ESD-TR-70-124, Vol. II MTR-1768, Vol. II

GRASP: A PL/I COMPATIBLE GRAPHICS SUBROUTINE PACKAGE FOR THE IBM 2260 DISPLAY STATION (LOCAL ATTACHMENT)

VOLUME II - PROGRAM SPECIFICATIONS

R. H. Bullen, Jr.

MAY 1970

Prepared for

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

Project 512A
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract F19(628)-68-C-0365

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-70-124, Vol. II

MTR-1768, Vol. II

GRASP: A PL/I COMPATIBLE GRAPHICS SUBROUTINE PACKAGE FOR THE IBM 2260 DISPLAY STATION (LOCAL ATTACHMENT)

VOLUME II - PROGRAM SPECIFICATIONS

R. H. Bullen, Jr.

MAY 1970

Prepared for

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

Project 512A
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract F19(628)-68-C-0365

FOREWORD

This report was prepared by The MITRE Corporation, Bedford, Massachusetts under Contract No. F19(628)-68-C-0365, MITRE Project 512A. Volume I gives an overview of the IBM 2260 Display Station and an introduction to the PL/I compatible routines which provide programming support for the IBM 2260 in local attachment. Volume II provides detailed program specifications.

REVIEW AND APPROVAL

This technical report has been reviewed and is approved.

CHARLES E. MCKUSICK, Lt Colonel, USAF
Chief, Systems Analysis Division
Directorate of Systems Design and Development

ABSTRACT

GRASP is a set of PL/I compatible subroutines which provides programming support for the IBM 2260 Display Station in local attachment; i.e., the attachment of a 2260 directly to a System/360 CPU channel via the IBM 2848 Display Control. The subroutines are coded in OS/360 Assembler Language and are reentrant. They permit the PL/I programmer to manipulate the 2260 as an I/O device in the same manner available to the Assembler Language programmer using the Graphics Access Method under OS/360 (with restrictions as noted in the Introduction to Volume I of this document). All errors, except those which normally result in OS/360 abnormal ends (ABENDs), are returned to the user via subroutine parameters. GRASP is designed to operate under the MFT configuration of OS/360.

Volume I of this document gives an overview of the 2260 and an introduction to the GRASP routines. Volume II gives detailed program specifications.

TABLE OF CONTENTS

		Page
GLOSSARY		vi
SECTION I	GAAP - ACTIVATE ATTENTION PROCESSING	1
SECTION II	GAQ - ATTENTION INQUIRY	2
SECTION III	GCLOSE - CLOSE GRAPHICS DCB	4
SECTION IV	GDAP - DEACTIVATE ATTENTION PROCESSING	5
SECTION V	GERASE - INITIATE ERASE OPERATION	6
SECTION VI	GLINENO - LINE NUMBER TO CONTROL CHARACTER CONVERSION	7
SECTION VII	GNCP - NUMBER OF CHANNEL PROGRAM VALUE	8
SECTION VIII	GNUNITS - NUMBER OF UNITS DEFINED BY DCB	9
SECTION IX	GOPEN - OPEN GRAPHICS DCB	10
SECTION X	GREAD - INITIATE READ OPERATION	12
SECTION XI	GUNIT - RETURN NUMBER OF UNIT CAUSING ATTENTION	14
SECTION XII	GWAIT - WAIT FOR COMPLETION OF I/O OPERATION	15
SECTION XIII	GWRITE - INITIATE WRITE OPERATION	16

GLOSSARY

This section is provided as a quick reference to certain concepts which are used in Volume II of this document, but which are described more fully in Volume I. Each definition is followed by a page reference in Volume I where more information can be found.

Acronyms

DCB

Data Control Block; the primary link between the user program and the external graphics devices. Generated by calling GOPEN to initialize the graphics file, and specified in calls to the I/O initiation routines (GREAD, GERASE, GWRITE) and to GCLOSE to terminate I/O processing. (Volume I, page 10.)

DECB

Data Event Control Block; generated for each I/O request (GREAD, GERASE, and GWRITE) and specified to GWAIT to wait for I/O completion. Its primary use is to pass I/O request argument information to the wait routine. (Volume I, page 10.)

GACB

Graphics Attention Control Block; required for each DCB for which attentions are to be processed. Generated by calling GAAP to initiate the processing of attentions, and specified in calls to GAQ for attention inquiry and GDAP to terminate attention processing. (Volume I, page 10.)

