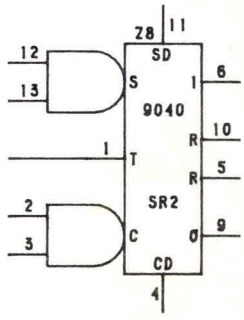
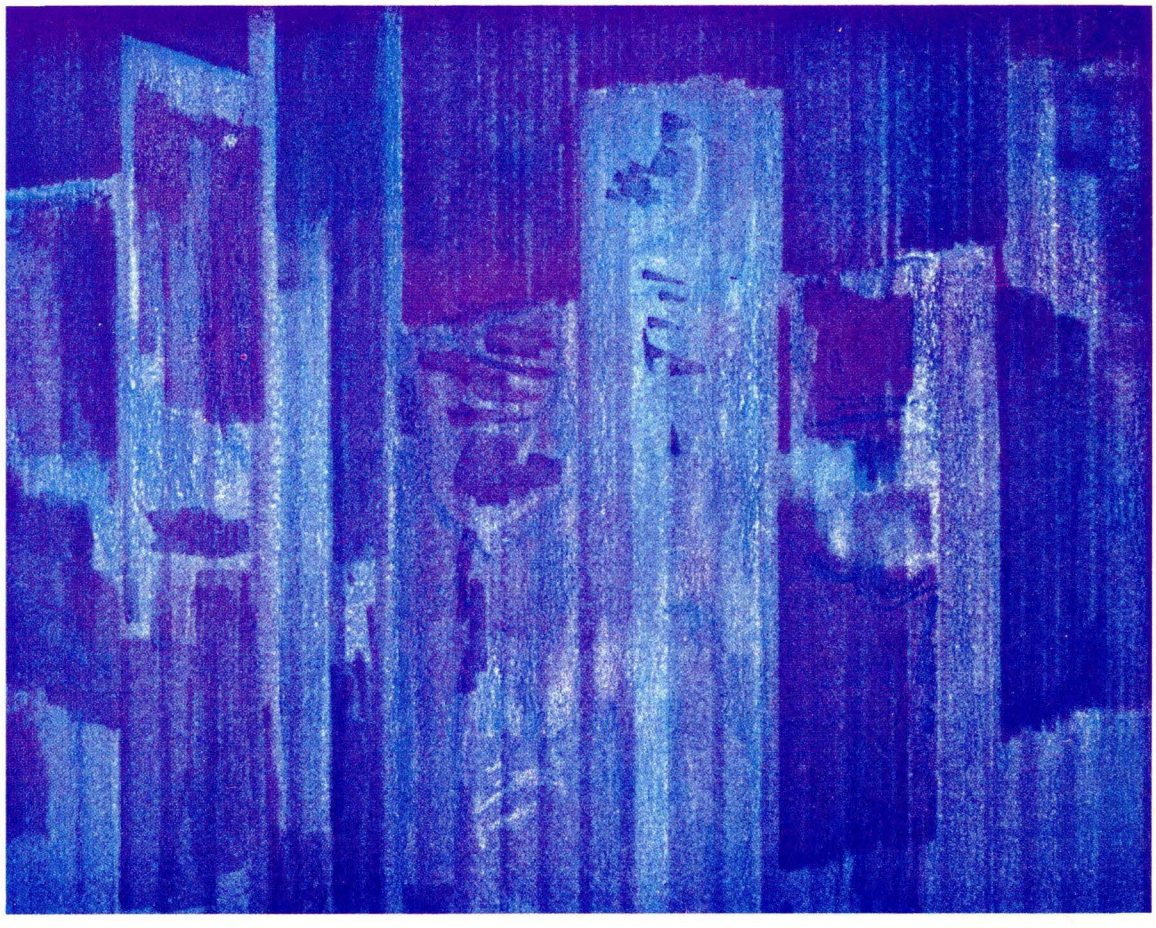
CALCOMP

APPLICATIONS
SOFTWARE





A.S.A.P.

AUTOMATED SYMBOLIC
ARTWORK PROGRAM

VOLUME I
SECTIONS 1-9

A.S.A.P.

AUTOMATIC SYMBOLIC ARTWORK PROGRAM

VOLUME I

CALCOMP APPLICATIONS SOFTWARE

July, 1974

CALCOMP
California Computer Products, Inc.
2411 W. La Palma, Anaheim, CA 92801

CONTENTS

Section		Page
1	INTRODUCTION	1-1
2	SYSTEM FLOW	2-1
	A.S.A.P. System Flow	2-1
	Translator	2-1
	Editor	2-1
	Drawing Processor	2-1
	Report Generator	2-3
	Symbol File	2-3
3	CONCEPTS	3-1
	Digitizer Table Layout	3-1
	Menu Area	3-1
	Menu Positioning	3-1
	Symbol Menu Area	3-1
	Control Menu Area	3-3
	Initialization Control Functions	3-3
	Drawing Control Functions	3-3
	Mode Control Functions	3-3
	Mode Option Controls	3-3
	Direction Compass Controls	3-3
	Macro Definition Control	3-5
	Error-Reset Control	3-5
	Keyboard Area	3-5
	Work Area	3-5
	Symbols	3-5
	Symbol Orientation	3-5
	Macros	3-7
	Macro Orientation	3-7
	Drawing Orientation	3-7
	Annotation	3-8
	Variable Annotation	3-8
	Junction Points	3-8
	Editing	3-8
	A.S.A.P. Language	3-9
	Expansion of Drawings and Macros	3-9
	Reports and Output Listings	3-9
	Windowing	3-10
4	USING A.S.A.P.	4-1
	A.S.A.P. System Control Card ($ASAP card)	4-1
	A.S.A.P. Language	4-3
	A.S.A.P. Commands	4-4
5	EDITING	5-1
	Digitizer Correction Cards	5-1
	A.S.A.P. Language Editing	5-3
	Begin and End Edit Cards	5-3

CONTENTS (Continued)

Section		Page
	Drawing and Macro Editing	5-4
	Card Editing	5-4
	String Editing	5-5
	Symbol Editing	5-6
6	PROCESSING	6-1
	A.S.A.P. Processing	6-1
	Plotting a Drawing	6-1
	Drawing and Macro Expansion	6-1
7	SYMBOL FILE	7-1
	General Description	7-1
8	SYMBOL CREATION	8-1
9	ERROR MESSAGES	9-1
	Descriptive Error Messages	9-1

NOTE

The A.S.A.P. manual is in two volumes. Volume I contains Sections 1 through 9. Volume II contains Sections 10 through 13.

ILLUSTRATIONS

Figure		Page
2-1	A.S.A.P. Flowchart	2-2
3-1	Digitizer Table Layout	3-2
3-2	A.S.A.P. Control Menu	3-4
3-3	Symbol Mirroring	3-6
3-4	Symbol Mirroring With Rotation	3-6
3-5	Sample of Mirror and Rotation	3-6
3-6	Macro As Defined	3-7
3-7	Macro Placed With X and Y Mirrors	3-7
3-8	Macro Rotated at an Angle of 180 Degrees	3-7
8-1	OAMP	8-5

TABLES

Table		Page
4-1	A.S.A.P. Language Punctuation (IBM-370 Version)	4-4
4-2	Letter Set Types	4-35
4-3	Line Types	4-40

Section 1
INTRODUCTION

The CalComp Automated Symbolic Artwork Program (A.S.A.P.) is a software system designed to produce high quality finished drawings on a CalComp graphic system from data obtained by digitizing rough sketches. A.S.A.P. fulfills this requirement for the user who has access to a large host computer and desires to digitize, draw, and save the graphic data describing any repetitive type of symbolic graphics task.

The A.S.A.P. program consists of four computer modules referred to as the Translator, Editor, Drawing Processor, and Report Generator, and a related Symbol File, all of which are described in Section 2. These modules can handle a wide variety of applications and provides the user with consistent quality drawings, storage of drawing data for reproduction, exclusive editing techniques, interchangeable symbol libraries, and compatibility between digitizing and hand coding.

A.S.A.P. hardware requirements include:

- Minimum 32K-word memory (32 bit or larger)
- Random-access storage device
- Plot data output device
- Digitizer

A.S.A.P. software support requirements include:

- FORTRAN compiler
- Overlay loader
- Random-access I/O capabilities
- CalComp Host Computer Basic Software

Section 2
SYSTEM FLOW

A.S.A.P. SYSTEM FLOW

In order to produce a drawing from digitized information, A.S.A.P. functions in four logically distinct operations. These operations consist of: (1) the translation of the digitized information into a meaningful language description of the drawing, (2) the maintenance of this language on a random access file called the Symbol file, (3) the actual production of the drawing, and (4) the generation of various reports describing the drawing contents. Throughout the following discussion, these A.S.A.P. operations will be referred to as: (1) Translator, (2) Editor, (3) Drawing Processor, and (4) Report Generator (See Figure 2-1).

TRANSLATOR

Input to the Translator is in the form of coordinate data produced by a digitizer using an A.S.A.P. Control Menu and Symbol Menu. The digitizer is used offline and normally produces a card deck containing the coordinate data. The Translator accepts and interprets this coordinate data and generates an A.S.A.P. language data file. The data file is in a form acceptable to the Editor and can contain original drawings and macros (subdrawings) or modifications to existing drawings and macros. During this translation process, the Translator makes corrections for paper shrinkage or paper expansion and drawing alignment on the digitizer table. The Translator can make additions or modifications to the Symbol Menu(s) and write these changes on the Symbol File. Digitizing errors that may cause fatal errors in processing can be corrected by hand coding correction cards which, unlike raw coordinate data, are English language descriptions of the changes to be made.

EDITOR

The Editor is used to maintain the Symbol File. It accepts A.S.A.P. language data and Symbol data and places it into the Symbol File. A.S.A.P. language data may come from the Translator, from the Drawing Processor, or from hand-coded cards. Symbol data comes from hand-coded cards only. The Editor enables the user to insert, modify, replace, or delete symbols, macros, and drawing data in the Symbol File. These operations may be performed using card or string editing. Card editing allows the user to add, delete, or replace individual card images in a macro or drawing, whereas string editing allows specified sequences of alphanumeric characters to be replaced within a new sequence of characters with macros or drawings.

DRAWING PROCESSOR

In the A.S.A.P. language, a drawing is passed from the Symbol File to the Drawing Processor by the Editor. The Drawing Processor interprets the A.S.A.P. language data and produces a drawing on a CalComp plotter. The actual processing procedures may vary within the Drawing Processor depending on the various A.S.A.P. language commands utilized. These procedures include expansion of called macros, the creation of plotting commands, and the creation of a file indicating A.S.A.P. language modifications due to windowing commands

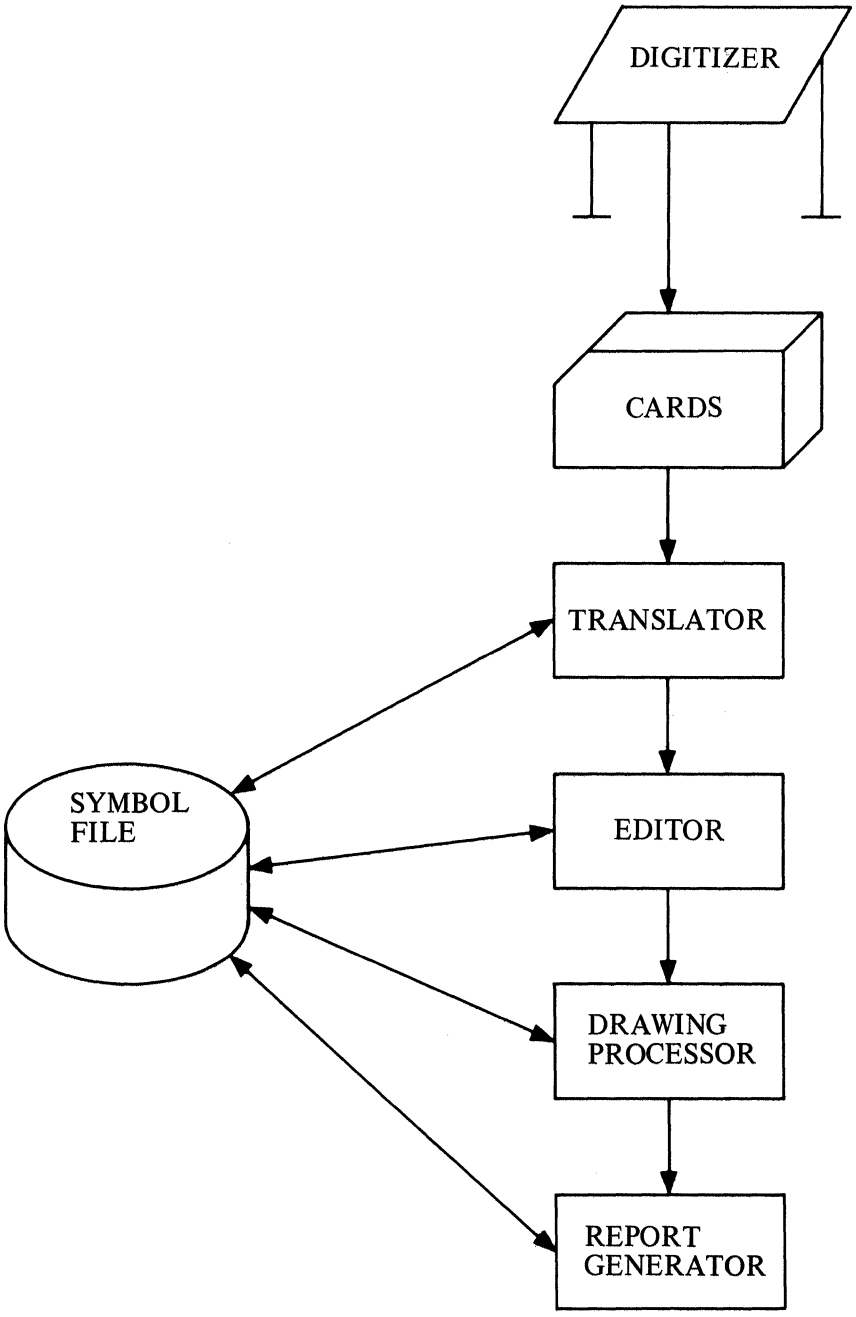


Figure 2-1. A.S.A.P. Flowchart

and continuity commands. A coordinate file is also produced which contains all of the X-Y coordinates generated in creating the desired plot. This file is sorted and used during the continuity processing and report generation.

