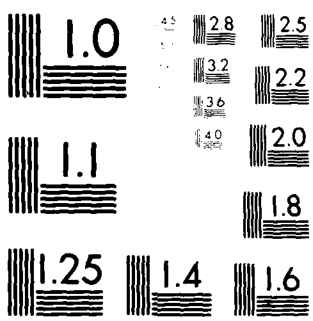
AD-A088 021 INCO INC MCLEAN VA F/G 9/2
USAF STANDARD SOFTWARE BASE.(U)
MAY 80 G KINCAID, R HICKS, W FULLERTON F30602-78-C-0204
UNCLASSIFIED INCO/1110-480-FR-16-D(F) RADC-TR-80-184 NL

END
DATE
FILMED
9-80
DTIC



MICROCOPY RESOLUTION TEST CHART

LEVEL II

(12)

B.S.

RADC-TR-80-184
Final Technical Report
May 1980

AD A088021

USAF STANDARD SOFTWARE BASE

INCO, Inc.

Gary Kincaid
Richard Hicks
William Fullerton
Gary Morgan
Brian Wolf
Thomas Trump

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

80 8 15 044

This report has been reviewed by the RADC Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-80-184 has been reviewed and is approved for publication.

APPROVED: Fred Haritatos

FRED HARITATOS
Project Engineer

APPROVED:

OWEN R. LAWTER, Colonel, USAF
Chief, Intelligence and Reconnaissance Division

FOR THE COMMANDER: John P. Huss

JOHN P. HUSS
Acting Chief, Plans Office

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: RADC-TR-80-184	2. GOVT ACCESSION NO.: AD-A088 021	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): USAF STANDARD SOFTWARE BASE		5. TYPE OF REPORT & PERIOD COVERED: Final Technical Report, Mar 78 - Mar 80
		6. PERFORMING ORG. REPORT NUMBER: INCO/1110-480-FR-16-D(F)
7. AUTHOR(s): Gary Kincaid, Richard Hicks, William Fullerton, Gary Morgan, Brian Wolf, Thomas Trump		8. CONTRACT OR GRANT NUMBER(s): F30602-78-C-0204
9. PERFORMING ORGANIZATION NAME AND ADDRESS: INCO, Inc., 7916 Westpark Drive, McLean VA 22102		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 31025F, 21830151
11. CONTROLLING OFFICE NAME AND ADDRESS: Rome Air Development Center (IRDE), Griffiss AFB NY 13441		12. REPORT DATE: May 1980
		13. NUMBER OF PAGES: 44
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Same		15. SECURITY CLASS. (of this report): UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer: Fred Haritatos (IRDE)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Message Processing
Standard Software Base
Terminal Transparent Display Language
Gateway Programs
AUTODIN

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the work performed to expand the capabilities of the USAF Standard Software Base supporting AN/GYQ-21(V) operations. The report discusses maintenance and improvements made to the baseline system, and the development and implementation of advanced capabilities and services.

TABLE OF CONTENTS

EVALUATION

Under this effort, the Standard Software Base (SSB) system was expanded to improve operational support to AN/GYQ-21(V) systems. The SSB system provides intelligence analysts with the capability to receive, send, build, edit, store and receive message traffic.

The present SSB baseline system was enhanced under software release III to include improved AUTODIN routing capabilities, precompiled application program displays and the integration of related software features.

A key aspect of the program has been the emphasis on achieving a high degree of modularity that allows a user to select only those subsystems that benefit their individual operations. Basic subsystem capabilities include data management, data distribution, terminal support, communication gateways, and utility support.

System enhancements were made in response to DODIIS requirements on a common-user basis. This enabled software cost to be shared among the user community. Users are able to select software modules with assurance that these modules will (a) interact with other SSB modules, (b) receive and support unique site applications, (c) not impose a burden on his system operations, and (d) accrue benefits arising from other modernization programs.

This effort was completed under RADC Technical Planning Objective R1E.

Fred Haritatos
FRED HARITATOS
Project Engineer

SECTION 1

1.0 INTRODUCTION

1.1 Purpose. This technical report describes work performed for the Air Force Intelligence Service under the provisions of Rome Air Development Center Contract No. F30602-78-C-0204. The report discusses the technical accomplishments performed and identifies the documents provided to the Government.

1.2 Background. The Standard Software Base (SSB) is a major component of the United States Air Force Common User's Baseline for the Intelligence Community (CUBIC). The SSB is the result of intelligence data handling developments initiated by the Air Force Intelligence Service (AFIS/IND) in 1975 to help users of the AN/GYQ-21(V) to develop their individual computer systems. Development of the SSB has proceeded under a carefully managed program to provide network and communications processing capabilities, data management services, applications programs with which to access and control message traffic, and enhancements to the AN/GYQ-21(V) operating systems. Figure I-1 illustrates a typical SSB Release III system configuration as of February 1980.

1.3 Objectives. The primary purpose of the work performed under the current contract was to extend the development of the Standard Software Base (SSB). Five general objectives were established to guide the technical effort required to meet this goal. They are:

a. Maintenance and enhancements of the baseline SSB system.

b. Analysis, design, and implementation of advanced capabilities.

c. Support to potential SSB users in the configuration, installation, and implementation of SSB capabilities.

d. Configuration control procedures, SSB distribution services, and training support services.

e. Documentation of SSB capabilities.

1.4 Report Organization. The final technical report is organized according to the following schema:

- Section 1 - This section describes the purpose of the report and provides a synopsis of SSB objectives.
- Section 2 - Contains a discussion of the major technical tasks performed during the contract and lists the contract deliverables provided to the Government.

