ESD-TR-68-379

ESD RECORD COPY

RETURN TO
SCIENTIFIC & TECHNICAL INFORMATION DIVISION
(ESTI), BUILDING 1211

ESD ACCESSION LIST
ESTI Call No. 63610
Copy No. 1 of 2 cys.

MTR-754

ESLF

DISPLAY SOFTWARE TECHNIQUES FOR AFICCS

Thomas A. Mackey

DECEMBER 1968

Prepared for

DIRECTORATE OF PLANNING AND TECHNOLOGY
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts

This document has been approved for public release and sale; its distribution is unlimited.

Project 512V
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract AF19(628)-5165

AD680384

When U.S. Government drawings, specifications, or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Do not return this copy. Retain or destroy.

ESD-TR-68-379

MTR-754

DISPLAY SOFTWARE TECHNIQUES FOR AFICCS

Thomas A. Mackey

DECEMBER 1968

Prepared for

DIRECTORATE OF PLANNING AND TECHNOLOGY
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts

This document has been approved for public release and sale; its distribution is unlimited.

Project 512V
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract AF19(628)-5165

FOREWORD

Contractor: The MITRE Corporation
Bedford, Massachusetts 01730

Contract Number: AF19(628)-5165

Air Force Contract Monitor: Charles L. Bruce, ESLFA

This report presents the results of a study of the current display software package within AFICCS and recommendations for its improvement.

REVIEW AND APPROVAL

This technical report has been reviewed and is approved.

WILLIAM F. HEISLER, Colonel, USAF
Chief, Command Systems Division
Directorate of Planning and Technology

ABSTRACT

This document presents a synopsis of the current display software package within AFICCS and recommendations for its improvement. The recommendations are based on the results of experiments conducted at the AFICCS Support Facility. The recommended improvements are independent of a particular AFICCS CPU; however, the display techniques are oriented to BR-90's and similarly configured display devices.

ACKNOWLEDGMENT

The author wishes to express his gratitude to O. Beebe, MITRE Corporation Dept. D-73, for his aid in the preparation of this document.

TABLE OF CONTENTS

	Page
SECTION I	
INTRODUCTION	1
SECTION II	
BACKGROUND	2
SECTION III	
AVAILABLE SOFTWARE TOOLS	5
3.1 GENERAL	5
3.2 EVALUATION	6
3.3 SUMMARY	6
SECTION IV	
RECOMMENDED SOFTWARE IMPROVEMENTS	7
4.1 OUTLINE	7
4.2 BUFFER MANAGEMENT	7
4.2.1 Modified IOCS	7
4.2.2 Multi-Strings	8
4.2.3 Projected Slide Data	8
4.2.4 Summary	9
4.3 CONTROL FEATURES	9
4.3.1 Buffer Control	9
4.3.2 Employment of Multi-Consoles	11
4.4 GRAPHICS	12
4.4.1 Dynamic Production	12
4.4.2 Quantified Data	13
4.4.3 Map Data	14
4.4.4 General Purpose Languages	15

TABLE OF CONTENTS (Concluded)

	Page
SECTION V	
SUMMARY	16
APPENDIX A	
GRAPHIC ALLOCATION OF MOBILE UNITS EXPERIMENT	17
APPENDIX B	
THE BROWSE EXPERIMENT	20
APPENDIX C	
RESULTS OF THE TEXT FORMAT ANALYSIS	23
APPENDIX D	
THE N-MODE PROGRAM	29
APPENDIX E	
BR-90 ASSEMBLY PROGRAM — BRASS	38
BIBLIOGRAPHY	39

LIST OF FIGURES

		Page
FIGURE 1 -	CONFIGURATION OF THE AFICCS DISPLAY SYSTEM	4
FIGURE C-1	TEXT FRAME #1	26
FIGURE C-2	TEXT FRAME #2	27
FIGURE C-3	TEXT FRAME #3	28
FIGURE D-1	OCC INTERRUPT PROCESSING	31
FIGURE D-2	EXTERNAL COMPUTER INTERRUPT ROUTINE	32
FIGURE D-3	REFRESH ROUTINE	33
FIGURE D-4	EXECUTIVE (IDLOOP)	34
FIGURE D-5	FIXED FUNCTION KEYBOARD ROUTINE	35
FIGURE D-6	VARIABLE FUNCTION KEYBOARD ROUTINE	36
FIGURE D-7	ALPHANUMERIC KEYBOARD ROUTINE	37

SECTION I

INTRODUCTION

This document presents pertinent software design characteristics for the AFICCS Display System. The characteristics which are not currently available within the system are recommended for implementation. The recommendations are based on experiments performed at the ESD/AFICCS Support Facility. By implementing the recommended software techniques, AFICCS members will have the necessary tools for developing graphic capabilities for their operational applications.

This document is organized into five sections and five appendices. This section is the first section. Section II presents the configuration of the AFICCS Display System and fundamental criteria for operating a display system. Section III presents a brief summary of the currently available software tools. Section IV contains the complementary, recommended software tools necessary for the development of AFICCS operational capabilities. Section V summarizes the previous sections in relation to the distributed data processing technique for graphic devices.

The appendices present the background information which form the basis of the recommendations. The first two appendices present the results of on-line interactive man/machine models. Appendix C presents the results of a feasibility study for the conversion of text data frames into graphic polystrings. Appendix D presents a discussion of the OCC-resident program, N-mode. Appendix E outlines the features of a MITRE-generated 1410 program which permits assembly of OCC programs.

SECTION II

BACKGROUND

The AFICCS Display System currently consists of the following:

i) an IBM 1410 functioning as the central processing unit, CPU, (HQ USAF is replacing the IBM 1410 with an IBM 360/50);

ii) an AN/FYQ-38 functioning as the computer interface buffer, CIB; and

iii) one to six AN/FYQ-45 (BR-90) functioning as operations control consoles, OCC.

The CPU controls the operation of the OCC's and exchanges data with the OCC's via the CIB. The CIB simulates an IBM 729-IV Magnetic Tape Unit (Figure 1).

Both IBM and Bunker-Ramo provided software packages for their respective equipment. Reference 1 describes the interface requirements for operational employment of the Display System.

Communication from the CPU to an OCC is facilitated by the fact that the CPU can interrupt an OCC. This interrupt feature, however, is a one way transaction. An OCC cannot interrupt the CPU when an OCC requires service. This lack of an interrupt from an OCC to the CPU is the main reason why usage of the OCC's is minimal.

When an OCC is operating, the CPU continually queries the CIB to determine if an OCC requires service. This process eliminates the possibility of the CPU performing other functions in a time-sharing manner. However, an application of the distributed data processing technique will provide a vehicle for allowing the CPU to process other tasks while an OCC is operating.

Distributed data processing is a method for utilizing available equipment in a pseudo-independent manner to alleviate the need for constant intercommunication between devices. In the case of a display system this implies the use of storage and control features of the display devices in a manner which minimizes the amount of necessary data exchange between the CPU and the display devices.

In the AFICCS Display System distributed data processing implies the use of the OCC's features to a greater degree than is currently being done. The current software requires referral to the CPU after each operator action. This is certainly not necessary. The succeeding sections of this paper will present methods for exploiting the features of the OCC's without completely revamping the existing software.