Upon completion of the Drawing Processor phase, the Editor is recalled to update the existing drawing in the symbol file including all changes generated.

REPORT GENERATOR

The Report Generator accepts as input the sorted coordinate files. From these files, a component report and a coordinate report can be generated.

The Component report lists symbols and called macros, sorted alphabetically by name, with their coordinate location on the drawing and the sequence numbers of the A.S.A.P. language card images that placed them.

The X-Y Coordinate report lists all coordinates sorted on increasing X and decreasing Y, the type of A.S.A.P. command which defined the coordinate, and the sequence numbers of the A.S.A.P. language card images that placed the coordinate.

SYMBOL FILE

The Symbol File is a multi-purpose data file used by all phases of the program but maintained only by the Editor. Contained in the Symbol File is drawing data, symbol data, macro data, and Symbol Menu data. The Symbol File is a random access file and as such must reside on a device with random access capability, such as a disk storage unit.

Section 3
CONCEPTS

This section discusses the various concepts found within A.S.A.P.

- Digitizer Table Layout
- Symbols
- Macros
- Annotation
- Editing
- Expansion of Drawings and Macros
- Reports and Output Listings

DIGITIZER TABLE LAYOUT

The digitizer table area is divided into two functionally distinct areas (See Figure 3-1). These areas are the menu area and the work area. Once the menu area has been positioned (see MENU POSITIONING), the remainder of the digitizer table becomes the work area. It should be noted that, prior to performing any digitizing, a digitizer origin should be established. The procedure for establishing a digitizer origin may vary from digitizer to digitizer, but all successive digitizing will be relative to the digitizer origin. Should the digitizer lose origin while digitizing a drawing, the digitizer origin must be reestablished prior to further digitizing. Therefore, when the initial digitizer origin is established, it should be marked on the digitizer table.

MENU AREA

The menu area is that area through which the user "communicates" with A.S.A.P. while digitizing a drawing. This area (which is approximately 8 inches by 23½ inches on the digitizer table) is divided into two functionally distinct areas: the Symbol menu (the upper 8 inches by 13½ inches of the menu area) and the Control menu (the lower 8 inches by 11 inches of the menu area). The menu area can be placed anywhere on the digitizer table; however, the bottom line of the menu must be parallel to the X-axis of the digitizer table.

MENU POSITIONING

Once the menu has been placed on the digitizer table and the digitizer origin established, the lower left-hand corner of the menu must be digitized. This indicates the position of the menu on the digitizer table to A.S.A.P.

SYMBOL MENU AREA

The Symbol menu area is a 12-row by 7-column grid forming 84 one-inch squares. Each square can contain the name and/or picture of a symbol or macro available for placement on a drawing.

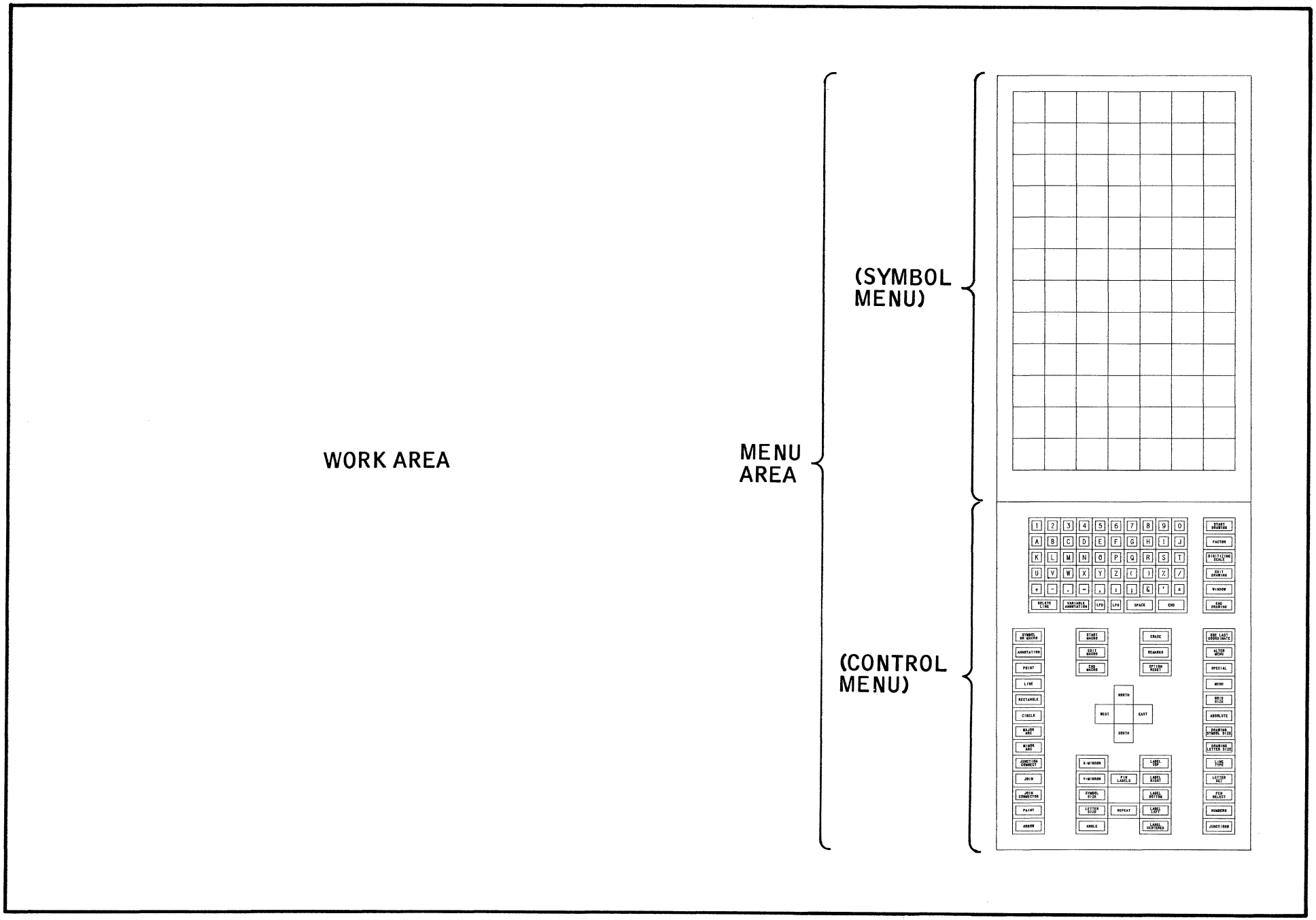


Figure 3-1. Digitizer Table Layout

The top 10 rows of the Symbol menu constitute the permanent Symbol menu area containing a maximum of 70 symbols and/or macros. Each Symbol menu is identified by a number from 1 to 99. When a new Symbol menu number is specified, a drawing of that Symbol menu is placed on top of the previous menu.

The bottom two rows of the Symbol menu are the temporary Symbol menu area. The assignment of symbol and macro to this area applies until END DRAWING is accessed.

To select a Symbol menu, see MENU in Section 4. To create or modify a Symbol menu, see ALTER MENU in Section 4.

CONTROL MENU AREA

The Control menu area is that area through which the user controls or directs the system while digitizing a drawing. The items on the Control menu area are grouped by control function type. These types are: (1) Initialization Control, (2) Drawing Control, (3) Mode Control, (4) Mode Option Control, (5) Direction Compass, (6) Macro Definition Control, (7) Error Reset Control, and (8) Keyboard Area. (See Figure 3-2.)

Initialization Control Functions

The initialization control functions (Start Drawing, Factor, Window, etc.) are used to specify information to the system about the current drawing or macro to be digitized. Specifications made within this area, when used, must be accessed only at the beginning of each drawing specified, or at the beginning of each macro specified.

Drawing Control Functions

Functions within the Drawing Control area (Grid Size, Line Type, Menu, etc.) if used, may be accessed at any time during drawing creation (assuming initialization control functions are completed). When a particular drawing control function is specified, that function remains in effect until it is respecified or reset (see ALTER MENU and SPECIAL in Section 4 for possible exception).

Mode Control Functions

Functions within the Mode Control Area (Symbol or Macro, Circle, Arrow, etc.) indicates to A.S.A.P. the type of placements that are to follow. Selection of a Mode Control function terminates the previously accessed Mode Control function. All options (see Mode Options and Direction Compass Controls) are reset to their default values when a mode control access is made.

Mode Option Controls

Functions with the Mode Option Control area (Symbol Size, Label Top, Letter Size, etc.) apply to the various Mode Control functions. The action each function produces depends upon the mode control currently in effect. The Mode Option controls may be respecified or reset at any time during mode control operations. Once specified, a Mode Option control remains in effect until it is reset or respecified or until a mode control is specified.

Direction Compass Controls

The Direction Compass Controls (North, East, South, West) are used to specify the direction that symbols, macros, and/or annotation are to be placed on the drawing relative to the placement coordinates specified. Like the Mode Option Controls, the Direction Compass Controls remain in effect until the direction is reset or until a mode control is changed. If the direction compass is not accessed while in SYMBOL OR MACRO or ANNOTATION mode controls, the default direction will be East.

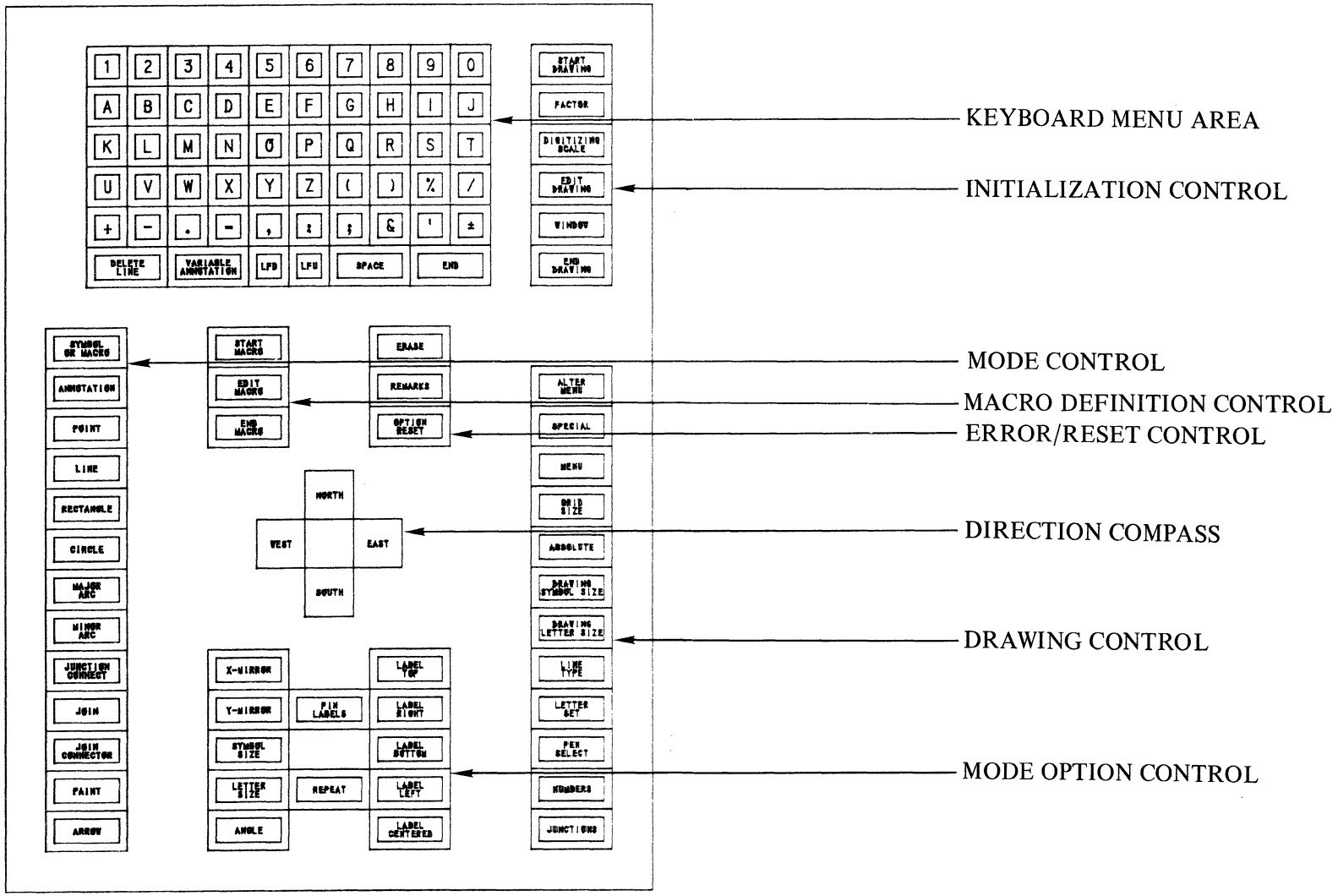


Figure 3-2. A.S.A.P. Control Menu

Macro Definition Control

The Macro Definition Control functions (Start Macro, Edit Macro, End Macro, etc.) are used to create or modify macros. A macro is defined as a subdrawing or combination of various mode control functions (symbols, lines, circles, other macros, etc.) which, once defined, can be used on this and subsequent drawings.