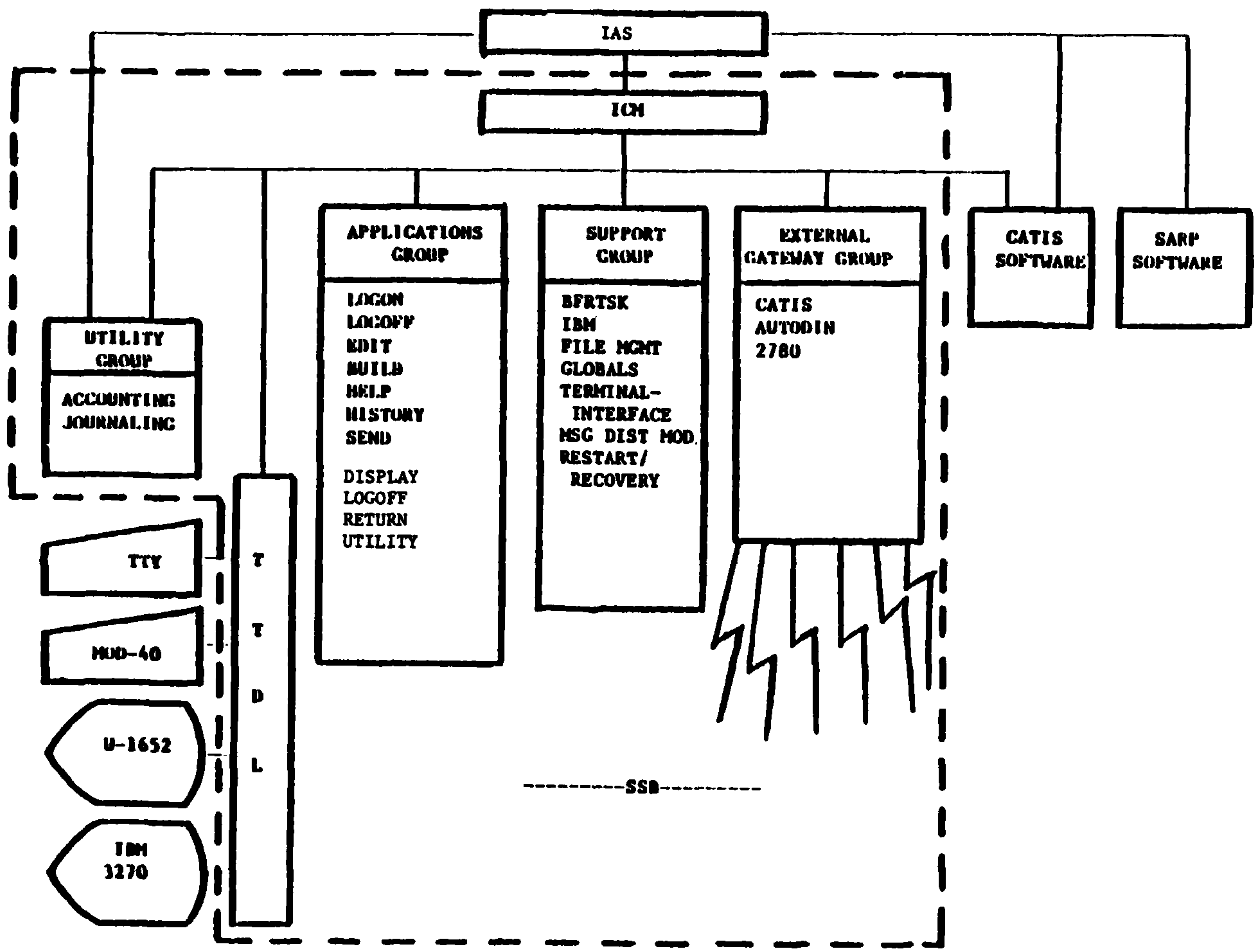


Figure I-1. SSB RELEASE III SYSTEM/SUBSYSTEM

- Section 3 - Discusses the installation and site support services provided during the contract.

- Section 4 - Describes the development of configuration management and quality control policies and procedures.

SECTION 2

2. TECHNICAL ACCOMPLISHMENTS

2.1 Maintenance of Baseline SSB/CATIS. The primary objective of this task was to correct software problems arising through operational use of SSB/CATIS systems. The responsibilities associated with this task included establishing a software incident data base and the means of producing software incident reports; providing centrally located software maintenance services; and maintaining the SSB technical documentation series.

2.1.1 Applications Programs.

a. Background. The SSB applications programs provide the machine functions required to process message traffic within the SSB system. Functions are available to display messages at an I/O terminal, direct them to the line printer, build, edit, and transmit messages, and to store messages for subsequent retrieval and use. Certain utility programs are included in the applications group. They provide the capability to transfer messages between queues and between users; maintain totals of messages on a user's queues; provide referral information concerning the classification and precedence attributes of the messages; and provide reports as to which users and terminals are active on the system. These programs were subjected to continuing development and refinement efforts to expand the range of services, reduce core and disk storage requirements, improve processing times, and make the user functions easier to operate.

b. Accomplishments.

(1) Build Function. The programs supporting the BUILD function and the EDIT function were combined. While the BUILD function still appears a unique operation to the user, it is now simply a portion of the editing program. This combination permits a user to use the features previously restricted to the Edit function alone. Text may be shortened, lengthened, rearranged, or deleted, using the editing commands available in the Edit program. Coding changes were made in the BLDSBA module to permit passing precedence and security values for a message header from a buffer rather than from a service request block (SRB). This change was required because the SRB was not large enough to accommodate additional information required by the addition of SI and SAO elements to the SSB message header. SI and SAO codeword processing is discussed in further detail in paragraph 2.2.3.

(2) Display Function. The DISPLAY and PRINT functions were combined under the DISPLAY function. The new function is supported by the SBADSP and DSPDRV programs. A wide range of printing options is now available in the new function. Included are the capabilities to:

- Print a message currently displayed on the terminal screen.

- Print one or more messsages without the necessity of first displaying the messages on the screen.

- Printing all messages currently on a user's queues.

- Deleting messages currently displayed on the terminal screen.

(3) HELP Function. This function is driven by the SBAHLP module. Minor changes in operating procedures were made to make the function easier to use. The new program automatically redisplays the selection menu when a function description has been completed. In previous editions of the program, the user had to enter a command in order to redisplay the menu, otherwise the function exited automatically. The menu selection was restricted to a single choice per iteration of the HELP cycle. Previous editions permitted the user to enter up to ten keywords in a command string to acquire function descriptions in the sequence in which they were entered. This feature resulted in a high incidence rate of typographical errors and was discontinued.

(4) The Edit Function. The Edit programs (SBAEDI and EDITOT) were completely rewritten. In addition to being expanded to accommodate the Build function, the number of editing options and services was greatly increased. The new features include:

- The ability to see the editing changes displayed as they are made. There was no such capability in the previous editor.

- The ability to use cursor control and other intelligent terminal features to make changes directly to the text displayed on the terminal screen. Previous versions of Edit required that changes be made indirectly through references and commands inserted on an editing command line.

- The ability to move backward as well as forward through the body of a message in order to access portions of text that had already been displayed at the terminal. No "back-up" capability was provided by earlier editing programs.

- The ability to create new pages of text between the existing pages of a message. In earlier editions of Edit, one could add pages only at the end of a message.

- A "cut and paste" capability which makes major text rearrangement easier.

Most of the remaining features in the new program consist of modifications and improvements to basic capabilities featured in earlier versions of the Edit program. These include the text insertion, deletion, and replacement functions, and improved techniques for line and page position controls. Other changes were required to accommodate handling SI and SAO codeword entries. These changes are described in paragraph 2.2.3., Codeword Processing.

(5) Application Program Conversion. The displays for each application program were rewritten to a central display library for permanent off-line storage. Concurrent with this effort, display formats, function commands, and parameter entry labels were standardized to improve clarity and make the user functions easier to understand and to operate. Additional modifications were completed to make the applications programs compatible with TTDL/III. Work in this subject area included replacing dynamic display definition calls with library retrieval calls and modifying any program logic dependent on execution-time display compilation. The technical effort expended in developing TTDL/III is described in paragraph 2.2.1, Terminal Interface Enhancements.

2.1.2 Center System Enhancements. Technical work performed on the supervisory center system modules of SSB was primarily directed at establishing compatibility between SSB and updated AN/GYQ-21(V) operating systems; RSX-11D Version 6.2 and later, the Interactive Applications System or simply, IAS.

a. Conversion to 6.2. The SSB system was modified to run under RSX-11D, Version 6-C (6.2) on both the PDP 11/45 and the PDP 11/70, with a variety of disk and magnetic-tape controllers and drivers. RSX system generation files were altered to reflect changes in the operating system, and to accommodate different device specifications. Several SSB batch jobs were also modified because of the change in operating systems. The teletype handler was modified to allow for external buffering provided by the new system.

Elements of the indirect MCR function released and supported under RSX-11M were incorporated into the SSB system to provide for this feature under RSX, V6.2.

b. Conversion to IAS. Some time after the conversion to 6.2 was complted, IAS conversion efforts began. All handlers were rebuilt against "HNDLIB", "TT16" was reconfigured, and virtually all SSB programs were re-taskbuilt.

The new capability provided by IAS includes: a larger node pool; multiple command language interpreter support; multi-user handlers; memory resilient overlays (used by the IAS executive for more efficient use of memory), and several new directives, e.g., mapping, and exiting with status.