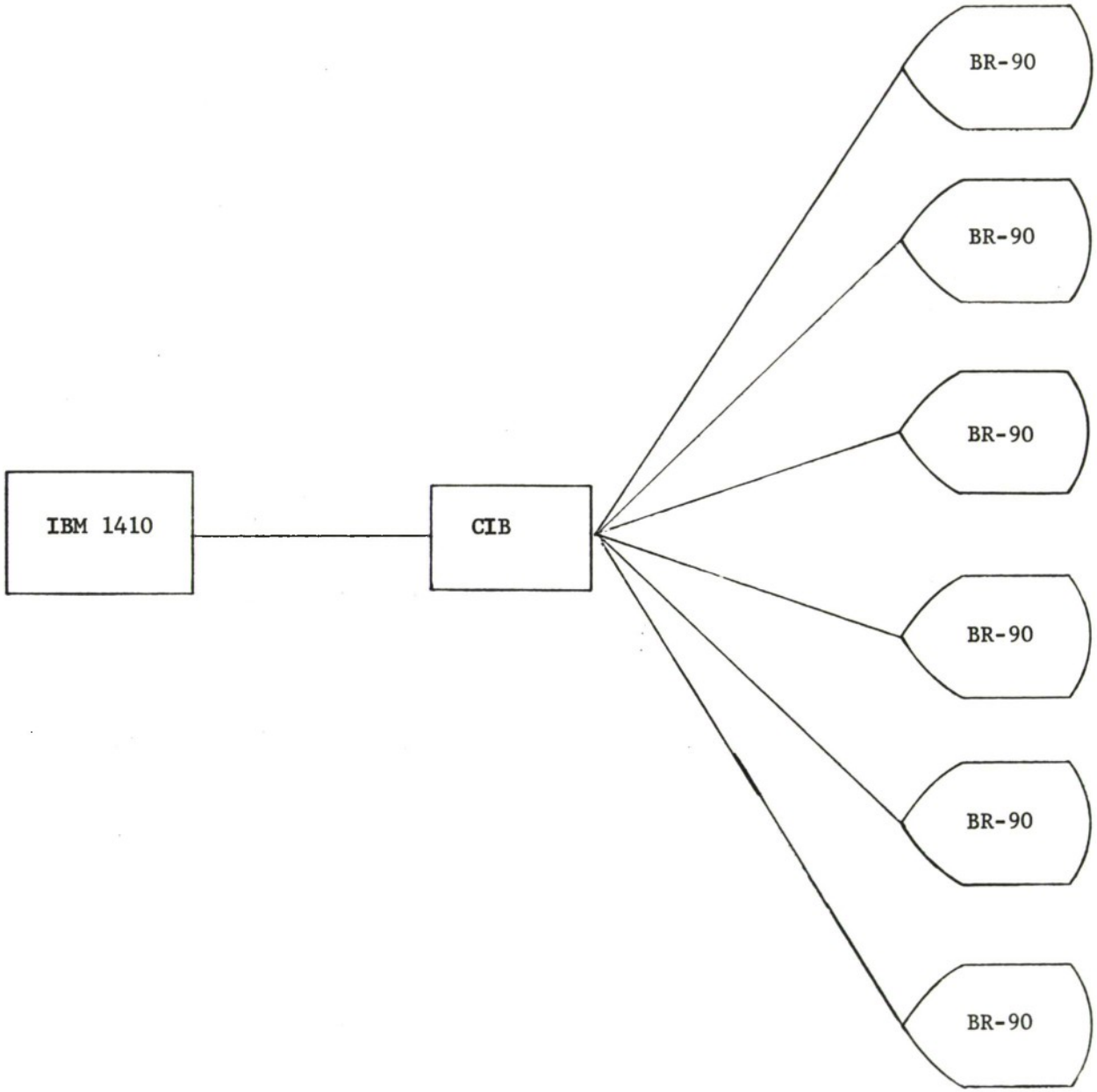

Figure 1. Configuration of the AFICCS Display System

SECTION III

AVAILABLE SOFTWARE TOOLS

3.1 GENERAL

A software package for a display system must address four interdependent areas:

i) Display Control Functions

The Bunker-Ramo supplied N-mode program provides for the execution of these functions. The program is an OCC resident program which permits the use of an OCC either for an off-line operation, or for an on-line operation with the CPU. In an on-line operation this program executes the directions of the CPU (reference Appendix D).

ii) CPU Requirements

IBM has produced a set of 1410 resident programs, called the Console Interface Programs, CIP. These programs handle the buffer management and the operational (rather than program) control of the display devices.

iii) Interface Requirements

The manner by which CIP and N-mode exchange information is dictated by Reference 1. This document states the prerequisites for passing data through the CIB and specifying formats of communication between the CPU and the OCC.

iv) User/Programmer Routines

The structure of the AFICCS organization provides for the interchange of capabilities among the user commands. Current capabilities such as QUEST II Overlay are available; however, utility routines for developing particular graphic capabilities are required.

3.2 EVALUATION

The current available software has been evaluated at the ESD/AFICCS Support Facility. This evaluation emphasized the need for the AFICCS Display System to interact with the operational requirements of AFICCS.

i) Advantages

a) In conjunction, CIP and N-mode reflect most of the operational requirements of interactive display devices;

b) With modification both CIP and N-mode will provide the necessary tools for developing AFICCS graphics capabilities;

c) Experience using the available software has been gained by the user commands; and

d) CIP and N-mode are operational.

ii) Disadvantages

a) Incomplete employment of the OCC capabilities;

b) Continual referral between CPU and OCC;

c) Dedication of the CPU during OCC operation; and

d) Insufficient tools for developing and manipulating graphic data frames within the CPU.

3.3 SUMMARY

The available software within the AFICCS Display System represents a basis upon which a functional display software package can be developed. This package will provide the software tools for employing the OCC's in a more effective manner. A complete redesigning of a software package is not warranted. The improvements that are necessary to the current software are presented in the following section.

SECTION IV

RECOMMENDED SOFTWARE IMPROVEMENTS

4.1 OUTLINE

These recommendations complement the current software of the AFICCS Display System. They are supplementary characteristics of a display software package which are required for gaining higher utilization of the AFICCS Display System than is currently being attained. Alterations to the current software package are minimal. The recommendations are divided into the following three categories:

i) Buffer Management;

ii) Control Features; and

iii) Graphics.

4.2 BUFFER MANAGEMENT

This section of the recommendations is the most crucial. The fundamental reasoning is based on the distributed data processing principle. The necessity for utilizing the OCC to its fullest extent is required to minimize the distinct number of referrals between the CPU and the OCC. This requirement can be achieved by the three following techniques:

i) Modified Input/Output Control System, IOCS;

ii) Multi-Strings; and

iii) Projected Slide Data.

4.2.1 Modified IOCS

The OCC has many external features which must be programmed for, both in the OCC and the CPU. For a particular capability, only certain OCC features are utilized; hence, there is no reason for coded instructions to be present when the particular capability is being executed. The core storage which contains the unused coded instructions should be utilized for storing information pertinent to the operating capability. This technique would permit more economical usage of the OCC core memory and is required for any distributed data processing operations.

4.2.2 Multi-Strings

Currently, the OCC displays one frame of text data and must return to the CPU for the next frame of text data to be displayed. This method requires one string of data where each blank in the text message requires a cell of the OCC memory. This procedure implies that each text message requires 2820 cells. The futility of this manner of processing text displays is presented in Appendix C.

To permit the displaying of more than one text frame before returning to the CPU it is recommended that the textual mono-string be converted to graphic poly-strings. The processing of these graphic poly-strings can encompass man/machine interaction by programming the OCC to handle light-gunned elements in a similar manner to the way it does now with the N-mode program.