NCP

Number of channel programs; an integer value which specifies maximum number of simultaneous I/O requests which may be active for a single DCB. May be specified on the DD statement by coding DCB=GNCP=value or in the call to GOPEN as an argument. If specified in the call to GOPEN, the value overrides any value on the DD statement. If specified on the DD statement, the "gncpvalue" argument to GOPEN must be zero. In this case, a GRASP function, GNCP, may be executed to determine the value coded on the DD statement. (Volume I, page 16.)

GLOSSARY (Continued)

Terms

Attention

An interrupt, generated by the display operator by depressing the ENTER key, which indicates to the program that data is available at the device for transfer to memory. (Volume I, page 14.)

Buffer Input

A mode of input provided in GRASP by the "B" code to the GREAD routine, which causes the entire contents of the display buffer, including special characters, to be transferred to memory. (Volume I, page 11.)

Keyboard Lock

An option available on Models 21 and 22 of the 2848 Display Control Unit which causes the keyboard to remain locked after I/O transfer. This option, implemented in GRASP as the "L" code to the GREAD, GERASE, and GWRITE routines, should be used only with these models of the 2848. (Volume I, page 6.)

Line Addressing

A mode of output, provided in GRASP by the "A" code to the GWRITE routine, which permits data display to begin at column 1 of a selected display line on the screen of a 2260. The specific line is selected by a control character which is sent to the device as the first byte of data. A GRASP function, GLINENO, is provided for translating line numbers to control characters. (Volume I, pages 6, 13.)

Line Number

An integer between 1 and 12 being the number of the display line on the screen of a 2260. See "Line Addressing". (Volume I, pages 6, 13.)

GLOSSARY (Concluded)

Terms (Continued)

Manual Input

A mode of input, provided on GRASP by the "M" code to the GREAD routine, in which only those characters on the display between the START (▶) and EOM (▬) symbols are transferred to memory. The START symbol is deleted from the screen. See "Short Manual Input". (Volume I, page 11.)

Short Manual Input

A mode of input, provided in GRASP by the "SM" code to the GREAD routine, which is identical to manual input with the exception that the START symbol is not deleted from the screen. This operation is, therefore, somewhat faster than manual input. (Volume I, page 12.)

Unit Number

An integer between 1 and 25, being an index into the list of devices allocated for a single DD statement. A function, GNUNITS, is available in GRASP for determining the number of devices allocated. The user program can then direct I/O requests to specific devices by using the "unit" argument of the GREAD, GWRITE, and GERASE routines. In addition, when an attention occurs, the user program may execute the function GUNIT which returns the number of the unit causing the attention. (Volume I, page 15.)

SECTION I

GAAP - ACTIVATE ATTENTION PROCESSING

The GAAP routine builds a GACB for the specified DCB and specifies that GACB to the control program. Any attentions occurring at devices defined by the specified DCB will be queued by the control program until the user calls the GAQ routine to inspect the attention queue. The format of the call is:

CALL GAAP (gacbptr, dcbptr, rtnval);

The arguments are defined as follows:

1. gacbptr - a PL/I POINTER variable which will be set by GAAP to the address of a GACB for the DCB pointed to by dcbptr. Attentions occurring at devices defined by the DCB are queued by the control program and may be inspected by calling GAQ with gacbptr specified as the first argument.

2. dcbptr - a PL/I POINTER variable previously set, by a call to GOPEN, to point to a valid grphics DCB.

3. rtnval - a FIXED BIN (31) variable set to indicate the results as follows:

   0 - normal return;

   1 - dcbptr does not point to a valid DCB;

   2 - a GACB has already been activated for this DCB; and

   3 - main storage was not available for control program requirements during activation of the GACB.

SECTION II

GAQ - ATTENTION INQUIRY

The GAQ routine (1) permits the selection of an attention from a specific device, (2) causes the user's program to wait pending a desired attention, or (3) clears the control program attention queue. The format of the call is:

CALL CAQ (gacbptr, mode, cond, unit, rtnval);

The arguments are defined as follows:

1. gacbptr - a PL/I POINTER variable specifying a valid GACB. The value of gacbptr must have been set by a call to GAAP to initiate attention processing.