2.1.3 Gateway Enhancements.

2.1.3.1 AUTODIN Transmission Gateway (TISGT1). The transmission portion of the AUTODIN gateway (TISGT1) was modified to transmit GENSER traffic in JANAP 128 format. The differences between DOI-103 and JANAP message formats were resolved by making changes to message formats as indicated below:

a. Format Line 2. For GENSER traffic, the normal DSSCS security prosign ('M') was replaced by a one-character representation (repeated four times) for the appropriate security classification, e.g., UUUU for unclassified, and CCCC for Confidential. The requirement for a card count was omitted since paper tape is the only valid format for GENSER traffic.

b. Format Line 4. When GENSER traffic is being transmitted, Format Line 4 will contain the appropriate ops-signal depending on message classification. That is, ZNR for unclassified messages, and ZNY for classified traffic. The character code for the classification (repeated five times) of the messsage being transmitted will immediately follow the ops-signal. For example:

ZNR UUUU identifies an unclassified message, while

ZNY SSSSS identifies a Secret message.

c. Format Line 11. For GENSER traffic, the normal ZEM character group preceding the first line of text in a DSSCS message, was replaced by the characters BT. These BT characters must also appear at the end of the text for GENSER messages. The requirement for the BT prefix and suffix does not apply to Query/Response messages.

d. Routing Indicator Checks. An additional safeguard was coded into the gateway to prevent inadvertant transmission of DSSCS messages over the GENSER lines. Checks are performed to ensure that only "Y" type routing indicators are accepted for transmission over the DSSCS network, and that only "R" type routing indicators are accepted for transmission over GENSER lines.

e. Network Dependent Site Requirements. The AUTODIN transmit gateway (TISGT1) processes either DSSCS messages or GENSER messages, and appears to the SSB system as if it were two separate gateways.

For DSSCS the network identifier is NET.AU, while for GENSER the network identifier is NET.GL. The network identifier must be consistent with the routing indicators in the message. The network identifier is specified by entering "AUTODIN" for DSSCS and "GENSER-TO-LINE" for GENSER in the SEND application program.

The following steps must be performed at a site to enable the site to originate GENSER messages:

- o Define the symbol C$GNSR in PRECO

- o The table of external routing indicators (ERIFIL) should have a GENSER type routing indicator for the site. The routing indicator should be six characters beginning with an "R".

- o Assemble and run COMFL, GOTFL, RIFFIL, ERIFIL, (this is done automatically by AT. @(13, 1) TISFILBLD)

Similarly, the following steps must be performed at a site to disable the site from originating GENSER messages.

- o The symbol C$GNSR should not be defined in PRECON.

- o The table of external routing indicators (ERIFIL) should not have a GENSER routing indicator for the site.

- o Assemble and run COMBIL, GOTFIL, RIFFIL, and ERIFIL with C$GNSR undefined.

2.1.3.2 AUTODIN Receive Gateway. The pilot line processing aspects of TISGTA were enhanced to differentiate between the pilot line of a "service message" and a "pilot line" for Format Line Two, which may indicate message disposition such as "suspected duplicates." The security resolution capabilities of the gateway were expanded to recognize the security parameters of the message in order to preclude unauthorized access.

In addition, the entire error recognition and reporting capability of the gateway were refined to improve effectiveness and clarity.

2.1.3.3 CATIS Gateway. The approach for the "dual-processor" CATIS capability has been altered from one based on a PTC ROM discipline to one involving a DDCMP discipline with modified handlers. Coordination is continuing under the enhancements contract with Bunker Ramo to resolve those interface problems remaining.

2.2 Implementation of Advanced Capabilities

2.2.1 Terminal Interface Enhancements.

a. Background. The Terminal Transparent Display Language software system is now in its third stage of development. Referred to as TTDL/III, it combines the features and capabilities of its predecessors, TTDL/I and TTDL/II. TTDL/I was developed as a prototype system to demonstrate the practicality of producing a generalized display language and to investiage the feasibility of providing dynamic display formatting for a diverse group of terminals. This efffort resulted in the development of a wide range of terminal independent capabilities and a user display system independent of the characteristics and restrictions of a specific terminal type. However, such broad capabilities and flexibility were achieved only through the classic tradeoff of capability versus performance. That is, the support of numerous terminals of diverse types exacted a toll in terms of CPU processing, disk time, RSX node pool usage, and the number of terminals which could actually be supported. Although TTDL/I was placed into operation and supported the development of SSB and CATIS, the system was excessively slow and placed a restriction on the number of terminals which could be supported. TTDL/II was developed to alleviate some of the limitations of TTDL/I, but even more specifically, to provide support for CATIS. For this reason, dynamic formatting and reformatting capabilities were eliminated and only the UNIVAC 1652 terminal was supported. TTDL/II proved to be an extremely stable system with greatly simplified internal data structures, improved service time, and a core management capability wich permitted the user to choose between response time and core usage. However, since TTDL/II was directed at a specific function its application was limited. what was needed was a system that combined the capabilities of both TTDL/I and TTDL/II with expanded performance and improved service times. TTDL/III was developed to meet this requirement. Several design goals were identified to control system development:

- Provide users of TTDL/I and TTDL/II a smooth transition to use TTDL/III
- Achieve the best possible service time
- Efficiently support a minimum of sixteen (16) terminals of a single type
- Minimize both core requirements and usage of the RSX-11D node pool
- Provide for future addition of, as yet undefined, terminal types

- Support diverse terminal types on a single processor

- Provide system generation capabilities to allow extensive system tailoring of required displays

- Provide support for the IBM 3270, the UNIVAC 1652, and the MOD-40 and other teletype terminals.

Based on these objectives, an optimum system was proposed to provide the operational environment within which TTDL/III would function.

b. Design Considerations. Air Force and INCO personnel met in February 1978 to consider alternatives and to specify design goals and operating environment. The Terminal Dependent Format concept of TTDL/II was selected in order to meet the primary requirements of processing speed and resource efficiency.

It was further decided to develop a multiple terminal system using synchronous alphanumeric polled terminals as the primary terminal type. The speed and efficiency of these devices is far superior to that of asynchronous terminals.

The requirement to support a dynamic display capability was dropped since its cost in terms of speed and efficiency outweighed the marginal benefits it could provide.

Further design improvements were examined for potential development. Among these were the retention of all TTDL/II capabilities, the ability to compile displays in an on-line mode and flash them to the terminal screen for review prior to storage in a display library, the capability for TTDL/III to be fully compatible for use in the PDP 11/45 as well as in the PDP 11/70, and the gradual degradation of resource allocation. The final target system agreed upon for development included:

- TDF type systems with pre-compiled displays

- Unique display libraries for unique terminal-type support

- Capability to flash displays to the terminal prior to display library storage

- Synchronous alphanumeric polled terminals designated as system primary terminal type

- All TTDL/II features carried over

o Operational on PDP 11/45 and 11/70 with gradual degradation of system resources.

c. Accomplishments. TTDL/III is a result of major modifications to TTDL/II and the development of several new modules. A compiler was written to compile displays, and a terminal spooler was written for each terminal type supported by TTDL/III.

The compiler module (TDCMPL), was designed specifically to be run in an off-line mode in order to conserve RSX system resources. However, the compiler may be run on-line by using Display Definition (DISDEF) or Compile application (CMPLAP) routines. When run in an on-line mode, the user can output his display to a terminal before saving it in the display library. This feature is unique to TTDL/III. Under TTDL/II, displays could only be compiled in UNIVAC 1652 format, a process which was performed off-line, and involved using many TTDL/I modules.

In TTDL/II, the terminal handler (TH) performed all terminal related processing. TH was a multi-user task which meant that there was a copy of TH resident in core for each active terminal. TTDL/III has made TH a single user task but has retained the quick response time of TTDL/II by removing from TH any code causing it to wait, and placing this code in the Terminal Spoolers. These spoolers are driven by AST's and wait only occasionally for a buffer. The processing of read/write to the terminals, assigning function keys, and paging functions were moved from TH to the individual spoolers; this processing usually varies with the hardware characteristics of each terminal type.