4.2.3 Projected Slide Data

The OCC has slide projection features which can be used as a storage vehicle for non-dynamic data. By associating a programmed raster* for a particular slide, a data element on a projected slide can be machine-recognized with a light gun operation. The data element would be uniquely identified in the following manner:

'MSSXY', such that

i) M designates the magazine number;

ii) SS designate the slide-number; and

iii) XY designate the location of the data element via the programmed raster.

---

*A programmed raster is an orderly placement of CRT data (usually dots or periods) on the CRT screen. A raster permits the use of the light gun feature of the OCC.

The combination of a number of slides for a unique programmed raster is necessary to eliminate having a programmed raster for each slide. This can be accomplished by aligning the slide data elements prior to the photography process.

The combining of encoded slide data with alphameric keyboard entries would be ordered by the capability being executed. This would permit the necessary inclusion of dynamic data with the static data of the projected slide.

4.2.4 Summary

Each of the above techniques are presented for the sole purpose of enlarging the current software buffer in the OCC. These techniques are valid for any display device since the primary aim of the recommendations is to minimize the number of unique data transferals between the CPU and the display devices.

4.3 CONTROL FEATURES

This section of the recommendations deals with the control of the OCC by the CPU. These features are CPU-resident and reflect the requirements levied by using the recommended buffer management techniques. In addition, a discussion for the use of a multi-console environment is presented.

4.3.1 Buffer Control

The capability to handle the modified OCC software package demands the formatting and decoding of data prior to, and subsequent to, the data transfer between the CPU and the OCC.

In the case of converting the textual mono-string to graphic poly-strings the main objective is the viewing of the data; however, man/machine interaction is enhanced by this technique. The CPU must execute the following steps:

i) determine the available storage in the OCC by use of the modified IOCS;

ii) reformat each page (frame) of textual mono-strings into graphic poly-strings;

iii) maintain a frame directory for each set of frames that is developed;

iv) transfer to the OCC the maximum number of frames that can be handled; and

v) respond to operator action for the remaining frames or execute utilities such as the printing of displayed frames when the viewing is complete.

For the employment of slide data the CPU must perform the decoding process which correlates the 'MSSXY' identification with the exact data element. This process would be 'front-end' prior to the execution of a particular capability by the CPU.

The entry of alphameric data at the OCC would require no encoding and would be interwoven with the encoded slide data in the order designated by the capability being processed. Hence, the CPU would decode the slide data and insert the alphameric keyboard data in the same order that the operator specified at the OCC.

Paramount to the effective use of these techniques is the development of a system interrupt feature. It is recommended that one of the CPU-resident routines be examined for the feasibility of instituting a process of checking the interrupt status of the OCC's at the CIB. Consider the AFICCS system routine DSKACC. DSKACC is exited when data has been successfully transferred from the disk to the core memory. DSKACC then returns control to the main program. It is recommended that the following modification be instituted:

i) upon completion of the DSKACC routine, control enters a CIB query routine;

ii) the CIB query routine checks the service requests of the OCC's at the CIB;

a) if no OCC requires service, the CIB query routine returns program control to the main line program following the parameters of the DSKACC call; or

b) if an OCC requires service, the CPU transmits the contents of core memory to a disk buffer, and calls the necessary OCC service routines; upon completion of the service routine, the main line program will be restored from the disk buffer, and processing will resume.

Implications of this technique for obtaining an interrupt feature are:

i) a requesting OCC could interrupt any program;

ii) only programs which use DSKACC could be interrupted; and

iii) CIB query and core swapping routines are required.

Means of handling these implications are:

i) develop a modified priority structure which dictates whether or not a main line program can be interrupted (could be as simple as an 'on-off' condition);

ii) utilize the most called system routine (DSKACC is only an example); and

iii) develop coding and provide core memory for these routines.

Without some form of interrupt the current procedure will remain in effect. However, the recommended uses of the OCC features are not dependent on a CPU interrupt. The distributed data processing principle will permit a more economical use of the AFICCS Display System.

4.3.2 Employment of Multi-Consoles

The combination of CIP and N-mode has the required structure for controlling more than one OCC. Hence, the employment of multi-consoles has been addressed. Specific uses fall into the following categories:

i) independent OCC operation such that each OCC is executing a different capability; and

ii) intercommunication of OCC's such that they are executing the same capability.

In the first category, monitoring of the OCC's could be accomplished. In the second category, the viewing of the same data on each OCC can be enhanced by permitting a light gun operation

for selecting a segment of the displayed picture. As CIP is designed, it can detect the light gunned element and transfer this fact to the remaining OCC's. This is an operation that CIP and N-mode can execute.

4.4 GRAPHICS

These recommendations deal primarily with the painting of a display frame to include vectors, circles, and alphamerics. These painting procedures can be subdivided into the following groups:

i) Dynamic Production;

ii) Quantified Data;

iii) Map Data; and

iv) General Purpose Languages.

4.4.1 Dynamic Production

Within the available software package there are no CPU-oriented tools for altering a graphic frame except by meticulous and archaic methods. There exists a MITRE-produced BR-90 Assembly Program, BRASS, which aids the preparation of BR-90 programs (reference Appendix E). In addition, a set of tools for altering a graphic frame prior to its transfer to an OCC is required.

One solution to this situation is a set of routines which would permit the following functions:

i) definition of a graphic frame in symbolic notation;

ii) manipulation of the elements of the graphic frame within the CPU; and

iii) translation of graphic elements into the necessary format for transferal to the OCC.

Each of these functions has been executed for particular programs (reference Appendix A and B). As a byproduct of BRASS, the definition of graphic frames in symbolic notation can be achieved; however, this involves manipulation of the BRASS output. A routine which permits direct symbolic usage should be available.

The requirements for manipulating and translating the graphic elements into the necessary format for transmission to the OCC have been described in a MITRE Corporation preliminary report. If the AFICCS Display System is ever to be operationally efficient this aspect of a graphics package must be addressed.

4.4.2 Quantified Data

Routines which permit the graphic representation of numeric data must be available. This type of display frame can be used to exemplify data that can be quantified; i.e., expressed numerically. For effective evaluation quantified data must be examined in at least the following two ways:

i) the data elements compared against one another, such as the Bar-graph capability;

ii) the data elements compared as components related to the whole, such as is used in budget reports and studies; i.e., slicing the tax dollar.

For comparing elements against elements, the usual Euclidean plane with Cartesian coordinates suffices. The use of the vector and the character generators of the OCC permits superimposing of the numeric measures. Four methods for displaying these measures are:

i) Solid line ———————— (one vector);

ii) Dotted line..... (many dots);

iii) Dashed line - - - - - - - - - (many short vectors); and

iv) Dash/Dot line-.-.-.-.-.-.-.-.-.-.-.-.-.-. (dots and short vectors interwoven).

For comparing elements related to the whole, the circle and vector generators of the OCC can be used. This method requires the following steps:

i) determine numeric value of the whole;

ii) determine fraction of each part to the whole;

iii) determine the portion of the circle each part represents (measure in radians); and

iv) use polar coordinates to ascertain the x-y coordinates of each vector used in partitioning the interior of the circle into the required slices.

4.4.3 Map Data

The first magazine of the OCC slide projection system has been provided for the Standard File of Geographic Reference Slides. The slides are grouped in the following manner:

i) Mercator projections

a) countries, states and geopolitical entities of the free world (slides 1 to 62);

b) continents and inter-continents (slides 63 to 76);

ii) polar stereographic projections (slides 87 to 94); and

iii) slide index (slides 97 to 99).