Error-Reset Control

The Error-Reset Control functions (Erase, Remarks, Option Reset, etc.) are used to: (1) delete previous digitizer accesses, (2) document the A.S.A.P. language file being created, and (3) default or reset various mode option controls or drawing controls.

Keyboard Area

The keyboard menu area contains numeric, alphabetic, and special characters and related control functions. This area is used to specify: numeric values for various digitizing controls, annotation to be placed on the drawing, remarks to document the drawing, etc.

WORK AREA

The work area is the area on which the drawing will be "digitized". Digitizer accesses in this area involve the placement of symbols, macros, lines, annotation, etc. on the drawing. A new drawing to be digitized is normally a roughly-sketched drawing on gridded paper. This drawing is placed on the work area of the digitizer table. To modify an existing drawing, the plot of the drawing is placed on the digitizer and the appropriate changes are then digitized.

SYMBOLS

The basic components used in creating drawings are called symbols. These symbols may be composed of line segments, circles, arcs, and alphanumeric characters. Each symbol in a given symbol library must be assigned a unique code name. By specifying that code name, or by selecting the symbol from a symbol menu, the user causes A.S.A.P. to select the symbol from the library and draw it. (The creation of symbols is described in Section 8.)

The points at which applicable input/output leads join a given symbol are called nodes. The node at which a symbol begins is called the entry node. The node at which a symbol ends is called the exit node. All other nodes are called auxiliary nodes. A.S.A.P. uses these nodes for the interconnection of symbols on a drawing.

SYMBOL ORIENTATION

In analyzing the drawing, it may be determined that a given symbol in the library may have to be arbitrarily rotated and/or mirrored to produce the desired view. To do this, one of four directions; none, one, or two mirrors; and/or an angle must be specified.

Part 1 of Figure 3-3 shows the transistor symbol as it appears in the library. Parts 2, 3, and 4 illustrate the effect produced by mirroring the symbol. If the direction specified is East, the symbols are placed starting at Entry Node A and ending at Exit Node B. If the direction specified had been West, the symbols would have been placed starting at Point B and ending at Point A. (Note that the symbols are *not* rotated.)

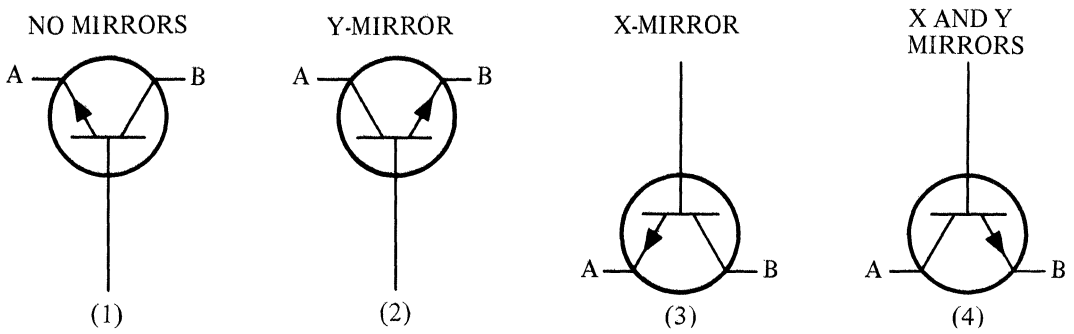


Figure 3-3. Symbol Mirroring

Figure 3-4 shows the transistor symbol drawn at a rotation of 90 degrees. If the direction specified is North, the symbols are placed starting at Entry Node A and ending at Exit Node B. If the direction specified had been South, the symbols would have been placed starting at Point B and ending at Point A.

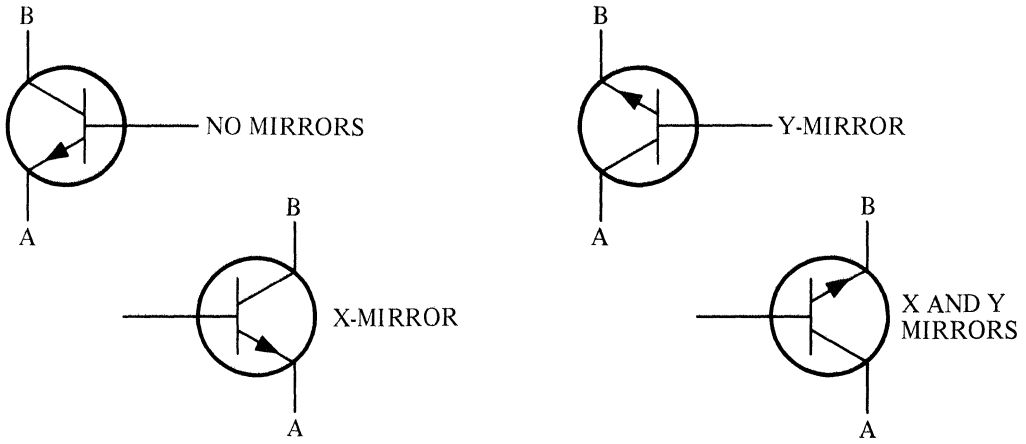


Figure 3-4. Symbol Mirroring With Rotation

It is important to note that when selecting the desired view, the mirrors must first be applied to the symbol as it appears in the library before selecting the direction and angle at which it is to be drawn.

In addition to the views illustrated in Figures 3-3 and 3-4, symbols may be drawn at any angle as specified by the user. For example, a symbol may be X-mirrored, drawn North, and at an angle of 30 degrees, as illustrated in Figure 3-5.

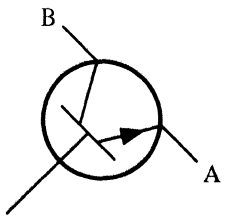


Figure 3-5. Sample of Mirror and Rotation

MACROS

If a drawing contains a pattern that is to be repeated on that drawing or other drawings, it may be identified to A.S.A.P. as a macro. A macro is assigned a unique name and may consist of lines, circles, annotation, symbols, or other macros.

MACRO ORIENTATION

In analyzing the drawing, it may be determined that a macro may have to be arbitrarily rotated and/or mirrored to produce the desired view. To do this, one of four directions; none, one, or two mirrors; and/or an angle must be specified.

Specifying a direction of East causes a macro to be placed as it has been defined (See Figure 3-6). A direction of North causes the macro to be drawn at a rotation of 90 degrees; West a rotation of 180 degrees; and South a rotation of 270 degrees. In all cases, everything in the macro is rotated except the annotation.

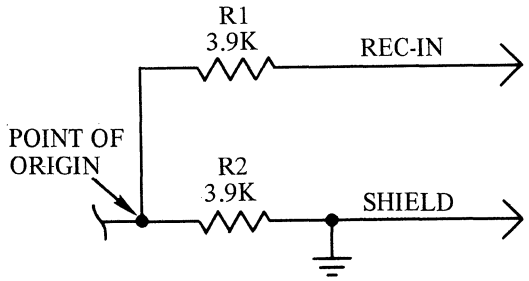


Figure 3-6. Macro As Defined

The specification of X and/or Y mirrors causes the macro to be reflected about its point of origin as in Figure 3-7. Note that the annotation is not mirrored.

The specification of an angle causes a macro, including annotation, to be rotated the specified number of degrees. Figure 3-8 illustrates the macro drawn at a rotation of 180 degrees.

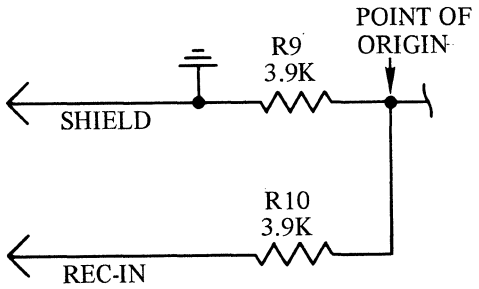


Figure 3-7. Macro Placed With X and Y Mirrors

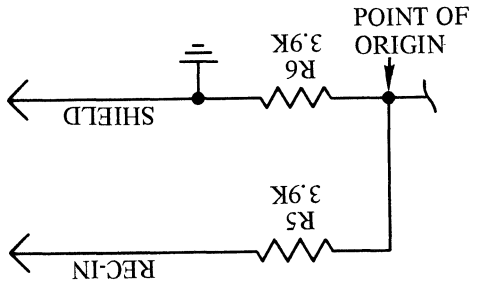


Figure 3-8. Macro Rotated at an Angle of 180 Degrees

DRAWING ORIENTATION

It may be desired to rotate and/or mirror an entire drawing. This may be done by specifying the appropriate mirror(s) and/or angle. It must be noted that mirroring a drawing does not cause the annotation on the drawing to be mirrored.

ANNOTATION

Annotation on the drawing may consist of:

- Text annotation
- Label annotation
- Symbol labeling

Text annotation (or general diagram annotation) may consist of character strings that are to be right- or left-justified about a point on the drawing.

Label annotation consists of character strings that are stacked, centered, and placed to the right, left, top, bottom, or centered about a specified point on the drawing.

Text annotation and label annotation may be rotated by specifying the desired angle.

Symbol labeling consists of character strings that are stacked, centered, and placed to the right, left, top, bottom, or center of a symbol that is placed on the drawing. It is important to note that if a symbol is drawn to the North, West, or South, or if it is mirrored,the symbol labeling is *not* rotated or mirrored. If the symbol is rotated by specifying an angle, then the associated symbol labeling is also rotated.

In all types of diagram annotation, the type and size of letters may be arbitrarily specified.

VARIABLE ANNOTATION

Variable annotation is permitted only within a macro. It is used when a macro is required in several places on a drawing and its associated annotation changes with each placement. Using variable annotation eliminates the need to create a separate macro for each placement.

Variable Annotation is digitized to signify a variable annotation string within a macro as it is being created. Then, when the macro is placed on the drawing, the actual annotation string to be used is digitized. A.S.A.P. will automatically insert the desired annotation string within the macro prior to plotting.

JUNCTION POINTS

Junction points are coordinate positions within a drawing or macro at which an interconnection may be made. This interconnection may be between symbols or between a symbol and a line segment. A.S.A.P. automatically defines junction points at the end points of line segments, the center of circles, the end points of arcs, the corners of rectangles, the point at which a macro is placed, the point about which annotation is placed, and designated node points of a symbol. These coordinate positions and other related information can be listed on the printer, if requested, by the Report Generator.

EDITING

Editing allows a symbol, macro, or drawing to be modified by making additions, deletions, or other types of alterations. Editing can be performed on the digitizer (such as windowing), by correcting the digitizer output card deck, by modifying the A.S.A.P. language describing the macro or drawing (card editing), or by replacing characters within the A.S.A.P. language with alternate characters (string editing).

A.S.A.P. LANGUAGE

The A.S.A.P. language is an English-like language used within the program to describe macros and drawings. All digitized data is transformed into this language. The particular statements derived from the digitized data are listed on the printer showing exactly how a macro or drawing is defined. Direct modification of the A.S.A.P. language describing a macro or drawing can be performed by card or string editing. The format of the language statements is described in Section 4. Non-digitized input is referred to as Hand Coding.

EXPANSION OF DRAWINGS AND MACROS

Drawings and macros when initially digitized and input to A.S.A.P. are referred to as being unexpanded. However, in order to create a plot of the drawing or macro, the drawing or macro must be expanded. Simply stated, the process of expansion is the replacement of all macro call statements (i.e., /MACRO/) with the language statements describing that macro.

Macros are retained on the Symbol File in their unexpanded format. Drawings, however, may be kept on the Symbol File in either their unexpanded or their expanded version. An expanded version of a drawing will not be saved on the Symbol File until the user determines that the plot is correct. At this time, the user can rerun the drawing with the Expand Option (EXPN) turned on and the expanded version will then replace the unexpanded version in the Symbol File. All macros used within the drawing and not required in any other drawing now may be removed from the Symbol File. Also, windows now may be used on the drawing without regard to macros, and the components within the drawing may be automatically numbered.

REPORTS AND OUTPUT LISTINGS

There are several reports and output listings generated by A.S.A.P. when a drawing is processed.

