOR TO /BR, /BR
333 @ 13,15 SAY SPACE(55)
334 *
335 IF ACCEPT = ",Y" THEN
336 ?? FLASH + "W.PRINTER/"
337 SET CONSOLE OFF
338 WAIT TO CHOICE
339 SET ONSCREEN ON
340 SET COLOR TO W/R, W/R
341 @ 22,10 SAY SPACE(60)
342 STORE SUBSTR(DATE,1,2) + ", " + SUBSTR(DATE,1,2) + ", "
343 SUBSTR(DATE,1,2) + " TO DATE:
344 ?? " TO PAGE:
345 ?? " TO NO PAGE
346 ?? " TO PRINT
347 ?? " TO EDIT
348 ?? " TO LIST
349 *
350 DO WHILE .NOT. BA()
APPENDIX B: MAINTENANCE MANUAL

Page 272

SNODTRPT.PRG Program Listing

351 DO WHILE (LINECT <= 60 .AND. .NOT. EOF())
352 @ LINECT,3 SAY SITENO PICT "99"
353 @ LINECT,7 SAY B->CLIN PICT "9999"
354 @ LINECT,15 SAY FEATURENO PICT "999999"
355 @ LINECT,24 SAY B->DESCRIPT PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!"
356 @ LINECT,52 SAY EFFDATE PICT "999999"
357 @ LINECT,65 SAY QTY PICT "999"
358 @ LINECT,70 SAY SERIALNO PICT "!!!"
359 LINECT = LINECT + 1
360 SKIP
361 ENDDO WHILE (LINECT <= 60 .AND. .NOT. EOF())
362
363 IF EOF() = .T. THEN
364 IF PAGENO > 1 THEN
365 @ 62,37 SAY "Page " + STR(PAGENO,2,0)
366 ENDIF PAGENO > 1
367 EJECT
368 SET DEVICE TO SCREEN
369 @ 13,25 SAY "FINISHED PRINTING THE REPORT"
370 DO DELAY
371 EXIT
372 ELSE
373 SET DEVICE TO SCREEN
374 @ 13,27 SAY "Printing Page Number " + STR(PAGENO) + 1,2,0) + ""
375 SET DEVICE TO PRINT
376 ENDIF EOF() = .T.
377
378 IF (LINECT > 60 .AND. PAGENO > 1) THEN
379 @ 62,37 SAY "Page " + STR(PAGENO,2,0)
380 ENDIF (LINECT > 60 .AND. PAGENO > 1)
381 @ 2,26 SAY "SITE SERIAL NUMBER REPORT"
382 @ 3,29 SAY "EFFECTIVE DATE: "
383 @ 3,45 SAY "MOLDATE"
384 @ 4,60 SAY "TODATE"
385 @ 6,52 SAY "EFFECT TOT QMPT SERIAL"
386 @ 7,2 SAY "SITE CLIN FEATURE# DESCRIPTION DATE"
387 @ 7,60 SAY "QTY QTY NUMBER"
388 @ 8,2 SAY "SITE CLIN FEATURE# DESCRIPTION DATE"
389 @ 8,51 SAY "SITE CLIN FEATURE# DESCRIPTION DATE"
390 PAGENO = PAGENO + 1
391 STORE 10 TO LINECT
392 ELSE
393 ENDDO WHILE .NOT. EOF()
394 ELSE
395 SET COLOR TO (R*/C , R*/C)
396 @ 4,52 SAY "EFFECT TOT QMPT SERIAL"
397 @ 5,2 SAY "SITE CLIN FEATURE# DESCRIPTION DATE"
398 @ 5,60 SAY "QTY QTY NUMBER"
399 SET COLOR TO (R*/W , R*/C)
400
APPENDIX B: MAINTENANCE MANUAL