A means for the superimposition of CRT data with background slide data had to be developed for the Allocation of Mobile Units Experiment (reference Appendix A). For the use of map data with the CRT data the following elements are necessary:

i) one CRT coordinate related to the appropriate lat/long coordinate (to avoid distortion, the CRT coordinate should be the center of the screen);

ii) the difference in the horizontal or x - coordinate, Δ x, for the CRT in relation to the difference in the latitude for the map;

iii) the difference in the vertical or y-coordinate, Δ y, for the CRT in relation to the difference in the longitude for the map; and

iv) a) for the maps portraying Mercator projections one can use a linear relation to superimpose the CRT data on the map data;

b) for the maps portraying polar stereographic projections one must use non-linear relations; i.e., spherical trignometric functions.

4.4.4 General Purpose Languages

As a long range consideration any display software package must have a compiler-type general purpose language. This is necessary for any sophisticated interactive use of computers with graphic terminals. There have been special purpose languages developed for graphic devices; however, command and control requires a general purpose language which can be readily used by both programmer and user. Any software package would be incomplete without some attempt to ease the programming required by display devices.

SECTION V

SUMMARY

This paper has presented a synopsis of the available display software within AFICCS and has recommended action that will complement the current software capabilities. The recommendations center on programming tools that are necessary for the development of graphic capabilities. The overriding factor throughout this paper has been the distributed data processing principle. Any synergistic coupling of machines requires first, the use of each machine to its potential, and second, the effective transferal of pertinent information between the devices. The recommendations are valid for any CPU which controls a display device, similarly configured as an OCC.

The main features of these recommendations are:

i) implementation can be accomplished with available personnel at the user commands;

ii) operation of the OCC's must be based on the distributive data processing principle; and

iii) independent of the current configuration of the AFICCS Display System.

APPENDIX A

GRAPHIC ALLOCATION OF MOBILE UNITS EXPERIMENT

1. BACKGROUND

This experiment was conducted for the following purposes:

i) evaluation of the 1410-resident CIP set of control programs and the OCC-resident N-mode program;

ii) investigation of the requirements for superimposition of CRT data with background slide data.

2. RATIONALE

The basis for this experiment rested on the following functions:

i) use of an asterisk (*) to designate a location on the CRT screen which related to a location on a projected slide;

ii) selection and projection of the particular slide on which the asterisk was to be superimposed;

iii) positioning of mobile units for particular slides;

iv) use of the following OCC features for directing the position of the asterisk;

   a) alphameric keyboard data;

   b) the light gun with a programmed raster; and

   c) the cursor;

v) use of circles to represent map-oriented distances about the asterisk;

vi) use of non-dynamic data frames to present more detailed information about a particular mobile unit; and

vii) simulation of sentinels to update the locations of the mobile units on prearranged routes.

3. METHOD

The following general techniques were employed:

i) for maps without a generally used coordinate system (e.g., city maps):

   a) map locations were recorded, identified, and stored on the disk with a special purpose program (e.g., for a city map the locations were street intersections); and

   b) location of the asterisk and the mobile units were confined to the pre-stored locations;

ii) for maps with a generally used coordinate system (e.g., military maps):

   a) map locations were related to CRT screen coordinates on the following basis:

      1) the center map coordinates were related to the center CRT coordinates (1000_8, 1000_8);

      2) the difference in the horizontal and vertical directions (Δ x and Δ y) were related to differences on the CRT screen;

      3) a linear algorithm of repeated additions (subtractions) yielded the required CRT screen coordinates for proper placement of the asterisk; and

      4) the use of this technique is limited to Mercator projected maps;

   b) location of the asterisk and the mobile units could be anywhere on the map.

iii) the program developed for this experiment involved no changes to the N-mode program and minimal changes to the CIP package.

4. RESULTS

This experiment provided a vehicle for gaining experience with the components of the AFICCS Display System software package. In particular, the use of various OCC features and the necessary

communication formats were examined in detail. Particularities and inconsistencies of the available software tools were documented in MITRE WP-2150. Some of the observations of this experiment were:

i) individual subroutines of CIP were adaptable to the specific usage of the programmer;

ii) graphic manipulation was cumbersome and detracted from maximum use of the OCC;

iii) housekeeping routines for deciphering the CPU/OCC communications were minimal; e.g., the address of the light-gunned element had to be converted from a 2-character number to a related address for the display buffer in the CPU;

iv) N-mode operated in accordance with TM #1; and

v) CIP handling of CIB communications required alteration.

5. CONCLUSIONS

The following conclusions were derived from this experiment:

i) CIP and N-mode provide the foundation for a display software package;

ii) additional (complementary) graphic utility routines are required for a display software package; and

iii) an approach to distributive data processing is necessary for obtaining operational graphics capability with the OCC.

APPENDIX B

THE BROWSE EXPERIMENT

1. BACKGROUND

This experiment is being conducted for the following purposes:

i) determination of the adaptability of the N-mode program to permit special user-oriented processing; and

ii) demonstration of the feasibility of distributed data processing with AFICCS.

2. RATIONALE

The initial phase of this experiment consisted of the design and implementation of an OCC Interface Routine (OCCIF) as described in Reference 2. OCCIF is a BRASS-coded* BR-90 program which coexists with a modified version of the N-mode program. OCCIF operates in response to a 1410 resident generalized serial file management capability** which permits file generation, updating, and browsing. File browsing enables the examination of entries of an AFICCS serial file according to user-defined specifications.

3. METHOD

A browsed entry requiring man-machine interaction is sent to the OCC, together with appropriate identification in an unformatted data stream. After initiating the transfer of the data stream to the OCC, the file management program waits in a test loop until the OCC requests an interrupt. OCCIF manipulates the input stream, formats the message, and displays it on the CRT. OCCIF accepts operator responses and then signals the CPU for transfer of the updated file entry.

Essentially, the central function accomplished by the BR-90 program OCCIF is the formatting of a raw message stream. In particular, the following changes and modifications were made to the standard N-mode program:

---

* Reference 3
** Reference 2

i) the cursor movement section of the refresh routine was deleted in order to free additional BR-90 core for program additions;

ii) the 'Write Display' routine was modified to permit variable length input streams;

iii) a new routine (FORMAT) was written which formats and displays the received error stream. This routine returns control to the N-mode executive; and

iv) a new routine (FETCH) was written to compress the operator response (typed on A/N keyboard) into a stream format and to signal an interrupt request to the CIB.

4. RESULTS

This initial phase successfully demonstrated that it is not difficult to modify the N-mode program and to assign user-specific (or any other) processing to the OCC. It was further demonstrated that it is sometimes advantageous not to use the CIP control program, but to include the necessary 1410/BR-90 interface code in the user 1410 program.

5. CONCLUSIONS

A natural extension of the concepts of this phase leads to the philosophy of distributed data processing. A clear advantage would be gained if the central processor was permitted to resume with its previous task (browsing a data file), instead of waiting in a test loop while the operator at the OCC determines the applicable response. Further phases of the planned experiment are:

i) the vehicle for this stage will be a modified version of the previously mentioned 'BROWSE' program, which will send a collection of data streams during one transmission to the BR-90. Immediately after this transmission, the CPU will continue executing the BROWSE program and store any further OCC data in a buffer. The CPU will periodically test the CIB for a service request. Concurrently, the OCC processor will format and display the received messages and accept the operator responses. After all available messages have been processed, the OCC will send a service request to the CIB. When the request is detected by the CPU an exchange of buffers will take place; the responses from the OCC will be stored on disk to be merged with the data file; and

ii) this phase of the experiment will supplement the previous effort by including the usage of the light-gun and the OCC projection system. Fixed operator responses, such as legal value tables, will be recorded on slides which will be projected onto the CRT screen along with a programmed raster. The operator can use the light-gun to indicate his selection of a desired table entry.