When digitizer input is read by the program, a translation listing is generated. Every digitizer access is sequenced by ten and listed with its corresponding menu or work area equivalent. When a sufficient number of accesses has been made to generate an A.S.A.P. language statement, that statement will be printed on the translation listing.

A listing of A.S.A.P. language statements is generated when any changes are made to the Symbol File. If a drawing is created, modified, or deleted, a listing of the updated drawing is generated. If a macro is created, modified, or deleted, a listing of the updated macro is generated. If a symbol is added to or deleted from the Symbol File, the change is listed.

Any error conditions encountered while processing are listed.

When macros and drawings are plotted, the plotter search address for each drawing and macro on the plot tape is listed.

At the completion of each A.S.A.P. run, the contents of the Symbol File is listed. Symbols, macros, and drawings are grouped in this list.

In addition to the above outputs, the following reports may be requested.

The X-Y Coordinate report contains:

1) the coordinates of all junction points sorted by increasing X and decreasing Y.

2) the types of A.S.A.P. Language commands that generated the coordinates.

3) the sequence numbers of the card images of the generated coordinates.

The Component report contains:

1) the names of all symbols and macros, sorted alphabetically.

2) the coordinate location of all symbols and macros.

3) the sequence numbers of the card images containing the symbol and macro names.

WINDOWING

Windowing is a method of "erasing" information from a drawing or a macro. When a window is specified, the contents of the window will be removed from the A.S.A.P. language file of the drawing or macro and consequently from the plot.

Section 4
USING A.S.A.P.

Input to A.S.A.P. is in three basic forms: (1) the A.S.A.P. System Control card, (2) the A.S.A.P. language, and (3) the digitizer input (the coordinates representing the various A.S.A.P. commands).

A.S.A.P. SYSTEM CONTROL CARD ($ASAP card)

The A.S.A.P. System Control card (referred to as the $ASAP card) is required whenever data is input to A.S.A.P. The function of this card is to inform A.S.A.P. what type of data is to be processed and what desired options are to be used. The format of the $ASAP card is:

Columns	Data
1-5	$ASAP
6	, (comma)
7-80	Type of processing and option list separated by commas and terminated by the character $

Each $ASAP card must have as its first data entry the type of processing required. The available processing types are:

DC	digitizer cards
DCC	corrections to digitizer cards followed by the digitizer cards being corrected
ALC	A.S.A.P. language cards
END	end of data input

The first three processing types cause the following options to occur: (1) the drawing being input or modified is processed and plotted (if one exists); (2) macros being input are processed but not plotted; and (3) all input is saved in the Symbol File. The last processing type must be used to end the job.

To alter the standard processing, the following options may be input with the $ASAP card:

Option	Result
ANBR	All eligible components within the drawing will be automatically numbered (optional item, see Section 11).
APTR	An aperture specification card is input immediately following the $ASAP card (has no effect unless the Painting option is available, see Section 12).
CMRP	A component report is output.

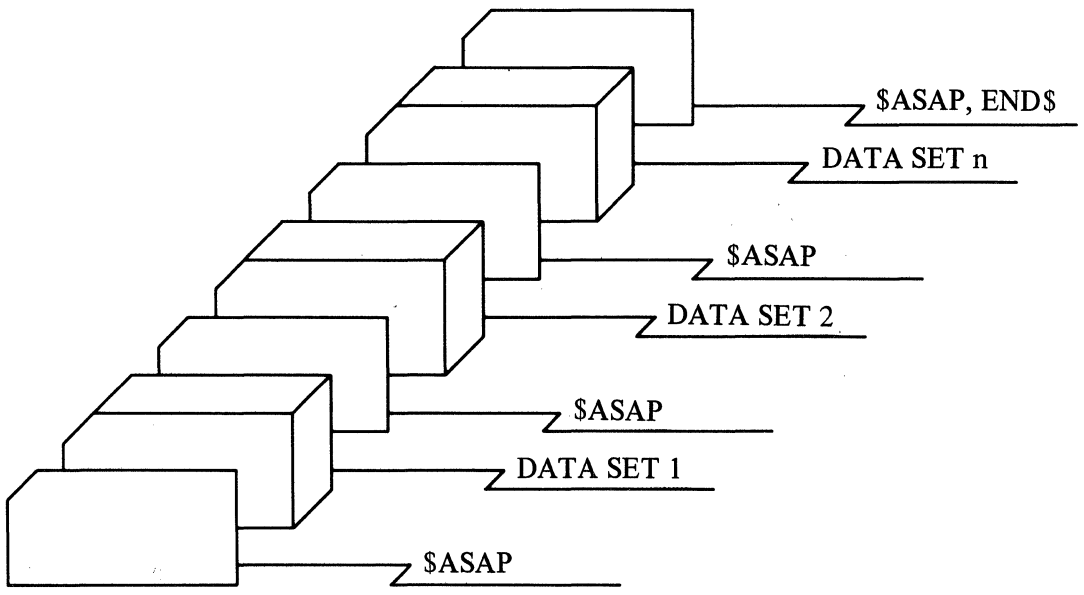
Option	Result
ICRP	An interconnect report and output file will be created (optional item, see Section 13).
ERRS	Digitized data will be processed even if it contains major errors.
EXPN	All macro calls within the drawing will be expanded into their A.S.A.P. language statements and will replace the macro call statements.
MODM	Menu modification is permitted.
NDRW	The drawing being input or modified will be edited only; does not affect macro processing.
NPLT	The drawing being input or modified will not be plotted.
PMAC	Macros input or modified will be plotted.
XYRP	A report of all junction points within the drawing sorted in increasing X–, decreasing Y is output.
INIT	Initialize a new symbol file.

NOTE: ANBR and ICRP may not be input together. Should this occur, the interconnect report will not be created (output).

Some typical $ASAP cards are shown below.

$ASAP, DC$	digitizer card input
$ASAP, DCC, ERRS, PMAC$	digitizer correction cards followed by digitizer cards – desired options are to process even with major errors and draw macros.
$ASAP, END$	Must follow last set of data input.

Each set of data input to A.S.A.P. must be preceded with a $ASAP card as shown in the following illustration:



The format of the $ASAP card requires that the type of processing be the first item on the card. Any desired options follow and can be input in any order. All items must be separated by a comma with the last item followed by a dollar sign (a comma preceding the dollar sign is optional). Any blanks are ignored and data may be continued from one card to another as in the following:

```
$ASAP, DCC,                    or      $ASAP, DCC
PMAC,                                  PMAC,
APTR, CMRP, ERRS, $                    APTR,
                                       CMRP,
                                       ERRS$
```

Should the $ASAP, END$ card be omitted, the A.S.A.P. program will issue a warning message and end the job normally as if the card were input (if possible with the computer system being used).

The following messages are issued by the program in processing the $ASAP card:

1) $ASAP CARD INPUT AS – card image input

2) INVALID $ASAP CARD INPUT AS – card image input

3) TOO MANY CHARACTERS IN OPTION WORD – word found

4) INVALID PROCESSING TYPE – word found

5) INVALID OPTION WORD – word found

6) INVALID DECK SETUP – no end card

7) NEW $ASAP CARD INPUT – previous ignored

Message 1 simply prints the $ASAP card as it was input. Messages 2-5 are error conditions. The present $ASAP card is ignored and the program searches through the input data until it finds the next $ASAP card. Message 6 has been discussed previously. Message 7 is output when a new $ASAP card is found before processing of the previous is completed (could be caused by the final dollar sign missing or a missing continuation card).

A.S.A.P. LANGUAGE

The A.S.A.P. language is an English-like language used to define drawings and macros. This language consists of code words; numeric values; drawing, symbol, and macro names; remarks; and annotation. Code words are one to four character abbreviations of the A.S.A.P. commands. A positive or negative numeric value following certain code words is represented by "nn.nnn". Positive integer values are represented by "nn". Drawing, symbol, and macro names are user specified names containing from one to eight alphanumeric characters represented by "NNNNNNNN." Remarks and annotation are user specified character strings represented by "AAA...A". The above items are separated by the A.S.A.P. punctuation listed in Table 4-1. Appropriate combinations of code words, numeric values, punctuation, etc. are grouped to form A.S.A.P. logical records. For example, an A.S.A.P. logical record contains all of the A.S.A.P. language required to specify a rectangle, circle, symbol, etc.

Table 4-1. A.S.A.P. LANGUAGE PUNCTUATION (IBM-370 VERSION)

CHARACTER	PUNCH CODE	FUNCTION
/	0-1	The slash is used to enclose Symbol and Macro names
"	7-8	The double quote is used to enclose the lines of a set of annotation.
>	0-6-8	The greater-than sign is used as a line feed-up indicator. This character separates lines of annotation and causes the line following the greater-than sign to be plotted above the preceding line.
<	12-4-8	The less-than sign is used as a line feed-down indicator. This character separates lines of annotation and causes the line following the less-than sign to be plotted below the preceding line.
,	0-3-8	The comma is used to separate code words. For example: . . .E, XM, SS=2 . , . . .
.	12-3-8	The period is used as a decimal point in numeric values.
*	11-4-8	The asterisk is used to indicate the end of an A.S.A.P. logical record, and causes that record to be processed. For example, an A.S.A.P. logical record may consist of a symbol and its associated data; i.e., . . .X=1.0000, Y=10.1250, E, XM, /QNPN/, "2N2345"* . . .
=	6-8	The equal sign establishes the code word at the left of the sign as the accepted substitute for the numerical value at the right of the sign. For example: LS=0.1250
-	11	a) The hyphen separates certain code words and their associated numeric values. For example: LTYP-3 b) The minus sign indicates a negative numeric value. For example: X=-2.5000
+	12-6-8	The plus sign indicates a positive numeric value. For example: X=+2.5000 Note: The use of the plus sign is optional.
@	4-8	The at sign indicates variable annotation.
|	12-7-8	The vertical bar begins the definition of a drawing, symbol, or macro.
—	0-5-8	The underline terminates the definition of a drawing, symbol, or macro.

A.S.A.P. COMMANDS

The A.S.A.P. commands are listed alphabetically with their definitions, digitizing procedures, and corresponding A.S.A.P. code words. The dashed boxes contain A.S.A.P. commands not available on the Control menu. These commands must be input by hand coded A.S.A.P. Language statements.

ALTER
MENU

ALTER MENU allows symbols or macros to be added to, deleted from, or moved on the temporary symbol menu area or on eligible permanent symbol menus. (See System Control Card MODM.)

- METHOD 1:

  To add a symbol or macro to a menu

  Digitizer accesses:

  1) ALTER MENU
  2) Keyboard entries (symbol or macro name)
  3) END on keyboard
  4) Symbol menu location in which symbol or macro is to reside

  A.S.A.P. statement:

  None

- METHOD 2:

  To delete a symbol or macro from a menu

  Digitizer accesses:

  1) ALTER MENU
  2) END on keyboard
  3) Symbol menu location of symbol or macro to be deleted

  A.S.A.P. statement:

  None

- METHOD 3:

  To move a symbol or macro from one location to another

  Digitizer accesses:

  1) ALTER MENU
  2) Two temporary or permanent symbol menu locations (present location followed by desired location)

  A.S.A.P. statement:

  None

ABSOLUTE

ABSOLUTE causes all subsequent placements on the work area to occur at their digitized locations instead of the nearest grid location.

Digitizer access:

ABSOLUTE

A.S.A.P. statement:

None

NOTE: To return to grid placement, see OPTION RESET or GRID SIZE.

ANGLE

ANGLE causes the next symbol, macro, or annotation to be rotated counterclockwise from the X-axis.

Digitizer accesses:

1) ANGLE (while in ANNOTATION Mode or SYMBOL OR MACRO Mode)

2) Numeric keyboard entries (angle in degrees)

3) END on keyboard

A.S.A.P. statement:

ANG=nn.nnr,

ANNOTATION

ANNOTATION allows alphanumeric characters to be placed on the drawing.

- METHOD 1:

  Text annotation (or general diagram annotation) can be placed to the right or to the left of a point on a drawing. Multiple lines of annotation may be plotted by inserting a line feed-up (LFU) or line feed-down (LFD) character at the appropriate position in the annotation character string. One line of annotation may contain a maximum of 400 characters. In text annotation placed to the right (AR), the placement location will be at the lower-left corner of the first character of the first line of annotation specified. Any subsequent lines in the annotation string will be aligned by the left-most character of the first line. This is the default annotation placement direction and is the same as accessing EAST on the direction compass while in ANNOTATION Mode.

  In text annotation placed to the left (AL), the placement location will be at the lower right-hand corner of the last character in the first line. Any subsequent lines are aligned by the rightmost character of the first line. Accessing WEST on the direction compass while in ANNOTATION Mode causes text annotation to be placed to the left.

  Digitizer accesses:

  1) ANNOTATION
  2) Placement direction
     a) WEST
     b) EAST
  3) Options
     a) LETTER SIZE
     b) ANGLE
  4) Keyboard entries (annotation character string)
  5) END on keyboard
  6) One or more work area accesses

  A.S.A.P. Statement:

  X=nn.nnn, Y=nn.nnn, (options selected) (placement direction) "AAA...A",*

- METHOD 2:

  Label annotation consists of lines of annotation that are stacked, centered, and placed about a point on a drawing. Multiple lines of annotation may be plotted by inserting a line feed-up (LFU) or line feed-down (LFD) character at the appropriate position in the annotation character string. A maximum of 14 lines may be specified and each line may contain up to 24 characters. The annotation is placed to the right of (LR), above (LT), to the left of (LL), below (LB), or centered about (LC) the point specified.