The Display Handler (DSPHAN) and the TH module were both modified to permit them to exit core when their respective functions are complete. This feature frees essential RSX system resources. The Build Screen Area and Restore subroutines of the Display Handler were rewritten for TTDL/III. Under the provisions of TTDL/II, a user was constrained by CATIS screen area requirements. In TTDL/III, the screen areas may be defined as the user wishes. Unlike TTDL/II, TTDL/III accommodates multiple-paged displays. In support of this feature, a checking routine was added to the Restore Display call to determine how many pages were contained in the display being restored.

The display field handling module (FIELDS) was modified to process fields for the IBM 3270, MOD 40, and teletypes. Fields for the teletype are processed as if they were being issued by the UNIVAC 1652.

Modifications were made to the common area of TTDL/II, to accommodate processing the different terminal types supported by TTDL/III. The IBM 3270 handler was modified and the Terminal Data Area had keywords added to it.

The Applications Interface Module was modified to save and to delete displays from the Page Work File, a feature required if the compiler is to be run on-line.

A new module was provided to print the contents of a terminal screen area on the system line printer. Operating in response to an assigned variable function key, this feature permits the user to acquire hard-copy printouts of the Data Text portion of the terminal screen.

The TTDL/II System Generation File was completely rewritten to allow extensive tailoring of the system based on site-unique requirements.

d. Documentation. A comprehensive two-volume programmer's manual was produced to describe the techniques used in programming under TTDL, and to describe the individual tasks that comprise the TTDL/III system. The manual has been added to the technical documentation series.

2.2.2 Interfaces to Other Networks and Systems. Two two-man trips were performed to Schierstein to conduct system maintenance and to resolve problems with TISGT6 processing of sectioned messages to tape, and problems with CATRUN. Trips were also made to Bunker Ramo in Westlake, California to update the TTDL/III system and to correct problemms in the "AIM" module. During this trip, all application programs were re-taskbuilt. Additional support was provided to TAC in January 1980 for Category III testing; support to PACAF in preparation of an installation; maintenance and file error resolution for FIC; and maintenance and update service for the SSB installation at AFAITC.

2.2.3 Codeword Processing.

a. Background. The AFIS requirement for the addition of two new security parameters in SSB messages was resolved during a two-stage technical effort. The two new parameters, the SI codeword, and the SAO codeword, had to be validated according to specific rules for all traffic within SSB, or transmitted to or received by SSB. Additionally, it was imperative that SSB internal security on all existing parameters be made completely reliable. It was decided to perform this work in two phases. The first phase would provide the services needed to build, edit, dislay, and print messages containing codeword parameters. The second and final phase would provide increased reliability for SSB security, additional validation criteria for the security monitor program, and full security checks in the Build/Edit program, the Transfer option in the Utility program, and the History Search and Retrive programs.

b. Processing Modifications.

(1) System Data Structures. The fixed header block of the basic SSB message logical block structure was revised to accommodate the codeword entires. The Block Sequence Number (H.BSN) was replaced by two fields to hold the SI and SAO codewords (H.CCMCW and H.CPCW). These two values were coded as a single Precedence and Security Table entry in order to provide a correspondence between the internal code used to store the values in the table, and the ASCII values used to display or print the codewords.

Three new parameters were added to the Transmission Control Character Table (TCC) in the AUTODIN Format Line File (Figure II-1) shows the TCC Table. These parameters permit validation of the SAO codeword, the minimum security level for the TCC, and the network identifier.

(2) Modifications to Programs and Files. Several system files and programs were modified to accommodate processing codewords. The components so affected are listed below:

SSB Global Routines: GBL6, a component of the SSB global routines used to pass information within the SSB system, was changed to remove all references to the block sequence number in the fixed header block. This allowed the codeword parameters to be added to the block without changing its size.

TPSGLB: Definitions of the symbolic parameters for SSB system data structures are contained in TPSBL and TPSGBL.OBJ. Revisions were made to add the codeword tables to the Precedence and Security File (PSTFIL) and to the fixed header block.

GENMAC and SSBMAC: The GENMAC portion of BIGMAC was modified to provide revised macros for PSTFIL and an expanded TCC macros for AFLFIL. The PSTFIL macros were revised to change the number of PSTFIL tables from 4 to 6 and to increase the length of the text field from 30 to 34 bytes. The TCC macros for AFLFIL were revised to allow three additional parameters.

The SMNCPC macro is SGBLMAC was changed to add two parameters in the call to the Security Monitor (SECMON). This macro is used in all modules which call the security monitor.

PSTFIL: The precedence/security tables in PSTFIL were expanded to include a table for the SI codewords and a table for the SAO codewords. PSTFIL is now classified at the SI/SAO level.

AFLFIL: The AUTODIN Format Line tables in AFLFIL were expanded to include the data for TCC validation based on the SAO codeword, the minimum security level, and a network identifier. All these additional parameters are optional.

GOTFIL: Though not necessary for the codeword additions, the GOTFIL component of TISFIL was changed to make QUERY/RESPONSE and CHANNEL SEQUENCE NUMBER read-only options so that these options are not available to the SEND application program and thus cannot be modified by the user.

STSFIL: Though not necessary for the codeword additions, the messages in STSFIL were augmented to permit the CSN to be dislayed for AUTODIN messages received and to not be displayed for AUTODIN messages sent.

(3) Application Program Modification: The following modules were changed to support codeword processing:

SBAEDI: The display screens created to display message header information were revised tc include input fields for the SI and SAO codeword parameters. Coding was provided to validate inputs to these fields before the system accepts the inputs and installs them into the message header. The addition of the codeword entries to the message header required the use of a buffer to pass precedence and security information between programs, as the standard 16-word SSB service request block was not large enough to accommodate all the information.

DSDRVR: The module used to output displays to the terminal and to the line printer was modified to include the two additional lines required to output ASCII text for the SI and SAO codewords to the calling program. The driver was also modified to output "UNKNOWN" in codeword fields for those instances where messages received over the AUTODIN network do not have recognizable security parameters values on Format Line 12.

SECMON: The Security Monitor subroutine was modified to validate codeword entries. The SI codeword value is compared with the classification value assigned to the message and must be less than or equal to the message classification in order to pass the check. The SAO codeword is validated in much the same manner and entails an additional check on the compartment handling rating specified for the message. In order to pass this test, a compartment handling code must be entered whenever an SAO codeword has been entered.

(4) Gateway Programs. The following gateway programs were modified as indicated in order to accommodate codeword processing:

TISTGT: A security validation routine was added to the TG.ADD routine in the Terminal Gateway. This new routine checks the security classification parameters in a message and verifies that the intended recepient is cleared to maintain messages of that classification on his queues. This check is performed for messages being constructed by the user, messages entering the SSB system, and messages specified for search and retrieval actions in the History function.

TISGT1: The AUTODIN transmission gateway (TISGT1), was modified to add the codewords and classification handling caveats on the first line of the text in a message on Format Line 12. Additional routines were provided to perform Transmission Control Character validation. These routines search the TCC table in the AUTODIN Format Line File, and subject each SSB message header to the following criteria:

- o If the TCC specifies an SAO codeword, it must match that in the header.
- o If the TCC specifies a maximum classification level, it must not be greater than that in the header.
- o If the TCC specifies a network identifier (either NET.AU or NET.GL), it must match the network identifier in the SRB.