APPENDIX C

RESULTS OF THE TEXT FORMAT ANALYSIS

1. BACKGROUND

This experiment was conducted for the purpose of determining the feasibility of converting a textual mono-string into graphic poly-strings. The motivating aspect is the use of the OCC core memory for the storage of more than one frame of data at any particular time.

2. RATIONALE

Textual data is currently sent to the OCC in a single string of 2825 characters via the CIB. This string has the following configuration and destination (OCC cells):

i) first eight characters are control information for the transfer of data;

ii) the ninth character (control information for the display function) is placed in the six least significant bits (LSB) of each OCC cell which stores the internally encoded display frame; and

iii) the remaining characters (10th, 11th,....) which compose the actual alphameric message, are placed in the six most significant bits (MSB) of each OCC cell which stores the internally encoded display frame.

Once the text string has been transferred to the OCC, the text string is stored in the following manner:

i) the first two cells contain the x-y coordinates of the top, left-hand side of the CRT screen (0010_8, 1750_8) and additional control information to designate an alphameric string;

ii) the next 2816 cells have the following configuration:

   a) 6 LSB contain character control information such as the size, blinking state, etc;

   b) 6 MSB contain the actual alphameric character, encoded in BCD; and

iii) the remaining cells contain display termination indicators.

The fundamental rationale of this experiment is the minimization of the core cells that are used to store the blank alphameric character. The presence of consecutive blanks can be decreased by starting a new alphameric string with the first non-blank character encountered following a certain number of consecutive blank characters. Hence, the process consists of analyzing the single text string and producing multiple graphic strings which display the same formatted message.

The following storage requirement is basic to alphameric strings:

i) let m be the number of characters in a string;

ii) hence, m+2 cells are required to store a string, such that:

a) the first two cells contain x-y coordinates which direct the CRT beam to a spot on the CRT screen;

b) m cells for storing the m alphamerics of the string; and

c) the final cell contains both the final alphameric of the message and an EOS indicator.

3. METHOD

The OCC text frame consists of 44 rows each with 64 possible entries; i.e., a 44 x 64 matrix. This matrix can be uniquely related on a linear basis to the CRT screen which has a 1024 x 1024 raster matrix. The positioning of the text characters on the CRT screen is performed by scanning the text matrix from left to right and then top to bottom; i.e., row-by-row.

The linear relations can be formulated as follows:

i) for the CRT x-coordinate:

$8\ (2s - 1)$ such that s is the number of the entry (column) in a given row;

ii) for the CRT y-coordinate:

$1023 - 23t$ such that t is the row number.

Employing this linear relation between the text frame and the CRT screen, the single text string was transformed into multiple graphic strings. This transformation was accomplished for varying

numbers of consecutive blanks, n, to be encountered prior to starting a new string. In particular, transformations were accomplished for 9 values of n. The results of this experiment are presented in figures C-1, C-2, and C-3.

4. OBSERVATIONS

The following results are the high points of this investigation:

i) the number of consecutive blanks to be encountered prior to starting a new string is <u>three</u> because:

   a) for n = 2 or 3, the minimum number of storage cells were used;

   b) for n = 3, the number of distinct strings is less, thereby minimizing the size of a frame directory which retains the location of strings in core.

ii) the low amount of required core storage for the display frames both in the 1410 and in the OCC was unanticipated (some saving was expected but the degree of the saving was surprising).

5. CONCLUSION

The conversion of the text mono-string into graphic poly-strings is feasible. In fact, it is necessary for the use of the AFICCS Display System in a distributed data processing manner.

TEXT FRAME #1

A. Text string has 377 non-blank characters

B. Current Requirements

2816 characters of 1410 disk storage
2820 cells of OCC storage

C. Transformed Requirements and Savings

n	1410 characters	1410 savings	% savings	OCC cells	OCC savings	% savings	# of strings
2	850	1966	69.8	425	2395	84.9	49
3	850	1966	69.8	425	2395	84.9	45
4	850	1966	69.8	425	2395	84.9	45
5	850	1966	69.8	425	2395	84.9	45
6	856	1960	69.6	428	2392	84.8	44
7	856	1960	69.6	428	2392	84.8	44
8	856	1960	69.6	428	2392	84.8	44
9	856	1960	69.6	428	2392	84.8	44
10	856	1960	69.6	428	2392	84.8	44

FIGURE C-1

TEXT FRAME #2

A. Text string has 631 non-blank characters

B. Current Requirements

2816 characters of 1410 disk storage
2820 cells of OCC storage

C. Transformed Requirements and Savings

n	1410 characters	1410 savings	% savings	OCC cells	OCC savings	% savings	# of strings
2	1432	1384	49.1	716	2104	74.6	86
3	1432	1384	49.1	716	2104	74.6	81
4	1436	1380	49.0	718	2102	74.5	79
5	1476	1340	47.6	738	2082	73.8	68
6	1566	1250	44.4	783	2037	72.2	54
7	1566	1250	44.4	783	2037	72.2	54
8	1636	1180	41.9	818	2002	70.8	47
9	1672	1144	40.6	836	1984	70.3	44
10	1700	1116	39.6	850	1970	69.8	42

FIGURE C-2

TEXT FRAME #3

A. Text string has 770 non-blank characters

B. Current Requirements

2816 characters of 1410 disk storage
2820 cells of OCC storage

C. Transformed Requirements and Savings

n	1410 characters	1410 savings	% savings	OCC cells	OCC savings	% savings	# of strings
2	1632	1184	42.0	816	2004	71.0	57
3	1632	1184	42.0	816	2004	71.0	54
4	1664	1152	40.9	832	1988	70.5	48
5	1664	1152	40.9	832	1988	70.5	48
6	1664	1152	40.9	832	1988	70.5	48
7	1664	1152	40.9	832	1988	70.5	48
8	1664	1152	40.9	832	1988	70.5	48
9	1664	1152	40.9	832	1988	70.5	48
10	1678	1138	40.4	839	1981	70.2	47

FIGURE C-3

APPENDIX D

THE N-MODE PROGRAM

1. BACKGROUND

The N-mode program was produced by Bunker-Ramo Corporation for the following purposes:

i) facilitate data-communications between the IBM 1410 and the OCC; and

ii) provide the 'normal' operating functions of the OCC features.

This program is core-resident in the OCC, and permits either on-line or off-line operation. During on-line operation the N-mode program operates as a slave program in response to the Console Interface Programs, CIP, of the 1410.

2. PROCEDURE

The N-mode program is structured so that it can handle two fundamental types of interrupts: first, OCC hardware generated; and second, those interrupts generated by an operator action.

i) OCC Hardware Generated

The executive routine which responds to these interrupts is called the OCC Computer Interrupt Processing (reference Figure D-1). The following two sets of routines execute the necessary instructions for these interrupts:

a) External Computer Interrupt Routine (reference Figure D-2); and

b) Refresh Routine (reference Figure D-3).

ii) 'Operator Action' Responses

The executive routine for responding to operator action is called IDLOOP (reference Figure D-4). The OCC Flag Register is employed to determine the type of operator action performed at the OCC. The following three sets of routines handle operator actions:

a) Fixed Function Keyboard Routine (reference Figure D-5);

b) Variable Function Keyboard Routine (reference Figure D-6); and

c) Alphanumeric Keyboard Routine (Reference Figure D-7).