- METHOD 2 (Cont)

  Digitizer accesses:

  1) ANNOTATION
  2) Placement direction
     a) LABEL RIGHT
     b) LABEL TOP
     c) LABEL LEFT
     d) LABEL BOTTOM
     e) LABEL CENTERED

NOTE: One of the above placement directions must be specified.

  3) Options
     a) LETTER SIZE
     b) ANGLE
  4) Keyboard entries (annotation character string)
  5) END on keyboard
  6) One or more work area accesses

  A.S.A.P. statement:

  X=nn.nnn, Y=nn.nnn, (options specified) (placement direction) "AAA...A",*

ARROW

ARROW causes a line segment with an arrowhead to be drawn. The tip of the arrowhead is drawn at the second coordinate of the pair of coordinates defining the line segment. The length of the arrowhead is 0.2 inches and the width is 0.1 inches.

Digitizer accesses:

1) ARROW

2) Options

   SYMBOL SIZE (multiple of standard arrowhead size)

3) Pairs of work area accesses

A.S.A.P. statement:

(options selected) ARW,X=nn.nnn,Y=nn.nnn, X=nn.nnn, Y=nn.nnn,*

CIRCLE

CIRCLE allows a circle to be placed on the drawing.

- METHOD 1:

  Keyboard specification of radius

  Digitizer accesses:

  1) CIRCLE
  2) Numeric keyboard entries (radius of circle)
  3) END on keyboard
  4) One or more work area accesses (centers of circles)

  A.S.A.P. statement:

  CIRC=nn.nnn, X=nn.nnn, Y=nn.nnn,*

- METHOD 2:

  Work area specification of radius

  Digitizer accesses:

  1) CIRCLE
  2) Work area (center of circle)
  3) Work area (point on circumference of circle)

NOTE: Subsequent pairs of work area accesses may be specified for additional circles.

A.S.A.P. statement:

CIRC=nn.nnn, X=nn.nnn, Y=nn.nnn,*

DELETE
LINE

When an error has been made in the selection of alphanumeric characters on the keyboard, DELETE LINE allows the removal of the current line of alphanumeric characters; that is, to the last line feed character or to the beginning of the character string.

Digitizer accesses:

DELETE LINE

A.S.A.P. statement:

None

DELTA X causes the pen to move horizontally nn.nnn inches from the current pen position.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

DX=nn.nnn,

NOTE: If instruction DX is immediately followed by instruction DY, the pen moves diagonally to the point specified.

DELTA Y

DELTA Y causes the pen to move vertically nn.nnn inches from the current pen position.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

DY=nn.nnn,

DIGITIZING SCALE

DIGITIZING SCALE controls the size of the drawing or the size of a macro that is being digitized in relation to how it is to be maintained on the A.S.A.P. Symbol File. DIGITIZING SCALE applies to work area accesses only, and does not affect the size of symbols, macros, annotation, etc., that are being placed on the drawing. The program assumes an initial digitizing scale of 1.0.

For example, a digitizing scale of 2.0 indicates the drawing or macro is being digitized at twice the size it is to be kept on the Symbol file.

Digitizer accesses:

1) DIGITIZING SCALE

2) Numeric keyboard entries

3) END on keyboard

A.S.A.P. statement:

None

NOTE: DIGITIZING SCALE may be accessed only during START DRAWING, EDIT DRAWING, START MACRO, or EDIT MACRO specification.

DOWN causes the pen to be lowered during execution of the next pen-positioning instructions.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

D,

DRAWING LETTER SIZE

DRAWING LETTER SIZE sets the default height in inches of all annotation on the drawing. The initial height is 0.07 inches. The height of one annotation character string can be changed by the option LETTER SIZE.

Digitizer accesses:

1) DRAWING LETTER SIZE

2) Numeric keyboard entries (height in inches)

3) END on keyboard

A.S.A.P. statement:

DLS=nn.nnn,*

NOTE: The value established by DLS remains in effect until the next DLS instruction.

DRAWING SYMBOL SIZE

DRAWING SYMBOL SIZE sets the default size at which symbols and macros will be drawn relative to their standard size. The initial size is 1.0. The size of one symbol or macro can be changed by the option SYMBOL SIZE.

Digitizer accesses:

1) DRAWING SYMBOL SIZE

2) Numeric keyboard entries

3) END on keyboard

A.S.A.P. statement:

DSS=nn.nnn,*

NOTE: The value established by DSS remains in effect until the next DSS instruction.

DRAWING X-MIRROR

DRAWING X-MIRROR causes all subsequent plotting to be mirrored about the X-axis.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

DXM*

NOTE: This instruction remains in effect until the NO DRAWING X-MIRROR instruction is specified using the hand coded A.S.A.P. Language Statement NDXM.

DRAWING Y-MIRROR

DRAWING Y-MIRROR causes all subsequent plotting to be mirrored about the Y-axis.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

DYM*

NOTE: This instruction remains in effect until the NO DRAWING Y-MIRROR instruction is specified using the hand coded A.S.A.P. Language statement NDYM.

EAST

EAST sets the orientation of symbols, macros, and text annotation.

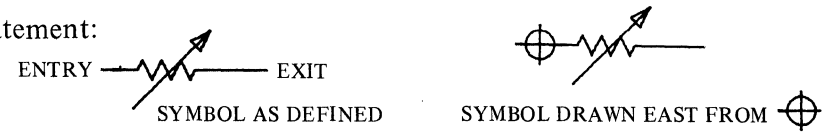
1) Symbol orientation – EAST causes a symbol to be drawn from its entry point to its exit point at a rotation of 0 degrees.

   Digitizer access:

   EAST (while in SYMBOL OR MACRO Mode)

   A.S.A.P. statement:

   E,



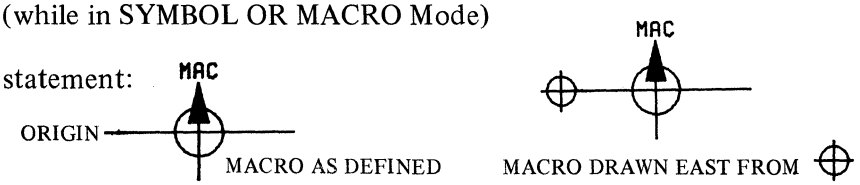
2) Macro orientation – EAST causes a macro to be drawn from its point of origin at a rotation of 0 degrees.

   Digitizer access:

   EAST (while in SYMBOL OR MACRO Mode)

   A.S.A.P. statement:

   E,



3) Text annotation orientation – EAST causes the annotation string to be drawn to the right of the point specified.

   Digitizer access:

   EAST (while in ANNOTATION Mode)

   A.S.A.P. statement:

   AR"

EDIT
DRAWING

EDIT DRAWING identifies the drawing to be edited on the digitizer, user remarks about the drawing (if desired), and the drawing's orientation on the digitizer. The name of the drawing is the 1-8 alphanumeric character name of the drawing residing on the SYMBOL FILE. The user remarks may consist of 1-250 alphanumeric characters. The drawing orientation specification consists of redigitizing the three target locations on the work area.

Digitizer accesses:

1) EDIT DRAWING

2) Alphanumeric keyboard entries (1-8 character name)

3) END on keyboard

4) Optional user remarks

   a) Alphanumeric keyboard entries (1-250 characters)

   b) END on keyboard

5) Work area (lower left target location)

6) Work area (lower right target location)

7) Work area (upper right target location)

A.S.A.P. statements:

|NNNNNNNN -1

RMK NNNNNNNN, (optional user remarks) X=nn.nnn, Y=nn,nnn,

EDIT
MACRO

EDIT MACRO identifies the macro to be edited on the digitizer and establishes its point of origin. The name of the macro is the 1-8 alphanumeric character name of the macro residing on the SYMBOL FILE. The point of origin is specified by redigitizing the original macro's point of origin.

NOTE: All subsequent digitizer accesses are relevant to the macro being edited until END MACRO is accessed.

Digitizer accesses:

1) EDIT MACRO

2) Alphanumeric keyboard entries (1-8 character name)

3) END on keyboard

4) Optional macro size specification

   a) DIGITIZING SCALE

5) Work area (macro's point of origin)

A.S.A.P. statements:

```
|NNNNNNNN -3
*RMK NNNNNNNN*
```

END PAINTING

(optional A.S.A.P. feature, see Section 12)

ERASE

ERASE causes the last digitizer access to be deleted. A maximum of 20 consecutive ERASEs are allowed.

NOTE: ERASE cannot be erased.

Digitizer access:

ERASE

A.S.A.P. statement:

None

FACTOR

FACTOR controls the size of the plot. The program assumes a FACTOR of 1.0. A FACTOR entry of 2.0 will make the plot twice as large, while a FACTOR entry of 0.5 will make the plot half as large.

Digitizer accesses:

1) FACTOR

2) Numeric keyboard entries

3) END on keyboard

A.S.A.P. statement:

FACT=nn.nnn,*

NOTE: 1) A FACT=nn.nnn statement applies only to the current plot and will not be retained on the Symbol File.

2) FACTOR controls *plot* size only (see DIGITIZING SCALE for scaling of input).

3) FACTOR may be accessed only during START DRAWING specification or EDIT DRAWING specification.

GRID
SIZE

GRID SIZE defines the resolution of digitizer placements on the work area. A GRID SIZE of one-half inch causes all work area placements to be moved to the nearest one-half inch interval relative to the drawing origin. The program assumes an initial GRID SIZE of 0.125 inch.

Digitizer accesses:

1) GRID SIZE

2) Numeric keyboard entries

3) END on keyboard

A.S.A.P. statement:

RMK GRID SIZE nn.nnn,

(optional A.S.A.P. feature, see Section 11)

(optional A.S.A.P. feature, see Section 11)

(optional A.S.A.P. feature, see Section 11)

JUNCTIONS

JUNCTIONS causes an "X" to be plotted at each subsequently generated junction point. A junction point is a coordinate on the drawing that is generated and required by the program to produce reports and make continuity connections. The JUNCTIONS option may be turned on and off as desired. Initially the JUNCTIONS option is turned off.

- METHOD 1:

  Turn JUNCTIONS on.

  Digitizer access:

  JUNCTIONS

  A.S.A.P. statement:

  PJP*

- METHOD 2:

  Turn JUNCTIONS off.

  Digitizer accesses:

  1) OPTION RESET
  2) JUNCTIONS

  A.S.A.P. Statement:

  NPJ*

LABEL
BOTTOM

LABEL BOTTOM causes Label Annotation to be plotted below the specified point, or causes Symbol Labels to be plotted below the specified symbol. In each case, the lines of annotation are stacked and centered. A space equal to one-half the size of the letters is left between the top line of annotation and the point or the bottom of the symbol.

Digitizer access:

LABEL BOTTOM (while in ANNOTATION Mode or SYMBOL OR MACRO Mode)

A.S.A.P. statement:

LB,

LABEL CENTERED

LABEL CENTERED causes Label Annotation to be plotted about the point specified, or causes Symbol Labels to be plotted about the center of the symbol specified. In each case, the lines of annotation are stacked and centered.

Digitizer access:

LABEL CENTERED (while in ANNOTATION Mode or SYMBOL OR MACRO Mode)

A.S.A.P. statement:

LC,

LABEL
LEFT

LABEL LEFT causes Label Annotation to be plotted to the left of the point specified, or causes Symbol Labels to be plotted to the left of the symbol specified. In each case, lines of annotation are stacked and centered. A space equal to one-half the size of the letters is left between the longest line of annotation and the point or the left of the symbol.

Digitizer access:

LABEL LEFT (while in ANNOTATION Mode or SYMBOL OR MACRO Mode)

A.S.A.P. statement:

LL,

LABEL
RIGHT

LABEL RIGHT causes Label Annotation to be plotted to the right of the point specified, or causes Symbol Labels to be plotted to the right of the specified symbol. In each case, the lines of annotation are stacked and centered. A space equal to one-half the size of the letters is left between the longest line of annotation and the point or the right of the symbol.

Digitizer access:

LABEL RIGHT (while in ANNOTATION Mode or SYMBOL OR MACRO Mode)

A.S.A.P. statement:

LR,

LABEL
TOP

LABEL TOP causes Label Annotation to be plotted above the point specified, or causes Symbol Labels to be plotted above the specified symbol. In each case, the lines of annotation are stacked and centered. A space equal to one-half the size of the letters is left between the bottom line of annotation and the point or the top of the symbol.

Digitizer access:

LABEL TOP (while in ANNOTATION Mode or SYMBOL OR MACRO Mode)

A.S.A.P. statement:

LT,

LETTER SET

LETTER SET controls the type of characters to be used in plotting annotation. See Table 4-2.

Digitizer accesses:

1) LETTER SET
2) Numeric keyboard entry
3) END on keyboard

A.S.A.P. statement:

LSET-n*

NOTE: Letter Set numbers 2 through 7 require more computer processing and plotting time than Letter Set number 1.