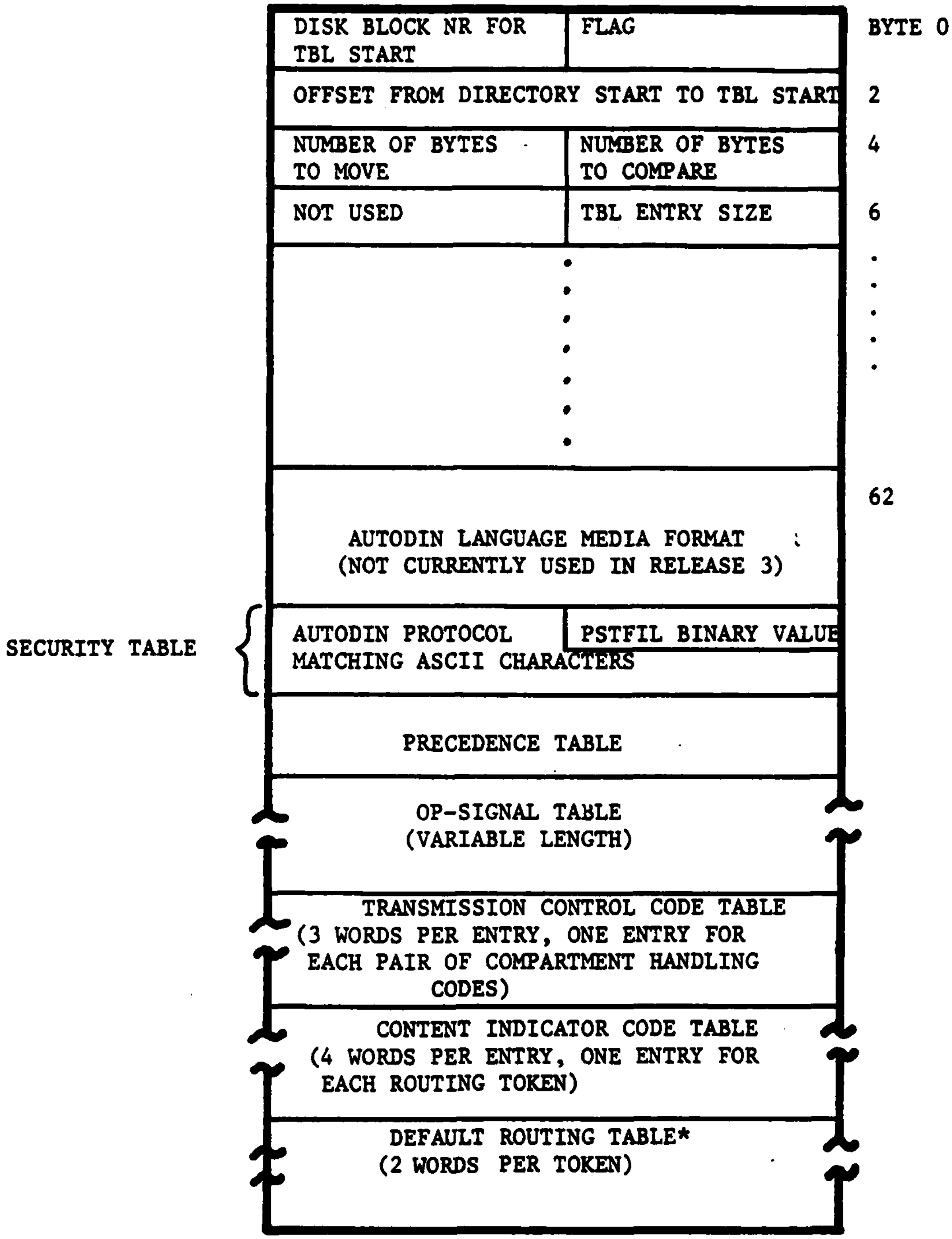
If no TCC is found, or if none satisfy all three criteria, the message will not be transmitted and will be returned to the originator attempting the transmission.

TISGTA: The AUTODIN "Receive" gateway was changed to scan the first two lines of text in a message in order to obtain the handling and codeword parameters for subsequent use by the security monitor subroutines. A code value of negative one (-1) was designated as the default value for any parameters not included but actually required in the message. The field for any mission parameters will be marked as "UNKNOWN" whenever the message is displayed or printed.

2.2.4 Sectioned Message Processing. The capability to process incoming sectioned messages was developed, coded, and unit tested during this contract. The changes to existing files and modules that this capability necessitated are identified and discussed in the following paragraphs.

a. Files. The files affected by the development of sectioned message processing include the AUTODIN Format Line File (AFLFIL), the AUTODN Section Message File (ASMFIL), the Section Ordering File (SOFFIL), the Network Characteristics Table, and the Routing Indicator File (RIFFIL).

(1) AFLFIL. A flag was established in the default routing indicator segment of this file to indicate whether the sectioned message processing program (SBGISM) would be enabled to process incoming sections of a message. The structure of the file is shown in Figure II-1.



* Flag added in this section.

Figure II-1. AFLFIL Structure

(2) ASMFIL. This file was rewritten from a previous sectioned message file (SMFFIL) and made network dependent. The file contains entries for each message section and includes information such as the section number, the Originating Station Routing Indicator, and the date/time group the message was received. Similar section message files will be created for each originating network transmitting sectioned messages to the SSB system. Figure II-2 shows the structure of the ASMFIL.

(3) SOFFIL. All network dependent information was removed from the file and a network identifier entry (NET.ID) was created to identify the originating network. A word for each of seven possible destination gateways and a flag to indicate the processing status for each of the potential gateways were added to the file. Moreover, a byte was added in which to store the total number of destintions. This value is decremented each time that a gateway has completed routing a specific message section. The structure of the SOFFIL is shown in Figure II-3.

(4) Network Characteristics Table. An additional table entry (NET.OM) was added to the table and SBGISM was designated as the program to process incoming sectioned messages. The structure of the table is shown in Figure II-4).

b. System Modules. Several system and gateway programs were affected by the addition of sectioned message processing. They include the AUTODIN Receive Gateway (TISGTA), the Message Distribution Module (TISMDM), the Sectioned Message Utility (SECUTY), and SSBCAT.

(1) TISGTA

- modified to create the ASMFIL and the file entries.

- AFLFIL flag will be checked for each destination to determine if the section ordering module will handle the message.

- Modified the MSN to include a destination block for NET.OM and destination blocks for each ultimate destination flagged with a unique flag to tell TISMDM not to these destinations.

- The MSN will be routed through TISMDM to allow for individual message accountability