Page 9

SNODTRPT.FRG Program Listing

401 | STORE 0 TO LINECT
402 | *
403 | DO WHILE NOT. EOF()
404 | DO WHILE LINECT < 15
405 |   # LINECT-7, 3 SAY STTENO PICT "99"
406 |   # LINECT-7, 7 SAY B->CLIN PICT "9999"
407 |   # LINECT-7,15 SAY FEATURENO PICT "999999"
408 |   # LINECT-7,24 SAY B->DESCRIPT PICT "!!!!!!!!!!!!!!!!!!!!!!!!!!!"
409 |   # LINECT-7,52 SAY EFFDATE PICT "999999"
410 |   # LINECT-7,60 SAY TOTALTY PICT "999"
411 |   # LINECT-7,65 SAY QTY PICT "999"
412 |   # LINECT-7,70 SAY SERIALNO PICT "!!!!!!!"
413 |   LINECT = LINECT + 1
414 |   SKIP
415 | IF EOF() = .T. THEN
416 |   SET COLOR TO W+/R, W+/R
417 |   # 24,18 SAY " End of File reached, Press any key to EXIT "
418 |   SET CONSOLE OFF
419 |   WAIT TO ACCEPT
420 |   SET CONSOLE ON
421 |   EXIT
422 | ENDIF EOF() = .T.
423 | ENDDO WHILE LINECT < 15
424 | *
425 | IF EOF() = .T. THEN
426 |   EXIT
427 | ENDIF EOF() = .T.
428 | SET COLOR TO R+/B, R+/B
429 | STORE "C" TO CHOICE
430 | # 22,57 GET CHOICE PICT ":";
431 | READ
432 | *
433 | ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
434 | *
435 | DO WHILE NOT. (CHOICE = "C" OR. CHOICE = "X")
436 | IF NOT. (CHOICE = "C" OR. CHOICE = "X") THEN
437 |   SET COLOR TO W+/R, W+/R
438 |   # 24,24 SAY " Response must be either C or X "
439 |   DO DELAY
440 |   STORE "C" TO CHOICE
441 | ENDF IF NOT. (CHOICE = "C" OR. CHOICE = "X")
442 | SET COLOR TO R+/B, R+/B
443 | # 22,57 GET CHOICE PICT "!
444 | READ
445 | ENDDO WHILE NOT. (CHOICE = "C" OR. CHOICE = "X")
446 | *
447 | DETERMINE IF THE USER WANTS TO QUIT OR CONTINUE
448 | *
449 | IF CHOICE = "C"
450 |   SET COLOR TO /BR, /BR
APPENDIX B: MAINTENANCE MANUAL

SNODTRPT.PRG Program Listing

451 | @ 07,2 SAY SPACE(76)
452 | @ 08,2 SAY SPACE(76)
453 | @ 09,2 SAY SPACE(76)
454 | @ 10,2 SAY SPACE(76)
455 | @ 11,2 SAY SPACE(76)
456 | @ 12,2 SAY SPACE(76)
457 | @ 13,2 SAY SPACE(76)
458 | @ 14,2 SAY SPACE(76)
459 | @ 15,2 SAY SPACE(76)
460 | @ 16,2 SAY SPACE(76)
461 | @ 17,2 SAY SPACE(76)
462 | @ 18,2 SAY SPACE(76)
463 | @ 19,2 SAY SPACE(76)
464 | @ 20,2 SAY SPACE(76)
465 | @ 21,2 SAY SPACE(76)
466 | STORE 0 TO LINECT

ELSE

468 | EXIT

469 | ENDDO WHILE .NOT. EOF()

470 | ENDDO WHILE .NOT. EOF()

471 | ENDIF ACCEPT = "Y"

473 | ERASE ALL TEMPORARY FILES CREATED DURING REPORT GENERATION

475 | CLOSE DATABASES

477 | SET CONSOLE OFF

478 | ERASE TEMPONE.DBF

479 | SET CONSOLE ON

480 | ERASE TEMPONE.NDX

481 | ERASE TEMPONE.NDX

482 | SET PRINT OFF

483 | RELEASE ALL LIKE M*, ACCEPT, CHOICE, COLCNT, LINECT, PAGENO,;

486 | RELEASE ALL LIKE M*, ACCEPT, CHOICE, COLCNT, LINECT, PAGENO,;

487 | SYSDATE, TODAY, TODATE

488 | RETURN

489 |****************************************************************************
PROCEDURE SNOPJRPT.PRG

AUTHORS
	LCDR EDWARD J. CASE, SC, USN
	LCDR WINSTON H. BUCKLEY, SC, USN
	LCDR ROBERT F. BRAVO, USN
	LCDR ROBERT L. BEARD III, SC, USN

PURPOSE
	PROVIDE THE USER A SPLICE SERIAL NUMBER PROJECT LEVEL REPORT.