3. SUMMARY

A review of the N-mode program has shown that it is organized and designed in a completely modular fashion. This design facilitates program additions and modifications which permit N-mode to operate in a 1410 interface environment other than CIP. Since each function of the Fixed Function Keyboard has been encoded as a separate module, additional core memory can be freed by deallocating those subroutine modules not required for a specific N-mode configuration. (Over 50% of the original N-mode core requirement is allocated to Fixed Function Keyboard modules). These freed memory cells can be used by the programmer for general storage, either for new special purpose subroutines or for storage buffers, and thus enhance the distributed data processing concept for the AFICCS Display System.

OCC INTERRUPT PROCESSING

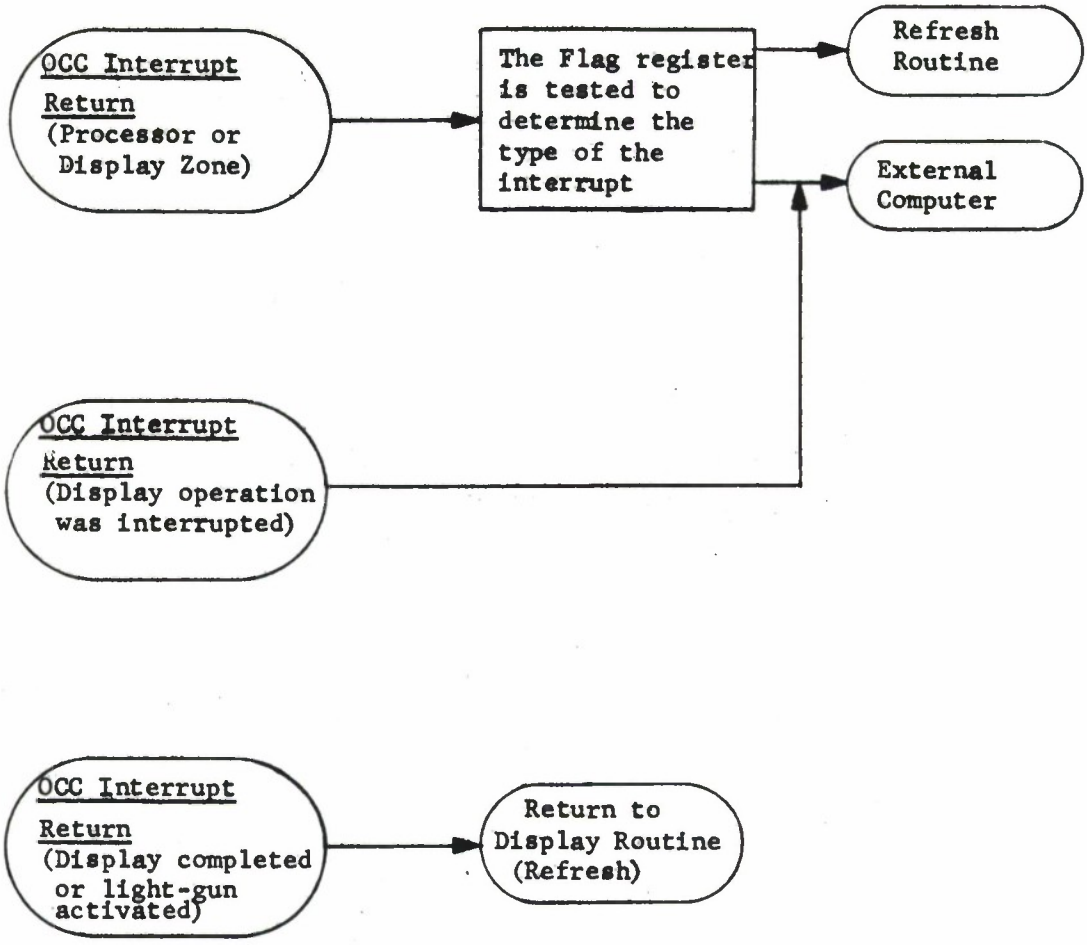

FIGURE D-1

EXTERNAL COMPUTER INTERRUPT ROUTINE

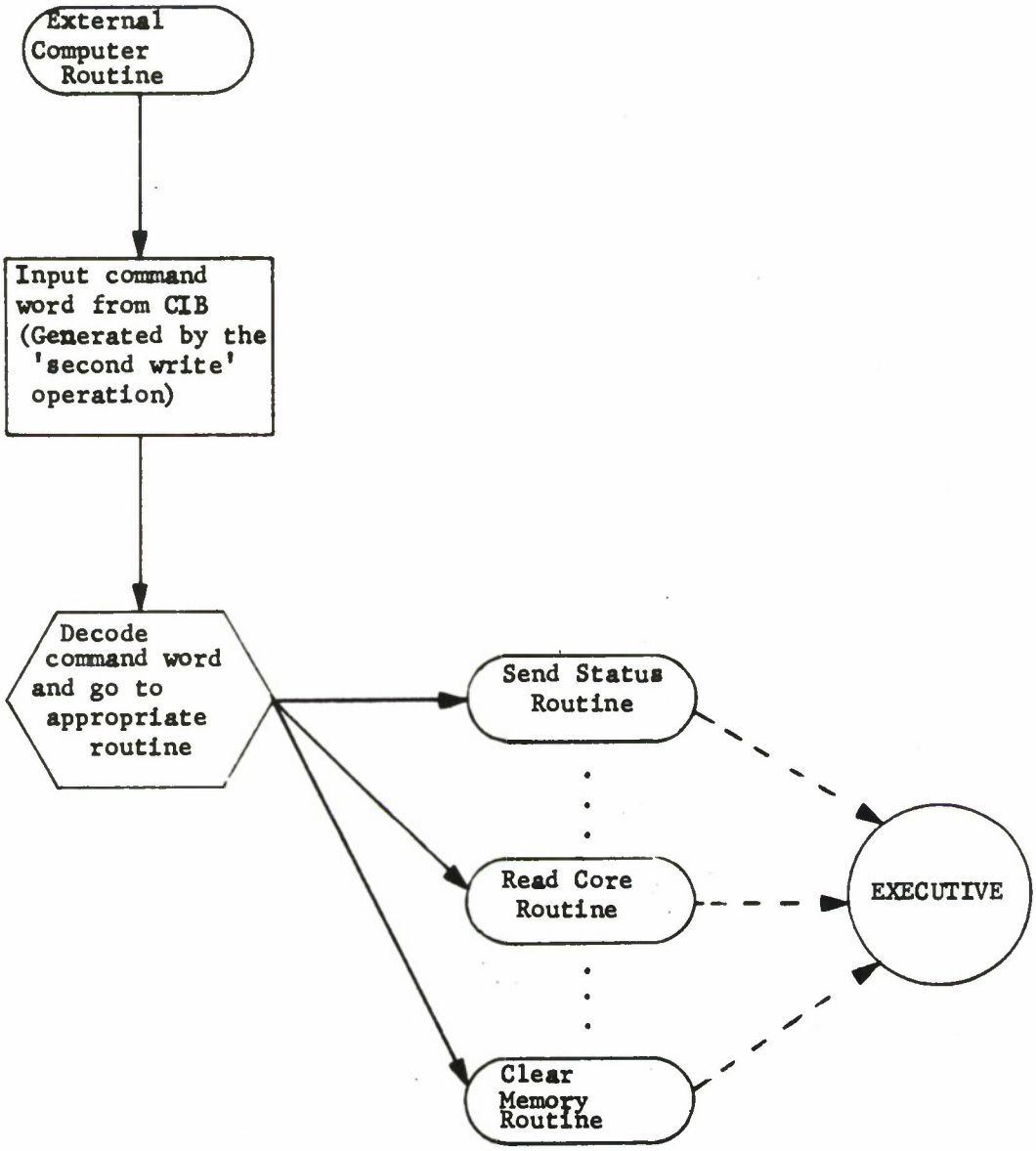

FIGURE D-2

REFRESH ROUTINE

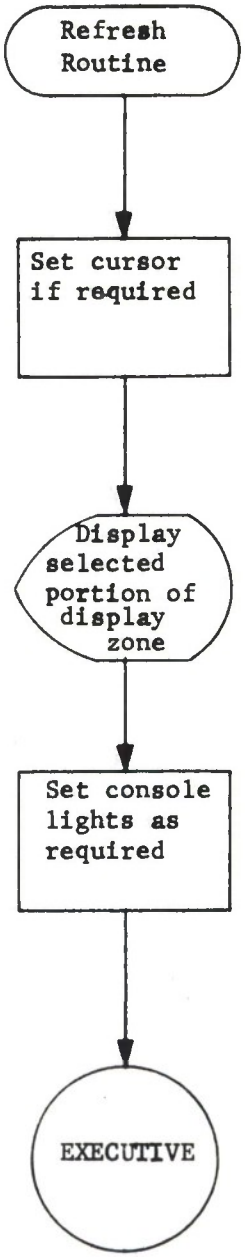

FIGURE D-3

EXECUTIVE (IDLOOP)