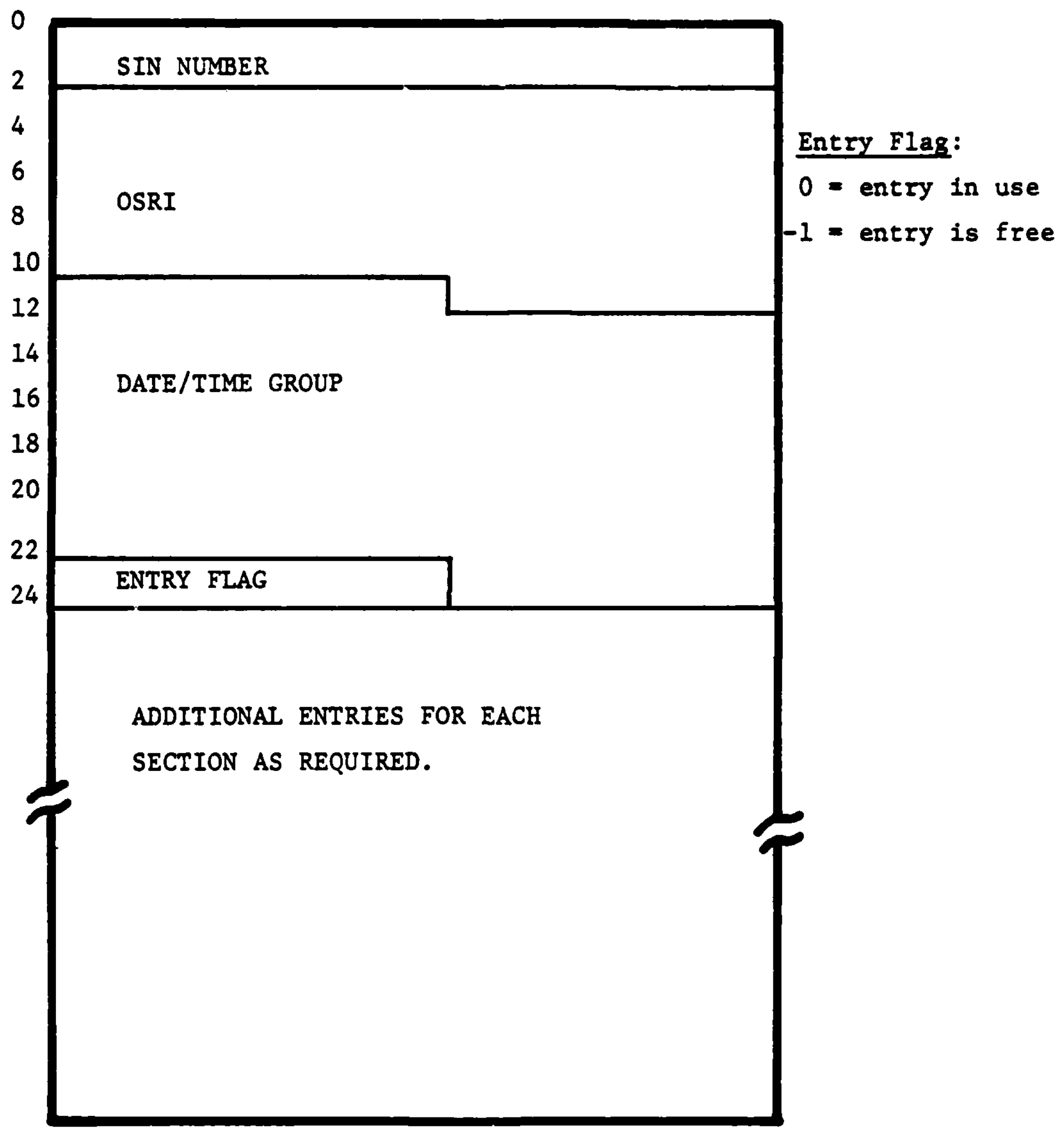


Figure II-2. ASMFIL Entry

Offset		
0		
2	SIN NUMBER	
4	UNUSED	ORIGINATING NET.ID
6	NET ID FLAG 1	DESTINATION NET ID 1
8	NET ID FLAG 2	DESTINATION NET ID 2
10	NET ID FLAG 3	DESTINATION NET ID 3
12	NET ID FLAG 4	DESTINATION NET ID 4
14	NET ID FLAG 5	DESTINATION NET ID 5
16	NET ID FLAG 6	DESTINATION NET ID 6
18	NET ID FLAG 7	DESTINATION NET ID 7
20	DESTINATION GATEWAY FLAG	ENTRY FLAG
22	CURRENT SEGMENT COUNT	TOTAL SEGMENT COUNT
24 26 28	TOTAL FIRST SEGMENT ARRIVED	
30	TIME MOST RECENT SEGMENT ARRIVED	
32	SEGMENT 1 MSN	
34	SEGMENT 2 MSN	
36	SEGMENT 3 MSN	
38	SEGMENT 4 MSN	
	↓ 254 -30 224/2 = 112 segments	
254	SEGMENT 112 MSN	

Entry Flag:

Ø = entry in use

-1 = entry free

Net ID Flags:

Ø = still processing MSNs
1 = SIN processing complete awaiting MDM's answer
2 = Sent SRB to NET gateway
4 = Successfully processed by gateway
10 = error by gateway.

Destination Gateways Flag:

When SOFFIL entry created:

- will be set to the number of destination networks.

When gateway processing completed:

- will be decremented by 1.

When = Ø:

- entry flag will be set to -1. SIN entry completed.

Figure II-3. SOFFIL Entry - Section Ordering File

<table>
<tr><td colspan="2">DEVICE NAME (NC.DEV)</td></tr>
<tr><td colspan="2">UNIT NUMBER (NC.UNT)</td></tr>
<tr><td>FLAG BYTE (NC.FLG)</td><td>NETWORK ID (NC.NNU)</td></tr>
<tr><td colspan="2">INPUT TASK (NC.TKI)</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td colspan="2">OUTPUT TASK (NC.TKO)</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td colspan="2">BUFFER SIZE (NC.SZE)</td></tr>
<tr><td>NUMBER OF READS
(NC.RED)</td><td>TERMINATE 2
CHARACTER (NC.TER)</td></tr>
<tr><td colspan="2">NET.OM</td></tr>
</table>

Figure II-4. Network Characteristics Table

(2) TISMDM

- Checks for the unique flag in a destination block (as assigned above in TISGTA) in order to ignore those destinations.

- Expects an SRB from SBGSOM containing the Network ID (a destination NET.ID from the SOFFIL entry). TISMDM will match the NET ID to a Network Characteristics Table entry and pass back to SBGSOM (in the same SRB as received) the module name of the gateway NET ID.

(3) SBGOSM

- Modified the creation of the SOFFIL directory entry to include destination Network IDs obtained by reading uniquely flagged destination blocks of the MSN and network flags as described above in SOFFIL.

- When all sections of a message are received SBGSOM will send SRBs to TISMDM requesting the module name for each destination gateway in SOFFIL entry. When TISMDM responds with the module name, SBGSOM sends an SRB containing SIN information to each gateway and marks the SOFFIL network flag accordingly.

- SBGSOM will expect an SRB in return from each destination gateway after it has completed processing of the SIN directory entry. SBGCON will mark the network flag in SOFFIL accordingly and decrement the count of total networks. When this count equals zero, ASMFIL and SOFFIL entries will be marked as free.

(4) SECUTY - Section Message Processing Utility

- If a list of all current SIN entries are requested, SECUTY must also ask the user to select which originating network he is interested in. SECUTY will then read the file associated with that network (e.g., ASMFIL and AUTODIN) for the list of SINs and also read the SOFFIL entry for each SIN. The output will include a list of destination gateways for each SIN entry along with curcurrently displayed information

- If information is requested for a given SIN entry, output will include a list of Destination gateways along with the status of the gateway's processing. The originating network for that SIN entry will also be displayed

2.3 Contract Deliverables and Related Documents. Two groups of documents were developed and produced for the Air Force during the course of the contract. The first group consists of documents specified in the CDRL. Most of the documents in this group comprise the AFIS/IND four-volume technical documentation series. Exceptions include the Test and Implementation Plan, the Technical Reports, and the monthly status and funding reports. The second group of documents is comprised of manuals, reports, and ad hoc documents produced to catalog a particular event or to meet operational requirements.