INPUT FILES	SERIALNO.DBF, SERNOPRJ.NDX, DESCRIP.DBF, DESCRIP.NDX

OUTPUT FILES: NONE.

CALLED BY: PROJRPTS.PRG

MODULES CALLED: DELAY.PRG

LOCAL VARIABLES: ACCEPT, CHOICE, LINCT, PAGENO, TODAY, TUDATE

DATE LAST TIME MODIFIED: 27 December 1985

CASE SELECTION = 2 SERIAL NUMBER PROJECT LEVEL REPORT

CALL SERIAL NUMBER DATABASE INDEXED ON EFFECTIVE DATE, SITE NUMBER, AND FEATURE NUMBER. RELATE TO DESCRIP FILE ON FEATURENO.

SET ESCAPE OFF
SET TALK OFF
SET COLOR TO W+/B, W*/B, B
CLEAR
USE SERIALNO
GO TOP
IF EOF() = .T. THEN
SET COLOR TO W*/R, W+/R
13,22 SAY "The SERIAL NUMBER Database is EMPTY!"
DO DELAY
RETURN
ENDIF
?? FLASH "PROJRPTS.COM"
4,24,0 SAY SPACE(80)
SET DUP'DU' R*/R*/ R*/R*/ R*/R*/
4,2,10 SAY "EQUIPMENT SERIAL NUMBER PROJECT LEVEL REPORT"
SELECT
USE SERIALNO INDEX SERNOPRJ.NDX
SELECT
USE SERIALNO INDEX DESCRIP
SELECT
USE SERIALNO INDEX FEATURE
SELECT
CLOSE SERIALNO
CLOSE SERNOPRJ
CLOSE DESCRIP
CLOSE FEATURE

CREATE THE SPLICE SERIAL NUMBER PROJECT REPORT AND CHECK IF THE REPORT IS TO BE PRINTED OR DISPLAYED ON THE SCREEN.

SET COLOR TO W+/BR, W+/BR
@ 13,16 SAY "Do you want a printed report? (Yes or No): "
SET COLOR TO /BR, /BR
@ 13,49 SAY "Y"
@ 13,56 SAY "N"
STORE "N" TO ACCEPT
@ 13,62 GET ACCEPT PICT ":"
READ

ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"

DO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
  IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
    SET COLOR TO W+/R, W+/R
    @ 24,24 SAY "Response must be either N or Y"
    DO DELAY
    STORE "N" TO ACCEPT
  ENDIF
SET COLOR TO /BR, /BR
@ 13,62 GET ACCEPT PICT ":"
READ

IF ACCEPT = "Y" THEN
  ?? FLASH + "W.PRINTER/
  SET CONSOLE OFF
  WAIT TO CHOICE
  SET CONSOLE ON
  SET COLOR TO W/B, W/B
  @ 22,10 SAY SPACE(65)
  STORE 0 TO PAGENO
  STORE 61 TO LINECT
  STORE DTODATE() TO TODAY
  STORE SUBSTR(TODAY,4,2) + " " + "OCT"(DATE()) + " 19", 7,2 TO 'TODATE
  SET COLOR TO R+/R+, R+/R+
  SET DEVICE TO PRINT
  DO WHILE .NOT. EOF()
    DO WHILE (LINECT <= 60 .AND. .NOT. EOF())
      @ LINECT, 3 SAY SPACE(30)
      @ LINECT, 7 SAY DESCRIP->CLIN
      @ LINECT, 15 SAY FEATUREN
  ENDWHILE
APPENDIX B: MAINTENANCE MANUAL