2. mode - a CHAR(1) variable or constant specifying the mode of the attention query:

   'R' - relinquish control;

   'W' - wait for an attention;

   'C' - test for an attention; and

   'X' - clear all attentions.

3. cond - a BIT(1) variable, ignored unless mode = 'C' is specified, giving the result of the conditional test:

   '1'B - the requested attention is present; and

   'Ø'B - the requested attention is not present.

4. unit - if mode = 'C' or mode = 'W', unit may specify a particular unit number (1 ≤ unit ≤ 25) or may be equal to 0 to imply all units. If mode = 'X' or mode = 'R', unit must be Ø. Unit is a FIXED BIN (31) variable or constant.

5. rtnval - a FIXED BIN (31) variable set to indicate the results as follows:

0 - normal return ;

1 - mode not CHAR(1) ; and

2 - (1) if mode = 'C' or 'W' then unit not in range Ø ≤ unit ≤ 25; and

(2) otherwise unit ¬= Ø.

The processing performed is a function of the value of the mode argument. Each of the modes is discussed below:

1. Relinquish mode (mode = 'R') - this mode is provided for compatibility with the asynchronous attention capability which is not currently available in GRASP. When coded in the user's processing program, it is treated as a wait mode.

2. Wait mode (mode = 'W') - the program issuing this mode of GAQ is placed in a wait state until the specified attention occurs.

3. Conditional mode (mode = 'C') - the COND bit is set to indicate whether or not the specified attention is present at the device. If an attention is present, the call to GAQ should be followed by a read (GREAD) to the device or the attention will be lost.

4. Clear mode (mode = 'X') - all outstanding attentions are cleared from the control program attention queue.

The call to GAQ must have been preceded by a call to GAAP to initialize attention processing.

SECTION III

GCLOSE - CLOSE GRAPHICS DCB

This subroutine causes the specified graphics DCB to be closed and its associated storage freed. The format of the call is:

```
CALL GCLOSE(dcbptr);
```

The argument is defined as follows:

1. dcbptr - a PL/I POINTER variable pointing to a valid, open DCB.

Any attention specifications which had been active for the DCB must have been deactivated by a call to GDAP. If dcbptr does not point to a valid DCB, an OS/360 Abend may occur.

0 - normal return ;

1 - mode not CHAR(1) ; and

2 - (1) if mode = 'C' or 'W' then unit not in range Ø ≤ unit ≤ 25; and

(2) otherwise unit ¬= Ø.

The processing performed is a function of the value of the mode argument. Each of the modes is discussed below:

1. Relinquish mode (mode = 'R') - this mode is provided for compatibility with the asynchronous attention capability which is not currently available in GRASP. When coded in the user's processing program, it is treated as a wait mode.

2. Wait mode (mode = 'W') - the program issuing this mode of GAQ is placed in a wait state until the specified attention occurs.

3. Conditional mode (mode = 'C') - the COND bit is set to indicate whether or not the specified attention is present at the device. If an attention is present, the call to GAQ should be followed by a read (GREAD) to the device or the attention will be lost.

4. Clear mode (mode = 'X') - all outstanding attentions are cleared from the control program attention queue.

The call to GAQ must have been preceded by a call to GAAP to initialize attention processing.

SECTION III

GCLOSE - CLOSE GRAPHICS DCB

This subroutine causes the specified graphics DCB to be closed and its associated storage freed. The format of the call is:

```
CALL GCLOSE(dcbptr);
```

The argument is defined as follows:

1. dcbptr - a PL/I POINTER variable pointing to a valid, open DCB.

Any attention specifications which had been active for the DCB must have been deactivated by a call to GDAP. If dcbptr does not point to a valid DCB, an OS/360 Abend may occur.

SECTION IV

GDAP - DEACTIVATE ATTENTION PROCESSING

The GDAP routine notifies the control program that attentions should no longer be accepted from devices defined by the DCB associated with the specified GACB. Storage occupied by the GACB is freed. The format of the call is:

```
CALL GDAP (gacbptr, rtnval);
```

The arguments are defined as follows:

1. gacbptr - a PL/I POINTER variable set by a previous call to GAAP to the address of a GACB.

2. rtnval - a FIXED BIN (31) variable set to indicate the results as follows:

   0 - normal return;

   1 - the DCB associated with the GACB pointed to by gacbptr is invalid; and

   2 - the GACB pointed to by gacbptr has not been specified in a call to GAAP.