TABLE 4-2. LETTER SET TYPES

LETTER SET NUMBER	LETTERS
1	ABCDEFGHIJKLMNOPQR STUVWXYZ1234567890
2	ABCDEFGHIJKLMNØPQR STUVWXYZ1234567890
3	ABCDEFGHIJKLMNOPQR STUVWXYZ1234567890
4	abcdefghijklmnopqr stuvwxyz1234567890
5	*ABCDEFGHIJKLMNOPQR* *STUVWXYZ1234567890*
6	*ABCDEFGHIJKLMNOPQR* *STUVWXYZ1234567890*
7	*abcdefghijklmnopqr* *stuvwxyz1234567890*

LETTER SIZE

LETTER SIZE controls the height of the characters used in plotting the next annotation. If LETTER SIZE is not accessed, the annotation is plotted at a height established by DRAWING LETTER SIZE.

Digitizer accesses:

1) LETTER SIZE (while in SYMBOL OR MACRO or ANNOTATION Mode)

2) Numeric keyboard entries (height in inches)

3) END on keyboard

A.S.A.P. statement:

LS=nn.nnn,

LFD

The LFD (line feed down) character separates lines of annotation. The line of annotation following the LFD character is plotted below the line which precedes the LFD character. A space equal to one-half the size of the letters is left between the lines of annotation.

Digitizer access:

LFD (may be accessed only while specifying lines of annotation)

A.S.A.P. statement:

< (See Appendix 3 for appropriate punch code)

LFU

The LFU (line feed up) character separates lines of annotation. The line of annotation following the LFU character is plotted above the line which precedes the LFU character. A space equal to one-half the size of the letters is left between the lines of annotation.

Digitizer access:

LFU (may be accessed only while specifying lines of annotation)

A.S.A.P. statement:

> (See Appendix 3 for appropriate punch code.)

LINE

LINE causes line segments to be placed on the drawing.

- METHOD 1:

  Multiple line segment specification

  Digitizer accesses:

  1) LINE
  2) Work area (starting point of line)
  3) Work area (intermediate points defining line segments)

     .
     .
     .
     .

  n) Work area (ending point of final line segment)

  A.S.A.P. statements:

  X=nn.nnn, Y=nn.nnn,*
  D,X=nn.nnn, Y=nn.nnn,*

     .
     .
     .
     .
     .

  D,X=nn.nnn, Y=nn.nnn,*

- METHOD 2:

  Single line segment specification

  Digitizer accesses:

  1) LINE
  2) SPECIAL
  3) Work area (starting point of line)
  4) Work area (ending point of line)

NOTE: Each subsequent pair of work area accesses defines an additional line segment.

A.S.A.P. statements:

X=nn.nnn, Y=nn.nnn,*

D,X=nn.nnn, Y=nn.nnn,*

LINE TYPE

All plotting, including symbols, macros, lines, rectangles, circles, arcs, and arrows, but *not* including annotation or painting, can be controlled by LINE TYPE. If not specified, a solid line (LINE TYPE of 1) will be used. The type of lines available are shown in Table 4-3.

Digitizer accesses:

1) LINE TYPE

2) Numeric keyboard entry

3) END on keyboard

A.S.A.P. statement:

LTYP-n,*

TABLE 4-3. LINE TYPES

TYPE NUMBER	LINE
1	
2	
3	
4	
5	
6	
7	

MAJOR ARC

MAJOR ARC causes an arc whose central angle is from 180 to 360 degrees to be plotted.

- METHOD 1:

  Specification of the arc's radius and its two end points.

  Digitizer accesses:

  1) MAJOR ARC
  2) Numeric keyboard entries (radius in inches)
  3) END on keyboard
  4) Work area (beginning point of arc)
  5) Work area (end point of arc)

  NOTE: Subsequent pairs of work area accesses may be specified for additional arcs.

  IMPORTANT

  The arc will be drawn in a counterclockwise direction from the first point to the second.

  A.S.A.P. statement:

  MAJA=nn.nnn,X=nn.nnn,Y=nn.nnn,X=nn.nnn,Y=nn.nnn,*

- METHOD 2:

  Specification of three points on the arc.

  Digitizer accesses:

  1) MAJOR ARC
  2) Work area (beginning point of arc)
  3) Work area (any point on the arc other than either of the two end points)
  4) Work area (end point of arc)

  NOTE: Subsequent groups of three work area accesses may be specified for additional arcs.

  IMPORTANT

  The three work area accesses may be made either in a clockwise or counterclockwise direction, but they must not be colinear.

  A.S.A.P. statement:

  MAJA,X=nn.nnn,Y=nn.nnn,X=nn.nnn,Y=nn.nnn,X=nn.nnn,Y=nn.nnn,*

MENU

MENU allows the specification of a Symbol Menu. This specification indicates the number (from 1 to 99) of the Symbol Menu positioned on the digitizer. If MENU is not accessed, Menu Number 1 is assumed.

Digitizer accesses:

1) MENU

2) Numeric keyboard entries (menu number)

3) END on keyboard

A.S.A.P. statement:

RMK MENU nn,

MINOR ARC

MINOR ARC causes an arc whose central angle is from 0 to 180 degrees to be plotted.

- METHOD 1:

  Specification of the arc's radius and its two end points.

  Digitizer accesses:

  1) MINOR ARC
  2) Numeric keyboard entries (radius in inches)
  3) END on keyboard
  4) Work area (beginning point of arc)
  5) Work area (end point of arc)

  NOTE: Subsequent pairs of work area accesses may be specified for additional arcs.

  IMPORTANT

  The arc will be drawn in a counterclockwise direction from the first point to the second.

  A.S.A.P. statement:

  MINA=nn.nnn, X=nn.nnn, Y=nn.nnn, X=nn.nnn, Y=nn.nnn,*

- METHOD 2:

  Specification of three points on the arc.

  Digitizer accesses:

  1) MINOR ARC
  2) Work area (beginning point of arc)
  3) Work area (any point on the arc other than either of the two end points)
  4) Work area (end point of arc)

  NOTE: Subsequent groups of three work area accesses may be specified for additional arcs.

  IMPORTANT

  The three work area accesses may be made either in a clockwise or counterclockwise direction, but they must not be colinear.

  A.S.A.P. statement:

  MINA, X=nn.nnn, Y=nn.nnn, X=nn.nnn, Y=nn.nnn, X=nn.nnn, Y=nn.nnn,*

NO DRAWING X-MIRROR suppresses X-mirroring of all subsequent plotting.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

NDXM*

NOTE: This instruction remains in effect until the DRAWING X-MIRROR instruction is specified using the hand coded A.S.A.P. Language Statement DXM.

NO DRAWING Y-MIRROR suppresses Y-mirroring of all subsequent plotting.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

NDYM*

NOTE: This instruction remains in effect until the DRAWING Y-MIRROR instruction is specified using the hand coded A.S.A.P. Language Statement DYM.

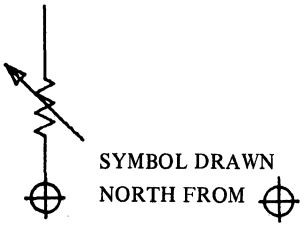
NORTH

NORTH sets the orientation of symbols and macros.

1) Symbol orientation – NORTH causes a symbol to be drawn from its entry point to its exit point at a rotation of 90 degrees.

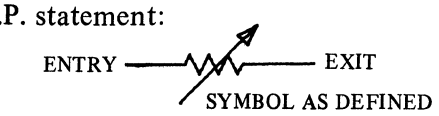
Digitizer access:

NORTH (while in SYMBOL OR MACRO mode)



A.S.A.P. statement:

N,



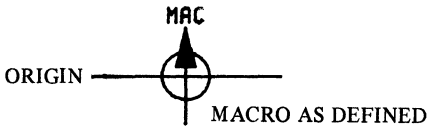
2) Macro orientation – NORTH causes a macro to be drawn from its point of origin at a rotation of 90 degrees.

Digitizer access:

NORTH (while in SYMBOL OR MACRO mode)

A.S.A.P. statement:

N,



NUMBERS (optional A.S.A.P. feature, see Section 11.)

OPTION RESET

OPTION RESET allows certain digitizing options to be reset.

I. ABSOLUTE causes all subsequent work area placements to occur at the nearest grid intersection of the last specified GRID SIZE.

Digitizer accesses:

1) OPTION RESET
2) ABSOLUTE

A.S.A.P. statement:

RMK GRID SIZE nn.nnn

II. ANGLE sets the value of ANGLE to 0 degrees.

Digitizer accesses:

1) OPTION RESET
2) ANGLE

A.S.A.P. statement:

None

III. DRAWING LETTER SIZE sets the value of DRAWING LETTER SIZE to 0.07 inches.

Digitizer accesses:

1) OPTION RESET
2) DRAWING LETTER SIZE

A.S.A.P. statement:

DLS = .070*

IV. DRAWING SYMBOL SIZE sets the value of DRAWING SYMBOL SIZE to 1.0.

Digitizer accesses:

1) OPTION RESET
2) DRAWING SYMBOL SIZE

A.S.A.P. statement:

DSS=1.00*

V. GRID SIZE sets GRID SIZE to 0.125 inch.

Digitizer accesses:

1) OPTION RESET
2) GRID SIZE

A.S.A.P. statement:

RMK GRID SIZE .125

VI. JUNCTIONS suppresses plotting of "X"s at subsequently defined junction points.

Digitizer accesses:

1) OPTION RESET
2) JUNCTIONS

A.S.A.P. statement:

NPJ*

VII. LETTER SET sets the value of LETTER SET to 1.

Digitizer accesses:

1) OPTION RESET
2) LETTER SET

A.S.A.P. statement:

LSET-1*

VIII. LETTER SIZE sets the value of LETTER SIZE to the same value as DRAWING LETTER SIZE.

Digitizer accesses:

1) OPTION RESET
2) LETTER SIZE

A.S.A.P. statement:

None

OPTION RESET (cont)

IX. LINE TYPE sets the value of LINE TYPE to 1.

Digitizer accesses:

1) OPTION RESET

2) LINE TYPE

A.S.A.P. statement:

LTYP-1*

X. NUMBERS (optional A.S.A.P. feature, see Section 11).

XI. PEN SELECT sets the value of PEN SELECT to 1.

Digitizer accesses:

1) OPTION RESET

2) PEN SELECT

A.S.A.P. statement:

PEN-1*

XII. SYMBOL SIZE sets the value of SYMBOL SIZE to the same value as DRAWING SYMBOL SIZE.

Digitizer accesses:

1) OPTION RESET

2) SYMBOL SIZE

A.S.A.P. statement:

None

XIII. X-MIRROR turns X-MIRROR off.

Digitizer accesses:

1) OPTION RESET

2) X-MIRROR

A.S.A.P. statement:

None

XIV. Y-MIRROR turns Y-MIRROR off.

Digitizer accesses:

1) OPTION RESET

2) Y-MIRROR

A.S.A.P. statement:

None

PAINT (optional A.S.A.P. feature, see Section 12)

PEN SELECT

PEN SELECT controls pen selection on multiple-pen plotters and aperture selection on photoplotters. The program assumes an initial pen value of 1.

Digitizer accesses:

1) PEN SELECT

2) Numeric keyboard entry

3) END on keyboard

A.S.A.P. statement:

PEN-n*

PIN LABELS

(Optional A.S.A.P. feature, see Section 13)

PLOT ANGLE causes all subsequent plotting to be performed at an angle measured in degrees counterclockwise from the X-axis.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

PANG=nn.nnn*

POINT

POINT allows a Junction Point to be specified on the work area.

Digitizer accesses:

1) POINT

2) Work area

NOTE: Subsequent accesses on the work area may be specified for additional Junction Points.

A.S.A.P. statement:

X=nn.nnn, Y=nn.nnn,*

RECTANGLE

RECTANGLE allows the specification of a rectangle on the drawing.

Digitizer accesses:

1) RECTANGLE
2) Work area (any corner of rectangle)
3) Work area (corner diagonally opposite first corner specified)

NOTE: Subsequent pairs of work area accesses may be specified for additional rectangles.

A.S.A.P. statement:

REC, X=nn.nnn, Y=nn.nnn, X=nn.nnn, Y=nn.nnn,*

REMARKS

REMARKS allows specification of information used to document the A.S.A.P. language defining the drawing. This documentation could include plotting instructions, references for future editing, the indication of possible digitizer mistakes, etc.

Digitizer accesses:

1) REMARKS

2) Alphanumeric keyboard entries (1-67 characters)

3) END on keyboard

A.S.A.P. statement:

RMK AAA. . . . AA

NOTE: Remarks are not plotted on the drawing.

REPEAT

REPEAT allows the next symbol or macro to be placed on the drawing a specified number of times.

- METHOD 1:

  Keyboard specification of number of repeats.

  Digitizer accesses:

  1) REPEAT (while in SYMBOL OR MACRO mode)

  2) Numeric keyboard accesses

  3) END on keyboard

  4) Work area

  A.S.A.P. statement:

  None

  NOTE: The first symbol or macro is placed at the work area location specified. Subsequent symbols or macros are placed at the terminal point of the preceding symbol or macro.

- METHOD 2:

  Work area specification of number of repeats.