Page 3

SNOPJRPT.PRG Program Listing

101 @ LINECT,24 SAY DESCRIP->DESCRIP
102 @ LINECT,52 SAY EFFDATE
103 @ LINECT,60 SAY TOTQTY
104 @ LINECT,65 SAY QTY
105 @ LINECT,70 SAY SERIALNO
106 LINECT = LINECT + 1
107 SKIP
108 ENDDO WHILE
109 *
110 IF EOF() = .T. THEN
111 IF PAGENO > 1 THEN
112 @ 62,37 SAY "Page " + STR(PAGENO),2,0)
113 ENDF
114 EJECT
115 SET DEVICE TO SCREEN
116 @ 13,25 SAY "FINISHED PRINTING THE REPORT"
117 DO DELAY
118 EXIT
119 ELSE
120 SET DEVICE TO SCREEN
121 @ 13,27 SAY "FINISHED PRINTING THE REPORT"
122 DO DELAY
123 EXIT
124 *
125 IF LINECT > 60 .AND. PAGENO > 1 THEN
126 @ 62,37 SAY "Page " + STR(PAGENO),2,0)
127 ENDF
128 @ 2,18 SAY "EQUIPMENT SERIAL NUMBER PROJECT LEVEL REPORT"
129 @ 4,62 SAY TOTDATE
130 @ 6,52 SAY "EFFECT TOT COMPT SERIAL"
131 @ 7,2 SAY "SITE CLIN FEATURE# DESCRIPTION DATE"
132 @ 7,60 SAY "QTY QTY NUMBER"
133 @ 8,2 SAY "="
134 @ 8,51 SAY "="
135 PAGENO = PAGENO + 1
136 STORE 10 TO LINECT
137 *
138 ENDDO WHILE .NOT. EOF()
139 *
140 ELSE
141 SET COLOR TO GR+/B, GP+/B
142 @ 4,52 SAY "EFFECT TOT COMPT SERIAL"
143 @ 5,2 SAY "SITE CLIN FEATURE# DESCRIPTION DATE"
144 @ 5,60 SAY "QTY QTY NUMBER"
145 SET COLOR TO /B, /B
146 STORE 0 TO LINECT
147 *
148 DO WHILE .NOT. EOF()
149 DO WHILE LINECT < 15
150 @ LINECT+7,3 SAY SITE#
APPENDIX B: MAINTENANCE MANUAL

Page 4

SNOPJRPT.PRG Program Listing

151 @ LINECT+7,7 SAY DESCRIP->CLIN
152 @ LINECT+7,15 SAY FEATURENO
153 @ LINECT+7,24 SAY DESCRIP->DESCRIPT
154 @ LINECT+7,52 SAY EFFDATE
155 @ LINECT+7,60 SAY TOTQTY
156 @ LINECT+7,65 SAY QTY
157 @ LINECT+7,70 SAY SERIALNO
158 LINECT = LINECT + 1
159 SKIP
160 IF EOF() = .T. THEN
161 SET COLOR TO W+/R, W+/R
162 @ 24,18 SAY " End of File reached, Press any key to EXIT "
163 SET CONSOLE OFF
164 WAIT TO ACCEPT
165 SET CONSOLE ON
166 EXIT
167 ENDF
168 ENDDO WHILE LINECT < 15
169 *
170 IF EOF() = .T. THEN
171 EXIT
172 ENDF
173 SET COLOR TO R+/B, R+/B
174 STORE "C" TO CHOICE
175 @ 22,57 GET CHOICE PICT "C"
176 READ
177 *
178 ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
179 *
180 DO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
181 IF .NOT. (CHOICE = "C" .OR. CHOICE = "X") THEN
182 SET COLOR TO W+/R, W+/R
183 @ 24,24 SAY " Response must be either C or X "
184 DO DELAY
185 STORE "C" TO CHOICE
186 ENDF
187 SET COLOR TO R+/B, R+/B
188 @ 22,57 GET CHOICE PICT "C"
189 READ
190 ENDDO
191 *
192 DETERMINE IF THE USER WANTS TO QUIT OR CONTINUE
193 *
194 IF CHOICE = "C"
195 SET COLOR TO /BR, /BR
196 @ 07,2 SAY SPACE(76)
197 @ 08,2 SAY SPACE(76)
198 @ 09,2 SAY SPACE(76)
199 @ 10,2 SAY SPACE(76)
200 @ 11,2 SAY SPACE(76)
APPENDIX B: MAINTENANCE MANUAL