The two error returns from GDAP generally indicate that control blocks have been overwritten.

SECTION V

GERASE - INITIATE ERASE OPERATION

The GERASE subroutine causes the screen of a 2260 to be erased. It must be followed by a call to GWAIT to ensure completion. The format of the call is:

CALL GERASE (decbptr, lockcode, dcbptr, unit, rtnval);

The arguments are defined as follows:

1. decbptr - a PL/I POINTER variable which will be set by the GERASE subroutine to the address of a DECB.

2. lockcode - a CHAR(1) VARYING string specifying the keyboard lock option:

   '' - no lock; and

   'L'- lock keyboard.

3. dcbptr - a PL/I POINTER variable which has been set by a call to the GOPEN routine to the DCB address.

4. unit - a FIXED BIN(31) variable or constant specifying the unit number (1 ≤ unit ≤ 25).

5. rtnval - a FIXED BIN(31) variable which is set to indicate the results:

   Ø - normal return;

   1 - lockcode incorrect; and

   2 - unit not in range 1 ≤ unit ≤ 25.

The GERASE subroutine generates a DECB and issues the GCNTRL macroinstruction. The address of the DECB is passed to the calling program.

The status of the erase request is not known until a call is made to GWAIT.

SECTION VI

GLINENO - LINE NUMBER TO CONTROL CHARACTER CONVERSION

This function translates a FIXED BIN (31) line number into the CHAR(1) control character for line addressing output. The format of the function reference is:

```
...GLINENO (line_number)...
```

The argument and returned value are defined as follows:

1. line_number - A FIXED BIN (31) variable or constant, 1 ≤ line number ≤ 12.

2. returned value - CHAR(1) defined as follows:

line number	returned value (in hexadecimal)
1	FØ
2	F1
3	F2
4	F3
5	F4
6	F5
7	F6
8	F7
9	F8
10	F9
11	FA
12	FB

If line_number is out of range FØ is returned.

SECTION VII

GNCP - NUMBER OF CHANNEL PROGRAM VALUE

This is a function which returns the NCP value of a graphics DCB. The DCB must have been previously opened by a call to GOPEN. The executing format is:

```
...GNCP (dcbptr)...
```

The argument and returned value are defined as follows:

1. dcbptr - a PL/I POINTER variable previously set to point to a graphics DCB by the GOPEN routine.

2. returned value - a FIXED BIN (31) variable set to the GNCP value.

Programming notes:

1. This function is useful in an application which permits the specification of the NCP value on the DD statement.

2. If dcbptr does not point to a graphics DCB, no error results, but the returned value is undefined.

SECTION VIII

GNUNITS - NUMBER OF UNITS DEFINED BY DCB

GNUNITS returns a FIXED BIN (31) value which is the number of 2260 devices defined by a DD statement. Since the input to this function is the ddname rather than a pointer to a DCB, the function may be executed at any time whether or not a file is open for the DD statement. The format of the function reference is:

```
...GNUNITS (ddname)...
```

The argument and returned value are defined as follows:

1. ddname - a CHAR(8) variable or constant, the value of which is the ddname of a DD statement defining one or more 2260 Display Stations.

2. returned value - a FIXED BIN (31) value being the number of units defined by the DD statement.

If no DD statement exists, a value of zero is returned. If the DD statement does not define graphics devices, the returned value is undefined.

SECTION IX

GOPEN - OPEN GRAPHICS DCB

This subroutine causes a graphics DCB to be generated and opened for a specified ddname. The calling format is:

CALL GOPEN (dcbptr, ddname, options, attntype, qncpvalue, Ø, rtnval);

The arguments are defined as follows:

1. dcbptr - a PL/I POINTER variable set by the GOPEN routine to the address of the generated graphics DCB.

2. ddname - a CHAR(8) variable or constant specifying the ddname describing the 2260's to be opened.

3. options- a CHAR(3) VARYING variable or constant specifying the open options. The possible values are:

   'I' - only read operations will be issued;

   'O' - only write operations will be issued;

   'IO' - both read and write operations will be issued;

   'IE' - read and erase operations (i.e., GERASE) will be issued;

   'OE' - write and erase operations (i.e., GERASE) will be issued;

   'IOE' - read, write, and erase operations (i.e., GERASE) will be issued.