  Digitizer accesses:

  1) REPEAT (while in SYMBOL OR MACRO mode)

  2) Work area (starting point of repeats, and first point used in determining the spacing between symbols and macros (X_1, Y_1))

  3) Work area (second point used in determining the spacing between symbols and macros, and the first point used in determining the number of repeats (X_2, Y_2))

  4) Work area (ending point of repeat, and is used in determining the number of repeats (X_3, Y_3))

  NOTE:

  1) The horizontal spacing between symbols or macros is the distance $X_2 - X_1$, or ΔX.

  2) The vertical spacing is the distance $Y_2 - Y_1$, or ΔY.

3) The number of columns is equal to $(X_3 - X_1)/\Delta X$.
If ΔX is 0, there will be one column.

4) The number of rows is equal to $(Y_3 - Y_1)/\Delta Y$.
If ΔY is 0, there will be one row.

IMPORTANT

When using REPEAT in ABSOLUTE Mode, erroneous ΔX and/or ΔY values may be generated.

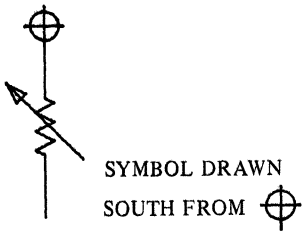
SOUTH

SOUTH sets the orientation of symbols and macros.

1) Symbol orientation – causes a symbol to be drawn from its exit point to its entry point at a rotation of 90 degrees.

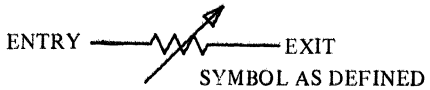
Digitizer access:

SOUTH (while in SYMBOL OR MACRO mode)



A.S.A.P. statement:

S,



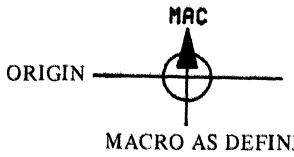
2) Macro orientation – SOUTH causes a macro to be drawn from its point of origin at a rotation of 270 degrees.

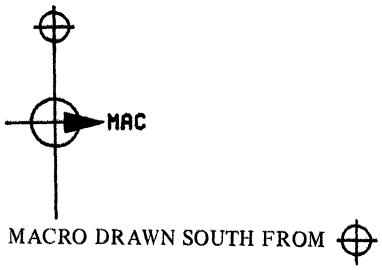
Digitizer access:

SOUTH (while in SYMBOL OR MACRO mode)

A.S.A.P. statement:

S,





SPACE

SPACE causes a blank character to be inserted in an alphanumeric character string.

Digitizer access:

SPACE (may be accessed only while making alphanumeric keyboard entries).

A.S.A.P. statement:

$\not\phi$ (blank character)

SPECIAL

SPECIAL applies only to WINDOW Mode and LINE Mode. See WINDOW or LINE.

START DRAWING

START DRAWING indicates that the specifications of a new drawing are to be entered. These specifications include the name of drawing to be digitized, optional user remarks about the drawing, optional drawing size specifications, and the drawing's orientation on the digitizer.

Digitizer accesses:

1) START DRAWING

2) Alphanumeric keyboard entries (1-8 character name)

3) END on keyboard

4) Optional user remarks

   a) Alphanumeric keyboard entries (1-250 characters)

   b) END on keyboard

5) Optional drawing size specifications

   a) FACTOR

   b) DIGITIZING SCALE

6) Work area (drawing origin, lower left corner of drawing)

NOTE: If the GRID SIZE option is to be used, this access must be on a grid intersection on the drawing.

7) Work area (lower right corner of drawing)

8) Work area (upper right corner of drawing)

A.S.A.P. statement:

```
|NNNNNNNN -1
*RMK NNNNNNNN, (optional user remarks) X=nn.nnn, Y=nn.nnn,*
(optional drawing size specifications)
X=0.0000, Y=0.0000,
/TARGET/,*
X=nn.nnn, Y=0.0000,
/TARGET/,*
```

START DRAWING (cont)

X=nn.nnn, Y=nn.nnn,

/TARGET/,*

XO=0., YO=0.*

NOTE: The above /TARGET/ statements cause the placement of a target symbol at the lower left, lower right, and upper right corners of the drawing. The XO=0., YO=0.* statement causes the targets to be placed in a separate search address.

START
MACRO

START MACRO indicates that the specifications of a new macro are to be entered. These specifications include the name of the macro to be digitized, optional macro size specification, and the macros point of origin.

NOTE: All subsequent digitizer accesses are relevant to the macro being digitized until END MACRO is accessed.

Digitizer accesses:

1) START MACRO

2) Alphanumeric keyboard entries (1-8 character name)

3) END on keyboard

4) Optional macro size specification DIGITIZING SCALE

5) Work area (macros point of origin)

A.S.A.P. statements:

|NNNNNNNN -3

RMK NNNNNNNN

SYMBOL OR MACRO

SYMBOL OR MACRO allows a symbol or macro to be selected for placement on the drawing.

- Method 1:

  Keyboard specification of symbol or macro.

  Digitizer accesses:

  1) SYMBOL OR MACRO
  2) Alphanumeric keyboard entries (1-8 character name)
  3) END on keyboard
  4) Options
     a) Symbol or macro direction
        1) EAST
        2) NORTH
        3) WEST
        4) SOUTH
     b) X-MIRROR
     c) Y-MIRROR
     d) SYMBOL SIZE
     e) LETTER SIZE
     f) ANGLE
     g) Symbol labeling
        (1) Symbol label direction (one of the following)
            (a) LABEL TOP
            (b) LABEL RIGHT
            (c) LABEL BOTTOM
            (d) LABEL LEFT
            (e) LABEL CENTERED
        (2) Alphanumeric keyboard entries (symbol label or variable annotation).
        (3) END on keyboard
     h) REPEAT

SYMBOL OR MACRO (Cont)

5) One or more work area accesses.

A.S.A.P. statements:

X=nn.nnn, Y=nn.nnn, (options selected)

/NNNNNNNN/,

(optional symbol labeling or variable annotation)*

- Method 2:

  Symbol menu specification of symbol or macro.

  Digitizer accesses:

  1) SYMBOL OR MACRO

  2) Symbol menu

  3) Options (same as Method 1).

  4) One or more work area accesses.

A.S.A.P. statements:

Same as Method 1.

SYMBOL SIZE

SYMBOL SIZE controls the size of the next symbol or macro. If SYMBOL SIZE is not accessed, the symbol or macro is plotted at the size established by DRAWING SYMBOL SIZE.

Digitizer accesses:

1) SYMBOL SIZE (while in SYMBOL OR MACRO Mode or ARROW mode)

2) Numeric keyboard entries (multiple of symbol's or macro's standard size)

3) END on keyboard

A.S.A.P. statement:

SS=nn.nnn,

TERMINATE signifies the end of an expanded macro and the end of a drawing.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

T*

USE LAST COORDINATE

USE LAST COORDINATE causes the last work area access to be reused. It is, therefore, considered a work area access and allows various drawing functions to be made continuously. For example, digitizing LINE, work area, work area, MAJOR ARC, USE LAST COORDINATE, work area, work area causes a line to be drawn and then an arc to be drawn beginning at the last coordinate of the line.

Digitizer access:

1) USE LAST COORDINATE (while in SYMBOL or MACRO, ANNOTATION, POINT, LINE, RECTANGLE, CIRCLE, MAJOR ARC, MINOR ARC, or ARROW Modes)

A.S.A.P. statement:

(The appropriate A.S.A.P. statements depending on the mode used when USE LAST COORDINATE is accessed)

VARIABLE ANNOTATION

VARIABLE ANNOTATION indicates, *within a macro*, where annotation is to be inserted at the time the macro is called.

Digitizer access:

VARIABLE ANNOTATION (must be defining a macro and be in SYMBOL OR MACRO mode or ANNOTATION mode).

A.S.A.P. statement:

"@"

WEST

WEST sets the orientation of symbols, macros, and text annotation.

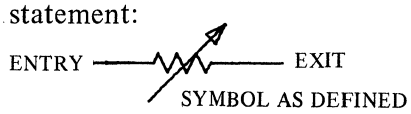
1) Symbol orientation – WEST causes a symbol to be drawn from its exit point to its entry point at a rotation of 0 degrees.

   Digitizer access:

   WEST (while in SYMBOL OR MACRO mode)

   A.S.A.P. statement:

   W,



2) Macro orientation – WEST causes a macro to be drawn from its point of origin at a rotation of 180 degrees.

   Digitizer access:

   WEST (while in SYMBOL OR MACRO mode)

   A.S.A.P. statement:

   W,

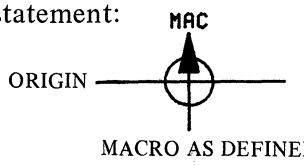
3) Text annotation orientation – WEST causes the annotation string to be drawn to the left of the point specified.

   Digitizer access:

   WEST (while in ANNOTATION mode)

   A.S.A.P. statement:

   AL"



WINDOW

WINDOW defines an area on an existing drawing or macro that is to be deleted. The appropriate A.S.A.P. language statements defining this area will be removed or modified on the Symbol File. The following items may be windowed:

1) Symbols — If the beginning and ending coordinates of a symbol are within the window, the entire symbol is deleted. If a symbol is deleted, its symbol label is also deleted. A portion of a symbol may not be deleted.

2) Macros
   a) unexpanded – The macro origin must be within the window.
   b) expanded – The portion that is within the window will be deleted.

3) Annotation — The placement coordinate of a set of lines of annotation must be within the window. A part of annotation string may not be deleted by a window.

4) Point — The point must be within the window.

5) Line — If a line intersects a window, only the portion of the line within the window will be deleted.

6) Rectangle — The entire rectangle will be deleted if any two of its corners are within a window. A portion of a rectangle may not be deleted.

7) Circle — The entire circle will be deleted if the center of the circle is within the window. A portion of a circle may not be deleted.

8) Arcs — The entire arc will be deleted if both end points are within the window. A portion of an arc may not be deleted.

9) Arrow — The entire arrow will be deleted if both end points are within the window. A portion of an arrow may not be deleted.

Painting data may not be windowed.

- METHOD 1:

  A polygon shaped window.

  Digitizer accesses:

  1) WINDOW
  2) Work area (starting point of polygon)
  3) Work area (intermediate points defining polygon)
     .
     .
     .
     .
     .
     .
     .

n) Work area (ending point of polygon; must not be same as starting point)

IMPORTANT

Each window must contain from 3 to 24 work area accesses specified in a counterclockwise direction.

A.S.A.P. statement:

WIN, X=nn.nnn, Y=nn.nnn,

X=nn.nnn, Y=nn.nnn,

.
.
.
.

X=nn.nnn, Y=nn.nnn*

- METHOD 2:

  A rectangular shaped window.

  Digitizer accesses:

  1) WINDOW
  2) SPECIAL
  3) Work area (any corner of rectangle)
  4) Work area (corner diagonally opposite first corner specified)

NOTE: Subsequent pairs of work area accesses may be specified for additional windows.

A.S.A.P. statement:

WIN, X=nn.nnn, Y=nn.nnn, X=nn.nnn, Y=nn.nnn, X=nn.nnn, Y=nn.nn, X=nn.nnn, Y=nn.nnn*

NOTE: 1) Windows may be specified only immediately following EDIT DRAWING specifications or EDIT MACRO specifications.

2) The A.S.A.P. statements for a window apply only to the current plot and are not retained on the Symbol File.

WINDOW OFF terminates previously defined windows.

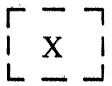
Digitizer access:

Any access, except ERASE, immediately following the last work area access defining a window.

A.S.A.P. statement:

WOFF-nn,*

NOTE: The WOFF-nn,* statement applies only to the current plot and is not retained on the Symbol File.



X causes the pen to move from its current position to the point nn.nnn inches parallel to the X-axis from the drawing origin or from the current macro origin.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

X=nn.nnn,

NOTE: If the instruction X=nn.nnn, is followed immediately by the instruction Y=nn.nnn, the pen moves directly to the point specified by X=nn.nnn, Y=nn.nnn,.

X-MIRROR

X-MIRROR causes the next symbol or macro to be mirrored about the X-axis from its point of origin.

Digitizer accesses:

X-MIRROR (while in SYMBOL OR MACRO mode)

A.S.A.P. statement:

XM,

NOTE: See Symbol and Macro Orientation, Section 3.

X-ORIGIN

X-ORIGIN: a) causes the pen to move parallel to the X-axis to a point nn.nnn inches from the current drawing origin; b) establishes a new drawing origin at the new pen position; and c) causes the next sequential search address to be written on the plot tape.

Digitizer accesses:

Not applicable

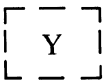
A.S.A.P. statement:

XO=nn.nnn,

NOTE: If the instruction XO=nn.nnn, is followed immediately by the instruction YO=nn.nnn, the pen moves directly to the point specified by XO=nn.nnn, YO=nn.nnn,.

CAUTION

All subsequent plotting is relative to the *new* drawing origin.



Y causes the pen to move from its current position to the point nn.nnn inches parallel to the Y-axis from the drawing origin or from the current macro origin.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

Y=nn.nnn,

Y-MIRROR

Y-MIRROR causes the next symbol or macro to be mirrored about the Y-axis from its point of origin.