SNOPJRPT.PRG Program Listing

```plaintext
201 @ 12,2 SAY SPACE(76)
202 @ 13,2 SAY SPACE(76)
203 @ 14,2 SAY SPACE(76)
204 @ 15,2 SAY SPACE(76)
205 @ 16,2 SAY SPACE(76)
206 @ 17,2 SAY SPACE(76)
207 @ 18,2 SAY SPACE(76)
208 @ 19,2 SAY SPACE(76)
209 @ 20,2 SAY SPACE(76)
210 @ 21,2 SAY SPACE(76)
211 STORE 0 TO LINECT
212 ELSE
213 EXIT
214 ENDIF
215 *
216 ENDO WHILE .NOT. EOF()
217 *
218 ENDIF
219 *
220 * RETURN TO CALLING PROGRAM
221 *
222 SET PRINT OFF
223 RELEASE ACCEPT, CHOICE, LINECT, PAGENO, TODAY, TODATE
224 CLOSE DATABASES
225 RETURN
226 ***********************************************************************
```
SNOSTRPT.PRG Program Listing

* PROCEDURE SNOSTRPT.PRG

* AUTHORS
  LCDR EDWARD J. CASE, SC, USN
  LCDR WINSTON H. DUCKLEY, SC, USN
  LCDR ROBERT F. BADO, USN
  LCDR ROBERT L. BEARD III, SC, USN

* PURPOSE
  PROVIDE THE USER A SPLICE SERIAL NUMBER SITE LEVEL REPORT.

* INPUT FILES
  SERIALNO.DBF, SERNOSIT.NDX, DESCRIPT.DBF,
  DESCRIPT.NDX

* CALLED BY
  SITERPTS.PRG

* MODULES CALLED
  DELAY.PRG

* GLOBAL VARIABLE:
  HISITE, LOSITE

* LOCAL VARIABLES:
  ACCEPT, CHOICE, ERROR, LINECT, MESSAGE, MSITE,
  PAGENO, TODAY, TODATE

* DATE LAST TIME MODIFIED
  DECEMBER 1985

* CASE SELECTION = 3

SET ESCAPE OFF
SET TALK OFF
SET COLOR TO W+/B, W+/B, B
CLEAR
USE SERIALNO
GO TOP
IF EOF() = .T. THEN
  SET COLOR TO W+/R, W+/R
  @ 13,22 SAY "The SERIAL NUMBER Database is EMPTY!"
  DO DELAY
  RETURN
ENDIF
?? FLASH + "S.REPORTS.SCR/"
@ 24,0 SAY SPACE(80)
SET COLOR TO R+/ , R+/R
@ 2,26 SAY "SITE SERIAL NUMBER REPORT"
SET COLOR TO W+/BR, W+/BR
@ 13,15 SAY "Enter site number for which the report is desired:"
CALL SERIAL NUMBER DATABASE INDEXED ON SITE NUMBER,
  FEATURE NUMBER AND SERIAL NUMBER. RELATE TO DESCRIPTION FILE.
SELECT 1
USE SERIALNO INDEX SERNOSIT.NDX
SELECT 2
USE DESCRIP INDEX DESCRIP
SELECT SERIALNO
SET RELATION TO FEATURENO INTO DESCRIP
*
DO WHILE .T.
SET COLOR TO /BR, /BR
STORE LOSITE TO MSITE
@ 13,66 GET MSITE PICT '99'
READ
IF .NOT. (MSITE >= LOSITE AND MSITE <= HISITE) THEN
SET COLOR TO W+R, W+R
STORE 'Response must be between '+ LOSITE +';
' and '+ HISITE +' TO ERROR
@ 24,22 SAY ERROR
DO DELAY
LOOP
ELSE
DO TOP
FIND MSITE
IF EOF() = .T. THEN
STORE "No serial numbers exist for site" + MSITE +;
', try another site" TO MESSAGE
SET COLOR TO W+R, W+R
@ 24,13 SAY MESSAGE
DO DELAY
LOOP
ELSE
EXIT
ENDIF .NOT. (MSITE >= LOSITE AND MSITE <= HISITE)
ENDO WHILE .T.
*
SET COLOR TO W+R, W+R
@ 13,15 SAY SPACE(60)
*
ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
*
@ 13,16 SAY "Do you want a printed report? (Yes or No): "
SET COLOR TO /BR, /BR
@ 13,49 SAY "Y"
@ 13,56 SAY "N"
STORE "N" TO ACCEPT
@ 13,62 GET ACCEPT PICT ":!"
READ
* ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
*
APPENDIX B: MAINTENANCE MANUAL