2.3.1 Contract Deliverable Documents.

CDRL No.	Title and Description
A001	Research and Development Status Report The initial report in this series was produced in May 1978 and covered the period from 1 March to 25 May 1978. Subsequent reports were prepared on a monthly basis. The last report in this series covered the period from 26 February to 29 February 1980.
A002	Test and Implementation Plan An interim plan was published in September 1978; the final in February 1980. This document describes the tests and procedures designed to demonstrate the capability of the SSB system to satisfy performance requirements. The tests measure system performance from log on time through message creation and subsequent transmission to external destinations.
A003	Standard Software Base Overview Manual This is Volume I in the AFIS four-volume documentation series. The document contains a general description of the Air Force Common User's Baseline for the Intelligence

Community (CUBIC) of which SSB is a major part. It is written for managers and systems oriented personnel who require knowledge of program philosophy and background, as well as information concerning related program modules

A004 SSB Program/System Subsystem Specifications, Maintenance and Installation Manual

This is actually a series of documents which collectively form Volume II in the AFIS four-volume documentation series. Separate volumes are dedicated to three major program groups; applications, center system, and gateways. A fourth volume contains maintenance and installation procedures. The Installation manual provides the information required to prepare site unique files and tables and how to integrate them into the SSB system.

A005 User, Operator, and Programmer's Manual

These three documents comprise Volume IV in the AFIS four-volume documentation series.

The User Manual is written for the intelligence analyst who will use the system as a functional tool. The manual contains an introduction to SSB philosophy and operating procedures and is illustrated with sample CRT screen printouts showing input formats, system prompts, and actual outputs of function operation.
The Operator's manual provides the information required by computer operators to startup, operate, terminate, and restart the SSB system. The manual includes listings of the batch jobs used to generate the system, and a list of error and advisory messages output to the system console while SSB is in operation.

The Programmer's manual is written for the experienced Macro-11 programmer who wishes to develop additional applications programs for the SSB system. The manual is comprised of three parts. Part 1 discusses applications programming and terminal interface procedures and techniques. Part 2 describes the intertask control module and the techniques used to pass information between tasks, and also provides examples of system usage. Part 3 contains machine listings of programs used as references of programming techniques.

A006 Test Analysis Report

This report is Volume III of the AFIS documentation series. It contains the results of tests performed in accordance with the conditions, controls, and procedures specified in the Test and Implementation Plan (A002) published under separate cover.

A007 Technical Report

An interim report was published in March 1979, covering the technical effort expended during the first year of the contract.

This, the final report documents the major technical tasks accomplished during the entire span of the contract.

2.3.2 Related Documents. The documents described in this section of the report were produced in addition to line items identified in the CDRL.

a. A Gateway Construction Manual was prepared and delivered to AFIS/IND in January 1979. The manual discusses the software and data structures with which a gateway must interact in order to communicate with SSB. The manual also identifies the macro calls applicable to gateway/SSB interface, and discusses the hardware factors that the designer of a gateway must consider.

b. A TTDL/III programmer's manual was developed and published in two volumes. Volume I was published in April 1979. It provides the information required by a programmer to use TTDL/III programming procedures and techniques. The compilation of displays, system components, and TTDL operational procedures are explained in detail. Volume II, published in July 1979, describes the major modules comprising the TTDL/III system.

c. A functional description of the Interprocessor Gateway (IPG) was developed by INCO's Advanced Systems Group and delivered to AFIS/IND in September 1978. Updates to this document were delivered to AFIS/IND on 14 December 1979.

d. An integration test plan was developed to validate the integration of the SSB/CSP installation for the Military Airlift Command at Scott AFB, Illinois. The document describes the tests and procedures required to demonstrate the capability of SSB, installed in one computer, to communicate with the Communications Support Processor installed in another computer, which in turn accesses the AUTODIN network via an Analytic's Telecommunications Line Controller.

e. A configuration plan was developed during the last year of the contract and published in February 1980. This document identifies and establishes the policy and procedures for the administration and control of SSB/TTDL software development and documentation activities.

SECTION 3

3. INSTALLATIONS AND SITE SUPPORT

3.1 <u>Installations</u>. SSB Release III was installed at operational sites in the United States and in USAFE. Installations conducted during 1973 were configured with TTDL/I as the terminal interface system. Installations performed in 1979 used TTDL/III as the interface between the user terminals and the rest of the SSB system. The first operational sites to be equipped with TTDL/III were Ramstein AB in Europe, and the Military Airlift Command at Scott AFB Illinois. SSB Release III installations are detailed in the following paragraphs.

3.1.1 <u>AFIS/IND</u>. The SSB Release III systems at AFIS/IND and USAFE were updated during this reporting period. The AFIS/IND site serves as an operational testbed for all development work. Enhancements are tested first at the INCO facility, and subsequently on the AFIS/IND system where function performance can be observed and measured in an operational environment. The USAFE installation is an operational site; only tested and approved software is installed on the system.

TTDL/III and the revised Applications Group programs were installed on the AFIS/IND 21(V) system during January through March 1979. Detailed testing of these modified programs were conducted concurrently at the INCO development site and at AFIS/IND in preparation for an update to the USAFE Release III system, and for an installation planned at MAC. Changes to Release III include improved and enhanced AUTODIN routing capabilities, precompiled application program displays, and the merging of logically related programs.

3.1.2 <u>Schierstein</u>

a. A specially modified SSB Release III system was installed in Schierstein, Germany. This modified SSB system contained none of the SSB applications programs, and consequently presented unique problems for configuration management and site support. During the course of the year, many site-specific modules had to be developed to support functions normally accomplished with the complete SSB release III system. The overall CATIS system at Schierstein passed a Defense Communications Agency Category III test on 16 March 1978. The CATIS system installed there used SSB Release III as the communications interface to AUTODIN. This system used TTDl/II as the terminal interface to applications programs written by Bunker Ramo. In this arrangement none of the SSB Release III applications programs were present; thus when problems that required message handling arose, they could not be corrected by using the SSB modules developed for precisely that function. In addition, the unique mission of CATIS required development and maintenance of unique modules not contained in SSB Release III. As the system became operational, additional requirements not addressed prior to installation, were identified.

These requirements include:

- More than one Disk Block for destinations in a TCF-formatted message

- The capability to perform AIG and DAG processing via specific AUTODIN gateway options passed from CATIS

- The capability to direct AUTODIN/GENSER message traffic to tape, with associated modules for non-automated initialized message accountability.

- An AUTODIN/DSSCS-to-tape capablity for verification of correct processing and for back-up when the diect AUTODIN line is down

- The capability to enter AUTODIN messages into the system from magnetic tape dumps of messages received at another location

- The capability for fully automated sectioned message processing

- A retransmission capablity including the ability to generate pilot lines as needed on retransmitted messages

- An operator initiated editing function to enter messages, with formats which intially caused rejection, into the CATIS system

- Automated addition of a ZYS operations signal where required

- Special user and terminal security validation functions

- Use of TDF as the terminal support system

b. In October 1979, the site was revisited in conjunction with the installation trip scheduled for Ramstein.

The primary purpose of the trip was to install and checkout a new AUTODIN Transmit Gateway (TISGT1) becasue previous versions of the gateway were not able to transmit any message with 5 or more route indicators due to rejection by the AUTODIN Switch. This failure in the software could not be duplicated at any other site and the situation had to be investigated since otherwise the Schierstein site could not benefit from improvements made to AUTODIN software in the past year. Of particular importance was the capability to remove trailing blanks which blocked communication with some important sites.

A secondary purpose was to install new versions of TISJOR and TISGTA (the AUTODIN Receive Gateway). The latter had several useful improvements, such as corrected processing for piloted messages. The TISJOR module included minor but very important repair, since the previous versions did not release buffers which would collapse the SSB system after a fixed number of transmissions/receptions.

The New TISGT1, TISGTA, and TISJOR brought a much higher degree of reliability to the operational CATIS site. However, the task of installing these modules was considerably more difficult than could have been anticipated. We had planned one day as more than enough, and felt obligated to reschedule our return trip to spend a total of over five days at Schierstein after seeing the initial results.