4. attntype - a string variable or constant specifying the type of attention handling to be performed. Currently, this must be CHAR(5) and have the value 'BASIC'.

5. qncpvalue- the NCP value. A FIXED BIN(31) variable or constant specifying the maximum number of I/O statements which will be executed for this DCB before a wait.

6. (not used) - This argument is provided for future expansion; it can be coded as a zero constant.

7. rtnval - a FIXED BIN(31) variable set by the open routine to indicate the results:

   Ø - normal completion ;

   1 - ddname not CHAR(8) ;

   2 - options invalid ;

   3 - attntype not 'BASIC' ;

   4 - ddname not found ;

   5 - device not a 2260 ;

   6 - open not successful; and

   7 - disastrous error .

The DCB generated by this routine is opened and its address is placed in dcbptr. I/O operations for the device(s) specified by ddname must specify this dcbptr as an argument. The dcbptr is then specified in a call to GCLOSE, where the DCB is closed and the storage is freed. Programming notes:

1. The qncpvalue may also be specified on the DD statement. In this case, the qncpvalue must be equal to Ø in the call to GOPEN.

SECTION X

GREAD - INITIATE READ OPERATION

GREAD causes a read operation to be initiated for a 2260 graphics device. A DECB is acquired and formatted and the address of the DECB is returned to the calling program. This address may then be specified in a call to GWAIT. The format of the call is as follows:

CALL GREAD (decbptr, read type, dcbptr, string, unit, rtnval);

The arguments are defined as follows:

1. decbptr - a PL/I POINTER variable set by the GREAD subroutine to point to the generated DECB.

2. readtype- a CHAR(3) VARYING variable or constant specifying the type of read:

   'M' - manual input;

   'ML' - manual input, keyboard lock;

   'SM' - short manual input;

   'SML' - short manual input, keyboard lock;

   'B' - buffer input; and

   'BL' - buffer input, keyboard lock.

3. dcbptr - a PL/I POINTER variable set by a call to GOPEN to the address of a graphics DCB.

4. string - a CHAR(n)VARYING string variable to receive the characters read.

5. unit - a FIXED BIN(31) variable or constant specifying the unit number (1 ≤ unit ≤ 25).

6. rtnval - a FIXED BIN(31) variable set to indicate the results as follows:

   Ø - normal completion

   1 - readtype invalid

   2 - unit not in the range 1 ≤ unit ≤ 25.

3 - max length(string)¬ = |240,480,960| when
read type = 'B' or 'BL'.

A GREAD macroinstruction is issued on the generated DECB. The address of the "string" is used as the area address. The maximum length of the string is used as the length specification.

If 'B' or 'BL' is specified, the maximum length of the string is compared to the buffer size of the addressed 2260. The 2848 model number is used to determine the buffer size as follows:

2848 Model 1	240
2848 Model 2	480
2848 Model 3	960
2848 Model 21	240
2848 Model 22	480

The status of the operation is not known until a call is made to GWAIT.

SECTION XI

GUNIT - RETURN NUMBER OF UNIT CAUSING ATTENTION

GUNIT returns a FIXED BIN(31) value which is the unit number of the 2260 Display Station causing an attention. The value returned by GUNIT is undefined unless it is preceded by a call to GAQ to determine that an attention exists. The format of the function reference is:

```
...GUNIT (gacbptr)...
```

The argument and returned value are defined as follows:

1. gacbptr - a PL/I POINTER variable pointing to the GACB associated with the DCB for which an attention has occurred; and

2. returned value - a FIXED BIN(31) value between 1 and 25 being the unit number of the 2260 causing the attention.

SECTION XII

GWAIT - WAIT FOR COMPLETION OF I/O OPERATION

A call to GWAIT is required for every call to GREAD, GWRITE, or GERASE. It is required to ensure that the operation is complete, to retrieve any error conditions, and, in the case of GREAD, to ensure that the current length of the specified string is set. The format of the call is as follows:

```
CALL GWAIT (decbptr, rtnval);
```

The arguments are defined as follows:

1. decbptr - a PL/I pointer variable which has been set by GREAD, GWRITE, or GERASE to point to a DECB.