Page 3

SNOSTRPT.PRG Program Listing

101 IF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y") THEN
102 SET COLOR TO W+/R, W+/R
103 @ 24,24 SAY "Response must be either N or Y"
104 DO DELAY
105 STORE "N" TO ACCEPT
106 ENDIF .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
107 SET COLOR TO /BR, /BR
108 @ 13,62 GET ACCEPT PICT ":" READ
109 ENDDO WHILE .NOT. (ACCEPT = "N" .OR. ACCEPT = "Y")
110 IF ACCEPT = "Y" THEN
111 ?? FLASH + "W.PRINTER/"
112 SET CONSOLE OFF
113 WAIT TO CHOICE
114 SET CONSOLE ON
115 SET COLOR TO W/B, W/B
116 @ 22,10 SAY SPACE(65)
117 DO WHILE *NOT. BOF()
118 LINECT,3 SAY SITENO
119 LINECT,7 SAY DESCRIP -> CLIN
120 LINECT,15 SAY FEATURENO
121 LINECT,24 SAY DESCRIP -> DESCIPT
122 LINECT,52 SAY EFFDATE
123 LINECT,60 SAY TOTQTY
124 LINECT,65 SAY (ITY
125 LINECT,70 SAY SERIALNO
126 LINECT = LINECT + 1
127 SKIP
128 ENDDO WHILE WHILE (LINECT <= 60 .AND. .NOT. EOF())
129 DO WHILE (LINECT <= 60 .AND. .NOT. EOF())
130 LINECT,3 SAY SITENO
131 LINECT,7 SAY DESCRIP -> CLIN
132 LINECT,15 SAY FEATURENO
133 LINECT,24 SAY DESCRIP -> DESCIPT
134 LINECT,52 SAY EFFDATE
135 LINECT,60 SAY TOTQTY
136 LINECT,65 SAY (ITY
137 LINECT,70 SAY SERIALNO
138 LINECT = LINECT + 1
139 SKIP
140 ENDDO WHILE WHILE (LINECT <= 60 .AND. .NOT. EOF())
141 IF EOF() = .T. THEN
142 IF PAGENO > 1 THEN
143 @ 62,37 SAY "Page " + STR(PAGENO, 2, 0)
144 ENDIF PAGENO > 1
145 EJECT
146 SET DEVICE TO SCREEN
147 @ 13,25 SAY "FINISHED PRINTING THE REPORT"
APPENDIX B: MAINTENANCE MANUAL  Page 283

SNOSTRPT.PRG Program Listing

DO DELAY
EXIT

ELSE
SET DEVICE TO SCREEN
@ 13,27 SAY " Printing Page Number " + STR(PAGENO + 1,2,0) + " "
SET DEVICE TO PRINT
ENDIF EOF() = .T.

IF (LINECT > 60 ,AND. PAGENO > 1) THEN
@ 62,37 SAY "Page " + STR(PAGENO,2,0)
ENDIF (LINECT > 60 ,AND. PAGENO > 1)
@ 2,26 SAY " SITE SERIAL NUMBER REPORT "
@ 4,60 SAY TODATE
@ 6,52 SAY "EFFECT TOT COMPT SERIAL"
@ 7,2 SAY "SITE CLIN FEATURE# DESCRIPTION DATE"
@ 7,60 SAY "QTY QTY NUMBER"
@ 8,2 SAY "=================================="
@ 8,51 SAY "=================================="
PAGENO = PAGENO + 1
STORE 10 TO LINECT

ENDDO WHILE .NOT. EOF()

ELSE
SET COLOR TO GR+/B, GR+/B
@ 4,52 SAY "EFFECT TOT COMPT SERIAL"
@ 5,2 SAY "SITE CLIN FEATURE# DESCRIPTION DATE"
@ 5,60 SAY "QTY QTY NUMBER"
SET COLOR TO /BR, /BR
STORE 0 TO LINECT