On the TISJOR front, the console printouts were checked for the previous six weeks and found that virtually all the error messages from SSB modules were attributeable to the sudden unavailability of buffers from BFRTSK. With BFRTSK at 31K it took about 50 transmissions/receptions to collapse SSB, since each incoming or outgoing message caused 512 bytes to disappear from BFRTSK. With the new TISJOR there were no such problems, however, there were several error message (ID=3000) from TISJOR. These were attributed to the omission in COMFIL of a few parameters. After we rebuilt COMFIL we unexpectedly had a series of "OPERATOR INTERVENTION NEEDED," and finally after TISTAP to clear out the TISJOR queue, all was quiet on the TISJOR front.

For TISGT1 there were very few problems: It successfully transmitted a message to 15 addresses. However, it was unable to handle any retransmission. This problem was not a TISGT1 error, since the Network ID for a retransmitted message was not what it should have been. There were some problems in RTX but TISGT1 was simpler to change. A second problem developed due to a procedural problem. At SSB sites it has been useful to have TISGT1 automatically retransmit a message if a "NO RESPONSE" occurs on the first attempt. TISGT1 puts out a demanding message "OPERATOR PLEASE INITIALIZE THE AUTODIN INTERFACE (TLC)." At the CATIS site a re-initialization of the TLC brought on a TLC collapse: all the lights go out and a system reboot is the only known way to recover. This TISGT1 procedure was changed so at the CATIS site all retransmissions would be manual.

The principle difficulty was in TISGTA where an unusual error condition occurred (non-specific data exception ID=63, meaning end-of-text not found). This error might have been caused by unusual characters between lines, for example, LF CR instead of the AUTODIN standard CR CR LF. It was not possible to ascertain the exact cause of the problem, since in an operational site it is not possible to use ODT or take dumps of incoming buffers. The effort to remove the spurious characters was reprogrammed to occur at a lower level, and this caused a large number of secondary bugs. After four days of debugging, with several disturbances in the operational procedures of the CATIS site, the module TISGTA was once again functioning and no sign of the ID=63 error occurred. The condition had not been encountered when receiving messages from SSB sites. The only site which had the problem was the CATIS site. It was unfortunate that the debugging had to be done at a live site using highly classified live data on a processor heavily loaded with live tasks.

To better understand the difficulties, one problem was to remove the operational version of TSIGTA, install a trial version and if any difficulties arose, it was necessary to remove the trial version and reinstall the new version without the luxury of rebooting as is normally done in program development. At one point the TLC became non-functional perhaps because the "remove" and "re-install" disturbed the BM handler and/or the input buffer manager. Eventually through trial and error we found we could eliminate any disturbance to the operational system by placing the TLC in alternate mode to create a window in which to perform the abort, remove, and re-install for the receive gateway. Placing the TLC in self-test necessitated a total reboot of the live system on several occasions, for inexplicable reasons.

3.1.3 AFAITC - Lowry AFB, Colorado. After the installation of SSB Release III in late August 1978, approximately six man-weeks of support was provided to eliminate minor problems encountered with TTDL/I support of the UNIVAC 1652 terinal. These problems had not been observed previously, since August, 1978, SSB installations at TCATA and at AFAITC were the first involving field sites equipped with the UNIVAC 1652 terminal. Several of these problems were traced to 1652 microcode changes made in the previous year.

After the problems with TTDL/I were resolved, AFAITC lost its SSB system in November 1978 because of hardware failures. New release tapes were prepared by INCO personnel and forwarded to AFAITC in December 1978. The new installation was completed onsite by Air Force personnel without contractor assistance.

An additional two man-weeks of support was provided through telephone consultation and background work to assist AFAITC in writing and installing their own application programs to support Air Force SSB course development.

3.1.4 TCATA, Ft. Hood, Texas. A complete SSB Release III system was installed at the TCATA facility in August 1978. Since then, a gateway was developed and implemented to convert data from FCS format on TRW software into TCF format and subsequent transmission over the AUTODIN network. Several trips were performed to support this site, including a trip to the TRW facility in California, and technical support during Category III testing acceptance testing at Ft. Hood.

3.1.5 AFSS, San Antonio, Texas. A complete SSB Release III system was installed at the AFSS site in October 1978. An updated version of Release III was subsequently installed in January 1980.

3.1.6 MAC, Scott AFB, Illinois. SSB Release III was integrated with the Communications Support Processor (CSP) system in March 1979. The installation was accomplished using TTDL/III to support the TTY terminals; the only terminals then available at MAC. In May 1979, the system was updated in order to handle 1652 terminals. The system at MAC passed DCA Category III testing in early July, 1979. A trip was made to the site later during the month to install additional software to accommodate SI/SAO security processing. The site was revisited in October 1979 to correct problems with message journalization.

3.1.7 Ramstein, AFB. A complete SSB Release III system including TTDL/III was installed in the HQ USAFE site at Ramstein AB in October, 1979. Installed as a standard SSB system with TTDL/III the release was modified to meet site specific requirements. The following changes were made:

(1) The length of the classification line in the message header was increased to 45 characters (21 characters is the standard) in order to accommodate additional security parameters in the AUTODIN receive gateway.

(2) The print module, SBSPRT, was modified to print additional lines of asterisks in the security banner whenever an SAO codeword is present.

(3) An additional classification was created to handle SVC or service messages, and an additional classification code and three colateral handling codes were provided to process GENSER messages marked for NATO release.

(4) A Watch Officer bypass was added to the security processing routine in the terminal gateway. This modification permits the Watch Officer to display any message in the system.

SECTION 4

4. CONFIGURATION CONTROL AND MANAGEMENT

4.1 Configuration Management.

a. Background. INCO created the position of SSB Administrator in early 1976 in anticipation of the need for configuration control. An SSB library was established to provide a central repository for software and documentation. Procedures were published to support configuration management of software still under development.

As the SSB library of modules expanded, and the number of site installations increased, the need for a better means to control the various configurations of both basic and modified software was realized.

By mid-1977, the full complement of SSB Release III was installed at AFIS/IND. Selected portions were implemented at two-field sites, and two additional installations were scheduled. An SSB Configuration Manager was assigned to control and coordinate developmental software with the release software installed at the various sites. The Configuration Manager's primary responsibility was the development of the policy and procedures required to implement and conduct configuration management.

b. Design Considerations. Alternative methodology and procedures were appraised in order to develop the most practical configuration management plan for SSB. Careful consideration was given to the preparation and distribution of forms to serve as the basis for the tracking system which would provide control over all SSB modules. For example, three versions of a given module could exist concurrently: The original version supplied with a release to the field; a version updated and maintained for immediate release; and a version containing enhancements still under development. Control of this software requires internal as well as external procedures. Developmental disk packs must be controlled and coordinated with release software and related documentation. Additionally, site-specific tapes must be maintained to reflect the software configuration at the various field sites, and to provide a 'back-up' should the original file be destroyed.

c. Technical Accomplishments. Concurrent with ongoing development and enhancement efforts, configuration management and control procedures and forms were prepared and implemented. Procedures were developed for work assignment, disk-pack cleanup, test performance and evaluation, and other necessary internal functions. Forms and procedures were prepared for field user requests, discrepancy reporting, modification approval, and other external functions addressed during the development of the

Configuration Management Plan. These items were documented and coordinated with AFIS/IND prior to phased implementation.

Currently, SSB Releaase III software is completely operational in accordance with procedures and guidelines set forth in the SSB Configuration Management Plan published in April 1978, and updated in February 1980.

MISSION
of
Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control Communications and Intelligence (C^3I) activities. Technical and engineering support within areas of technical competence is provided to ESD Program Offices (POs) and other ESD elements. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.

ILMED

8