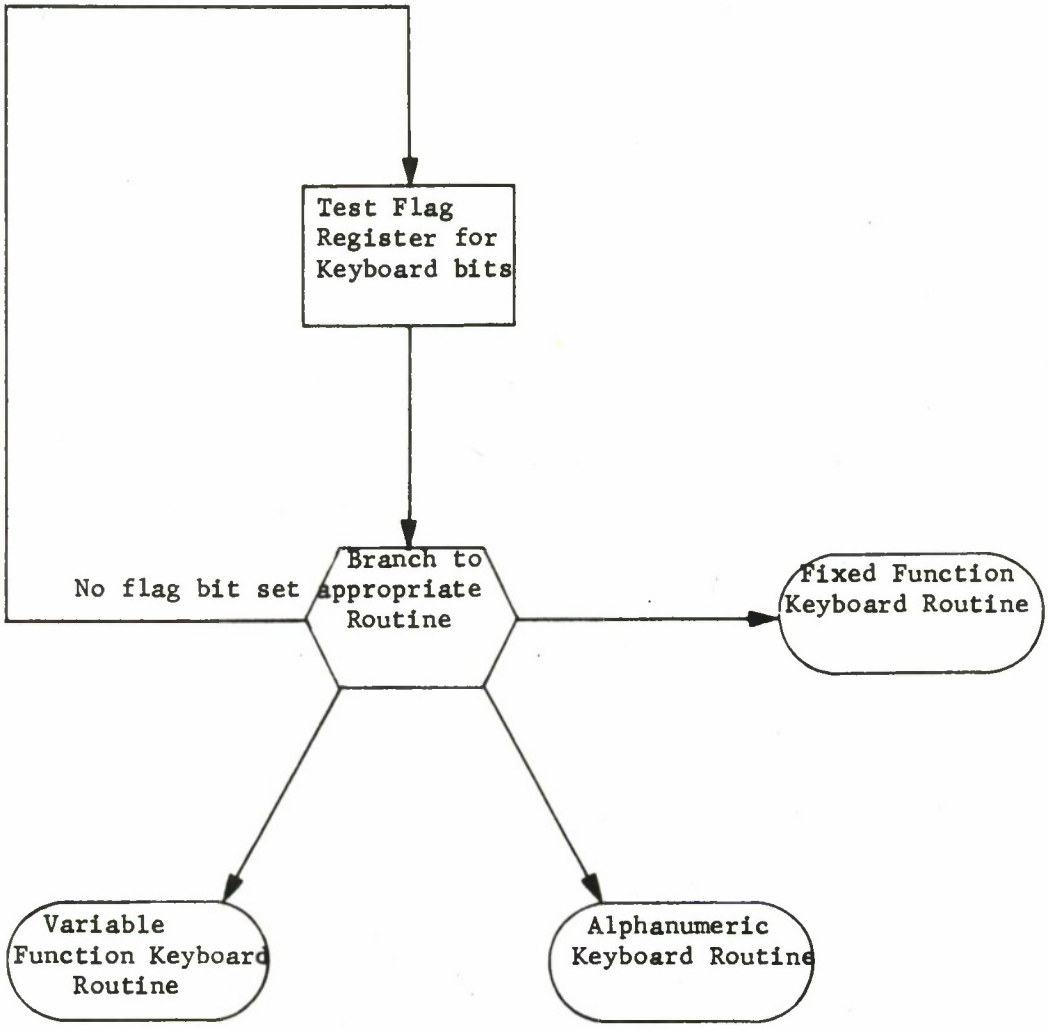

FIGURE D-4

FIXED FUNCTION KEYBOARD ROUTINE

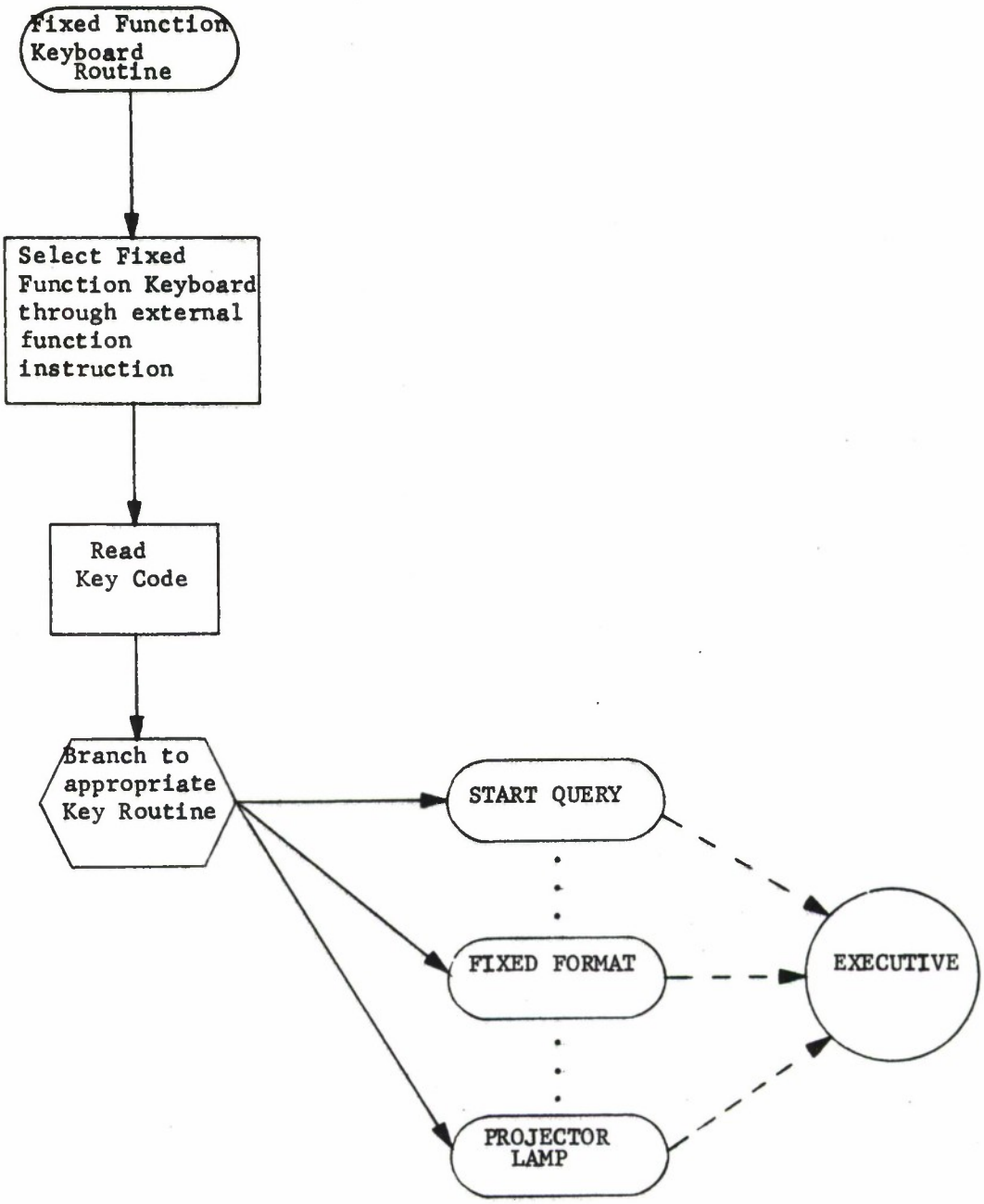

FIGURE D-5

VARIABLE FUNCTION KEYBOARD ROUTINE

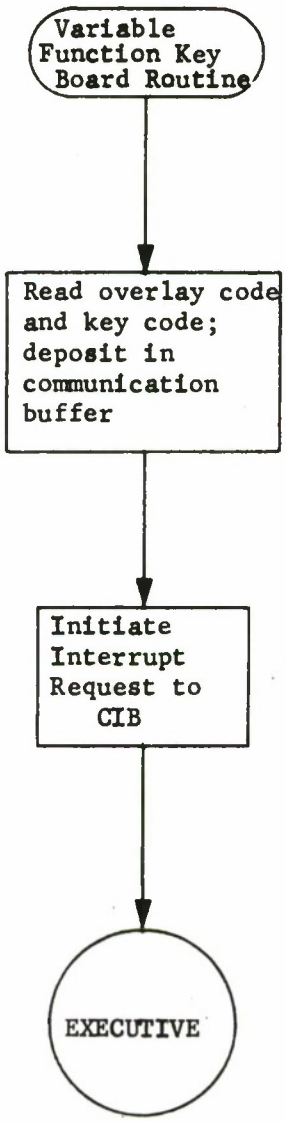

FIGURE D-6

ALPHANUMERIC KEYBOARD ROUTINE

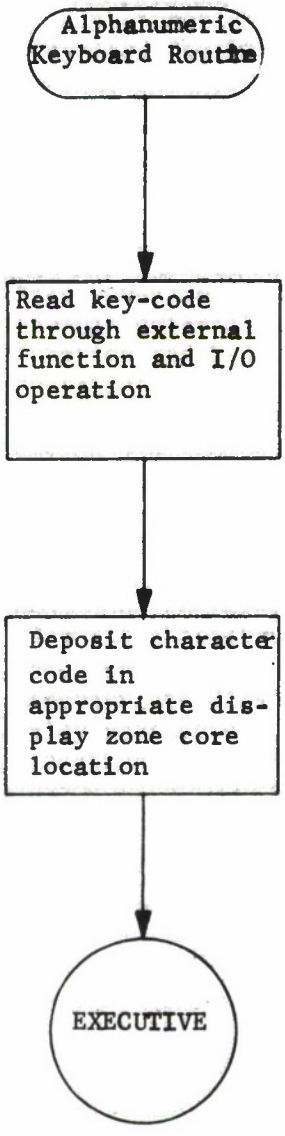

FIGURE D-7

APPENDIX E

BR-90 ASSEMBLY PROGRAM — BRASS

BRASS*, a BR-90 assembly program, was designed for use at the AFICCS Support Facility specifically to provide a vehicle for experimental program development under the Display Console Technology Task. Outstanding features of the assembler are the following:

i) tape (not disk) oriented;

ii) AFICCS independent; and

iii) operational on third-generation computer systems (emulation).

During the BRASS implementation phase, the standard BR-90 Normal Mode Control Program (N-mode) was used as part of the test package and the following highlights were noted:

i) N-mode consists of 4000 source statements with 800 user-labels; and

ii) BRASS assembled N-mode in 12 minutes using 1/5 of the allowable user-label area for N-mode labels.

---

*Reference 3

REFERENCES

1. IBM, Bunker-Ramo, The Operation of the AFICCS Display System in the IBM 1410 Data Processing System, Technical Memorandum #1, December 1966.

2. The MITRE Corporation, Design Specifications for AFICCS Serial File Manipulators, ESD-TR-68-290, Bedford, Mass., May 1968.

3. The MITRE Corporation, BR-90 Assembly Program -- BRASS, ESD-TR-68-132, Bedford, Mass., March 1968.

BIBLIOGRAPHY

Bunker Ramo Corporation, Programming Reference Manual For Message Console AN/FYQ-45, April 1967.

C. Christensen and E.N. Pinson, Multi-Function Graphics For A Large Computer System, Proc. of Nat. Conf. of ACM, 1967, pp. 355-365.

H. Corbin and W. Frank, Display Oriented Computer Usage System, Thompson Book Co., 1966.

J.C. Gray, Compound Data Structure for Computer Aided Design; A Survey, Proc. of Nat. Conf. of ACM, 1967, pp. 355-365.

F. Gruenberger, ed., Computer Graphics, Thompson Book Co., Washington, D.C., 1967.

IBM, Programmer Procedures Manual For Console Interface Programs (CIP), November 1967.