DO WHILE .NOT. EOF()
DO WHILE LINECT < 15
@ LINECT+7,3 SAY SITENO
@ LINECT+7,7 SAY DESCR->CLIN
@ LINECT+7,15 SAY FEATURENO
@ LINECT+7,24 SAY DESCR->DESCRIPT
@ LINECT+7,52 SAY EFFDATE
@ LINECT+7,60 SAY TOTDATE
@ LINECT+7,65 SAY QTY
@ LINECT+7,70 SAY SERIALNO
LINECT = LINECT + 1
SKIP
IF EOF() = .T. THEN
SET COLOR TO W+/R, W+/R
@ 24,18 SAY " End of File reached, Press any key to EXIT "
SET CONSOLE OFF
WAIT TO ACCEPT
SET CONSOLE ON
EXIT
ENDIF EOF() = .T.
202 ENDDO WHILE LINECT < 15
203 *
204 IF EOF() = .T. THEN
205 EXIT
206 ENDDO EOF() = .T.
207 SET COLOR TO R+/B, R+/B
208 STORE "C" TO CHOICE
209 @ 22,57 GET CHOICE PICT ":"
210 READ
211 *
212 * ENSURE THAT THE USER'S RESPONSE IS EITHER "Y" OR "N"
213 *
214 DO WHILE .NOT. (CHOICE = "C" .OR. CHOICE = "X")
215 IF .NOT. (CHOICE = "C" .OR. CHOICE = "X") THEN
216 SET COLOR TO W+/R, W+/R
217 @ 24,24 SAY "Response must be either C or X"
218 DO DELAY
219 STORE "C" TO CHOICE
220 ENDIF .NOT. (CHOICE = "C" .OR. CHOICE = "X")
221 SET COLOR TO R+/B, R+/B
222 @ 22,57 GET CHOICE PICT ":"
223 READ
224 ENDDO WHILE . NOT. (CHOICE = "C" . OR. CHOICE = "X")
225 *
226 * DETERMINE IF THE USER WANTS TO QUIT OR CONTINUE
227 *
228 IF CHOICE = "C"
229 SET COLOR TO /BR, /BR
230 @ 07,2 SAY SPACE(76)
231 @ 08,2 SAY SPACE(76)
232 @ 09,2 SAY SPACE(76)
233 @ 10,2 SAY SPACE(76)
234 @ 11,2 SAY SPACE(76)
235 @ 12,2 SAY SPACE(76)
236 @ 13,2 SAY SPACE(76)
237 @ 14,2 SAY SPACE(76)
238 @ 15,2 SAY SPACE(76)
239 @ 16,2 SAY SPACE(76)
240 @ 17,2 SAY SPACE(76)
241 @ 18,2 SAY SPACE(76)
242 @ 19,2 SAY SPACE(76)
243 @ 20,2 SAY SPACE(76)
244 @ 21,2 SAY SPACE(76)
245 STORE 0 TO LINECT
246 ELSE
247 EXIT
248 ENDIF CHOICE = "C"
249 *
250 ENDDO WHILE . NOT. EOF()
SNOSTRPT.PRG Program Listing

251   *
252  ENDIF ACCEPT = "y"
253   *
254  * RETURN TO CALLING PROGRAM
255   *
256  SET PRINT OFF
257  RELEASE ACCEPT, CHOICE, ERROR, LINECT, MESSAGE, MSITE, PAGENO,;
258        TODAY, TODATE
259  CLOSE DATABASES
260  RETURN
261  ************************************************************
BIBLIOGRAPHY


423
<table>
<thead>
<tr>
<th>No. Copies</th>
<th>Distribution List</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145</td>
</tr>
<tr>
<td>2</td>
<td>Library (Code 0142) Naval Postgraduate School Monterey, CA 93943-5002</td>
</tr>
<tr>
<td>1</td>
<td>Computer Technology Curriculum Officer (Code 37) Naval Postgraduate School Monterey, CA 93943-5000</td>
</tr>
<tr>
<td>2</td>
<td>Barry A. Frew (Code 54Fw) Naval Postgraduate School Monterey, CA 93943-5000</td>
</tr>
<tr>
<td>2</td>
<td>Commanding Officer Naval Supply Center (Code 60) Attn: LCDR Robert L. Beard III Charleston, SC 29408-5000</td>
</tr>
<tr>
<td>1</td>
<td>Commander Naval Supply Systems Command Attn: LCDR Ted Case (SUP 043A) Washington, DC 20376-5000</td>
</tr>
<tr>
<td>1</td>
<td>Commanding Officer Navy Fleet Material Support Office Attn: LCDR Winston Buckley P.O. Box 2010 Mechanicsburg, PA 17055-0787</td>
</tr>
</tbody>
</table>