2. rtnval - a FIXED BIN(31) variable set to indicate the results as follows:

   Ø - normal completion; if the I/O operation was a read, the current length of the specified string has been set to the number of characters read;

   1 - (GREAD only) - more than "n" characters were present at the device, where n is the maximum length of the string specified in the read operation. Only n characters are transferred from the device; the remaining characters are effectively lost;

   2 - permanent error; and

   3 - (GREAD only) - no data was present at the device. This can occur on a manual input read if no START symbol is on the screen.

SECTION XIII

GWRITE - INITIATE WRITE OPERATION

A call to the GWRITE subroutine causes a write operation to be initiated to a 2260 Display Station. A DECB is generated and its address is passed to the calling program. The DECB may then be passed to the GWAIT subroutine to wait for the completion of the event. The format of the call is:

```
CALL GWRITE (decbptr, writetype, dcbptr, string, unit, rtnval);
```

The arguments are defined as follows:

1. decbptr - a PL/I POINTER variable which will be set by the GWRITE subroutine to point to the generated DECB.

2. writetype - a CHAR(3) VARYING variable specifying the type of write:

   'B' - write buffer;

   'BL' - write buffer, keyboard lock;

   'EB' - erase, write buffer;

   'EBL' - erase, write buffer, keyboard lock;

   'A' - write line address;

   'AL' - write line address, keyboard lock;

   'EA' - erase, write line address;

   'EAL' - erase, write line address, keyboard lock;

3. dcbptr - a PL/I POINTER variable set by a call to GOPEN to point to a graphics DCB.

4. string - a CHAR(n) VARYING string variable containing the characters to be written.

5. unit - a FIXED BIN(31) variable or constant specifying the unit number (1 ≤ unit ≤ 25).

6. rtnval - a FIXED BIN(31) variable set to indicate the results as follows:

Ø - normal completion;

1 - writetype invalid;

2 - unit not in range $1 \le \text{unit} \le 25$;

3 - current length (string) ¬= $|240,480,960|$

when writetype = 'B', 'BL', 'EB', or 'EBL'; and

4 - current length (string) = 0.

A GWRITE macroinstruction is issued on the generated DECB. The address of the string is used as the area address and the length specification is the current length of the string.

If 'B', 'BL', 'EB', or 'EBL' is specified, the current length of the string must be equal to the buffer size of the addressed 2260 (see GREAD).

If 'A', 'AL', 'EA', or 'EAL' is specified, the first byte of data to be written (which must be included in the current length of the string) indicates the line address (see GLINENO).

The status of the operation is not known until a GWAIT subroutine is issued.

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

3. REPORT TITLE

GRASP: A PL/I COMPATIBLE GRAPHICS SUBROUTINE PACKAGE FOR THE IBM 2260 DISPLAY STATION (LOCAL ATTACHMENT)
VOLUME II - PROGRAM SPECIFICATIONS

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

N/A

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

Richard H. Bullen, Jr.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
MAY 1970	25	0

8a. CONTRACT OR GRANT NO.

F19(628)-68-C-0365

b. PROJECT NO.

512A

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)

ESD-TR-70-124, Vol. II

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

MTR-1768, Vol. II

10. DISTRIBUTION STATEMENT

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

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY

Directorate of Systems Design and Development, Electronic Systems Division, AF Systems Command, L. G. Hanscom Field, Bedford, Massachusetts

13. ABSTRACT

GRASP is a set of PL/I compatible subroutines which provide programming support for the IBM 2260 Display Station in local attachment; i.e., the attachment of a 2260 directly to a System/360 CPU channel via the IBM 2848 Display Control. The subroutines are coded in OS/360 Assembler Language and are reentrant. They permit the PL/I programmer to manipulate the 2260 as an I/O device in the same manner available to the Assembler Language programmer using the Graphics Access Method under OS/360 (with restrictions as noted in the Introduction to Volume I of this document). All errors, except those which normally result in OS/360 abnormal ends (ABENDS), are returned to the user via subroutine parameters. GRASP is designed to operate under the MFT configuration of OS/360.

Volume I of this document gives an overview of the 2260 and an introduction to the GRASP routines. Volume II gives detailed program specifications.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
COMPUTER DISPLAYS						
PROGRAMMING IBM 2260						
PL/I COMPATIBLE GRAPHICS SUBROUTINES						
PL/I GRAPHICS ACCESS METHOD						