IBM, Computer Program Specifications For Console Interface Programs (CIP), November 1967.

E.L. Jacks, A Laboratory For the Study of Graphical Man-Machine Communication, AFIPS Conf. Proc., Vol. 26, FJCC 1964, pp. 343-350.

H.E. Kulsrud, A General Purpose Graphic Language, Comm. of the ACM, April 1968, pp. 247-254.

M.H. Levin, An Introduction to Computer Graphic Terminals, IEEE Proc., Vol. 55, September 1967, pp. 1544-1552.

I.E. Sutherland, SKETCHPAD - A Man-Machine Graphical Communication System, AFIPS Conf. Proc., Vol. 23, SJCC 1963, pp. 329-346.

I.E. Sutherland, Computer Graphics, Datamation, May 1966, pp. 22-27.

A.H. Vorhaus, General Purpose Display System, Datamation, July 1966, pp. 59-64.

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* The MITRE Corporation Bedford, Massachusetts	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP N/A

3. REPORT TITLE

DISPLAY SOFTWARE TECHNIQUES FOR AFICCS

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

N/A

5. AUTHOR(S) *(First name, middle initial, last name)*

Thomas A. Mackey

6. REPORT DATE December 1968	7a. TOTAL NO. OF PAGES 47	7b. NO. OF REFS 16
8a. CONTRACT OR GRANT NO. AF19(628)-5165 b. PROJECT NO. 512V c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) ESD-TR-68-379 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* MTR-754	

10. DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES N/A	12. SPONSORING MILITARY ACTIVITY Directorate of Planning and Technology; Electronic Systems Division; Air Force Systems Command, L. G. Hanscom Field, Bedford, Mass.

13. ABSTRACT

This document presents a synopsis of the current display software package within AFICCS and recommendations for its improvement. The recommendations are based on the results of experiments conducted at the AFICCS Support Facility. The recommended improvements are independent of a particular AFICCS CPU; however, the display techniques are oriented to BR-90's and similarly configured display devices.

DD FORM 1 NOV 65 1473

Security Classification

Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Display Software						
AFICCS						
BR-90						

Security Classification