Digitizer accesses:

Y-MIRROR (while in SYMBOL OR MACRO Mode)

A.S.A.P. statement:

YM,

NOTE: See Symbol and Macro Orientation, Section 3.

Y-ORIGIN: a) causes the pen to move parallel to the Y-axis to a point nn.nnn inches from the current drawing origin; b) establishes a new drawing origin at the new pen position; and c) causes the next sequential search address to be written on the plot tape.

Digitizer accesses:

Not applicable

A.S.A.P. statement:

YO=nn.nnn,

CAUTION

All subsequent plotting is relative to the new drawing origin.

Section 5
EDITING

There are four unique methods available by which drawings may be modified or edited. Additions, deletions, or other modifications to drawings and macros can be performed by: (1) Windowing (defining an area of a drawing on the digitizer which encloses items to be removed); (2) digitizer corrections (preparing punched cards to correct a digitized data set); (3) card editing (preparing hand-coded A.S.A.P. language statements to modify a drawing in the Symbol File); and (4) string editing (replacing character strings within the A.S.A.P. language with alternate character strings).

The remainder of this Section will discuss how to perform digitizer corrections (which are processed by the Translator), card editing, and string editing (both processed by the Editor). Windowing is described in Section 4.

DIGITIZER CORRECTION CARDS

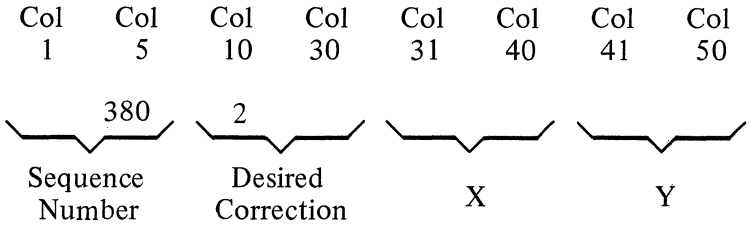
The Translator converts (or translates) digitizer coordinates into A.S.A.P. language statements. Translator input consists of: (1) cards containing coordinate information produced by the digitizer; and (2), if required, user-produced editing cards for modifying the digitized data set.

As the Translator reads each coordinate produced by the digitizer, a sequence number is assigned to that coordinate. Then, as each of the coordinates is processed, its value and assigned sequence number are listed on the printer. When an appropriate number of coordinate accesses have been processed, the Translator produces the translated A.S.A.P. language statement(s).

The Translator is designed to facilitate corrections of digitizer-generated errors that could render all or part of the drawing unusable. For example, an incorrect menu number selection could cause an entire drawing to be made up of unwanted symbols. To accommodate this type of error, the Translator is designed to read editing type cards. These cards contain the desired corrective action and a sequence number corresponding to the originally assigned sequence number generated by the Translator. With these sequence numbers, the Translator can then insert, replace, or delete a data line depending upon the corrective action required. In the above example of the incorrect menu number, assume that the generated data lines were as follows:

	Translator-assigned sequence number	Translation of user-accessed coordinates
	370	MENU
	380	1
RMK MENU 1	390	END on keyboard
	400	GRID SIZE
A.S.A.P. language statement generated by Translator	410	.
	420	1

Here, the user accessed MENU, the Number 1, and then keyboard END (-). If the user had intended to access menu Number 2, a simple digitizer correction card (shown in the card illustration below) containing the sequence Number 380 and a 2 could be edited into the digitized data set by the Translator.



Digitizer Correction Card

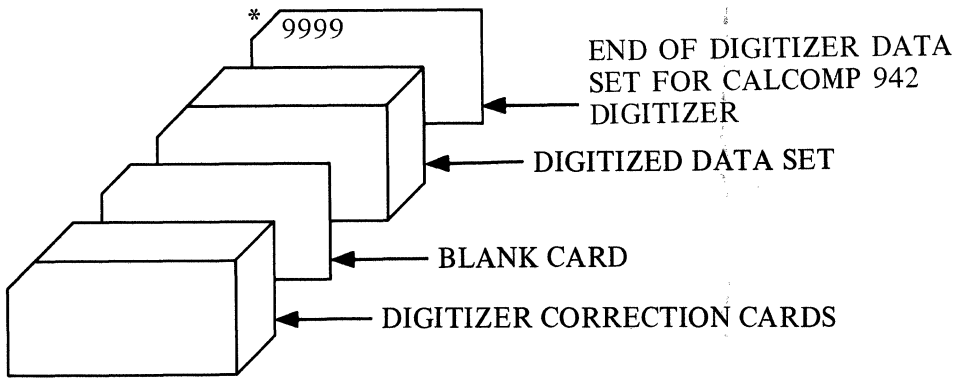
The format of the digitizer correction card is:

Columns	Data
1-5	Translator sequence number (right-justified)
10-30	Desired correction (left-justified)
31-40	X-coordinate (floating point)
41-50	Y-coordinate (floating point)

The desired correction field must contain the corrective action desired, left-justified, exactly as that particular action appears on the Control Menu (i.e., START MACRO, ERASE, 2, etc.). However, if the correction involves an area within the work area or Symbol Menu then this field must contain WORK AREA or SYMBOL MENU. When either of these is used, the X and Y coordinates (in floating point format -nn.n) must be input. Work area accesses are relative to drawing or macro origins and X and Y coordinates of the Symbol Menu accesses are relative to the lower left-hand corner of the Symbol Menu.

To delete a particular access via digitizer correction cards, the sequence number of the data line to be removed must be input with the action ERASE. Insertion of an access requires an input sequence number between those listed.

Once the desired corrections have been made, this deck of cards, followed by a blank card, must be placed in front of the digitized data set, and the Translator must be reexecuted.



*NOTE: End of Digitizer Data Set Card may vary depending on the digitizer being used.

Note that the drawing or macro being corrected must be deleted via Editor Card editing, prior to reexecuting the Translator or an error will occur (see DRAWING AND MACRO EDITING).

A.S.A.P. LANGUAGE EDITING

The primary function of the Editor is to maintain the Symbol File which consists of: (1) A.S.A.P. language describing drawings and macros, and (2) symbol definitions and associated menus. The A.S.A.P. language is kept as a 72-column card image plus a generated sequence number in order to facilitate manipulation by the user. Symbols are kept intact as a string of codes and data.

Input to the Editor may come: (1) from the Translator in the form of A.S.A.P. statements translated from digitized data created by the user, (2) in the form of A.S.A.P. statements generated by the Drawing Processor and (3) from hand-coded A.S.A.P. statements or symbol definitions created by the user.

The above described input is used to introduce a new drawing, a macro, or a symbol to the system; to add or delete A.S.A.P. statements within an existing drawing or macro; or to replace an existing symbol. Note that symbol data can not be modified, it must be replaced in its entirety.

In addition to the above capabilities, macros and drawings may be modified via the string-editing feature. String editing allows the user to replace a specified sequence of alphanumeric characters with a new sequence of characters. This feature is especially useful when large-volume annotation is required, as the annotation strings may be prepared on a keypunch and then added to the drawing by the Editor, thus reducing digitizer strokes.

BEGIN AND END EDIT CARDS

Input to the Editor varies in format depending upon its intended purpose (e.g., string editing, symbol definition, etc.). In all cases, however, the input cards pertaining to a particular drawing, macro, or symbol must be preceded by the Begin Edit card and succeeded by the End Edit card. The format for these two cards is as follows:

Begin Edit Card	
Column 1	Begin Edit character
Columns 2-9	Alphanumeric name of item desired – may be punched anywhere in field, as blanks are ignored. Maximum of eight characters.
Column 10	Blank
Columns 11-12	Type code – numeric
	$\geqslant 0$ symbol – number and count of pin labels in the symbol.
	-1 drawing – either new or modifying an existing one
	-3 macro – either new or modifying an existing one
	-4 delete item from the Symbol File
	No other codes are allowed.
Columns 13-20	Blank

*Columns 21-30	X-dimension of a new drawing (−1 code) in units of 0.0001 inch right justified with no decimal point punched.
*Columns 31-40	Same as Columns 21-30, except field contains the Y-dimension.
Columns 41-80	Blank
End Edit Card	
Column 1	End Edit character
Columns 2-80	Blank

*Used only when inputing a new drawing with hand-coded A.S.A.P. language statements.

DRAWING AND MACRO EDITING

Drawings and macros may be: (1) initially placed on the Symbol File, (2) deleted in their entirety from the file, or (3) modified via card or string editing.

Initial placement of the drawing or macro onto the Symbol File occurs when the name on the Begin Edit card (Columns 2-9) is not an existing name in the file. These input cards may come from translated digitizer data or from hand-coded data. Once the Editor has ascertained that a legal entry can be made, all of the cards following the Begin Edit card up to the End Edit card are placed on the Symbol File, thus creating the definition of that macro or drawing. Note that the Editor does not check for valid A.S.A.P. language statements as this is the function of the Drawing Processor.

Deletion of a macro or drawing already existing on the Symbol File occurs when the name in Columns 2-9 of the Begin Edit card is found in the file and the code in Columns 11-12 is -4. The Begin Edit card must be immediately followed by the End Edit card – any cards in between are ignored.

Modification of an existing macro or drawing is accomplished via card or string editing. In either type of modification, the name in Columns 2-9 of the Begin Edit card must already be in the file and the type code in Columns 11-12 must match the type code in the file. If the type code does not match, or the name can not be found, editing is not performed.

CARD EDITING

Card editing consists of adding, deleting, or replacing card images containing A.S.A.P. statements and is accomplished by matching the input card sequence numbers with the sequence numbers of the card images on the Symbol File. The card images on the Symbol File are sequenced by 100, starting at 100, and are printed when the drawing or macro is placed in the file.

Replacing an existing card image simply requires an input card with the same sequence number punched in Columns 73-80 (right justified). Upon finding the match, the Editor will remove the existing card image and replace it with the new one.

Deleting card images may be done in two ways: (1) by deleting a single card image; or (2) by deleting a sequence of card images in a from-to set. The first method requires a blank card with the correct sequence number as input (which simply replaces the existing card image with an all-blank card image, thus deleting the existing card image since blanks

are ignored). The second method requires a pair of input cards, the first of which has a negative sequence number (equal to the first existing card image to be deleted), and the second, which has the sequence number (positive) equal to the last card image to be deleted. Thus, if existing card images 500-900 are to be deleted, the input from-to pair sequence numbers would be -500 and 900. Note that to delete only the existing card image 500, either a blank card, whose sequence number is 500, or a from-to pair of cards, whose sequence numbers are -500 and 500, may be input.

Adding card images to an existing drawing or macro is accomplished by inputting new cards with a sequence number between the sequence numbers of the existing card images on either side of the insertion point. There is no limit to the number of card images which may be inserted at any one point. After the initial input card, any others to be added at that point may have the same sequence number or a blank sequence number and they will be inserted in the same order as input. For example, to insert a card image between 500 and 510, use the sequence number 501; to insert five card images between 500 and 510, use the sequence No. 501 followed by four blank sequence numbers.

STRING EDITING

String editing is used to replace a character or string of characters on an existing card image with another string of characters. The string to be replaced is called the match string and it is replaced by the replacement string. The match string must be less than or equal to 72 characters, but it may be split over two existing card images (i.e., Column 1 of the second card image is considered to be adjacent to Column 72 of the first card image). The replacement string has no maximum length. A maximum of 100 strings is allowed.

String editing may be used with or without Card Editing. If it is combined with Card Editing, the String Edit cards must be input before the Card Edits. If string editing is used alone, the String Edit cards must be bounded by the Begin Edit and End Edit cards. Note also that string editing is performed only on the card images already existing on the Symbol File and cannot be used on any of the card edits.

The following special characters are used for string editing, and must not be used within the replacement string.

begin string edit	#
end string edit	$
first time replacement	?
every time replacement	!
match any character	¬
match blanks	¢

The format of the card input containing the A.S.A.P. statement is: Columns 1 thru 72, the A.S.A.P. statement; Columns 73 through 80, a right-justified sequence number.

In addition, the replacement string must not contain the Begin Edit character | or the End Edit character ___.

The format of the string edit input is as follows:

Column 1	Begin string edit character #
Columns 2, n	Match string ($\leqslant$ 72 characters)
Columns n+1	Replacement type character ? or !
Columns n+2, m	Replacement string
Columns m+1	End string edit character $

In defining the match string, two special characters may be used to simplify its specification. The first character is the "match any" ($\rightharpoonup$). If the match string is specified to be A $\rightharpoonup$ C, then any character strings such as ABC, A C, ACC, A1C, etc. encountered within the macro or drawing will be matched because the string editor will match any character between A and C. The second character is the "match blank" ($\not\subset$). Should the match string be specified as A $\not\subset$ C, then any character string with one or more blanks separating A and C such as A C, A $\not\subset$ C, etc. will be matched. Note that the existing string AC does not match in either case.

Replacement strings are defined as first time occurrence only (?), or every time occurrence (!). That is, only the first match string will be replaced or every time the match string is found, it will be replaced. Since, however, replacement strings are processed in the order of their input, the same match string can be used with multiple first-time replacements. For example, "replace first-time ABC with DEF", "replace first-time ABC with GHI", and "replace first-time ABC with JKL" would result in the first ABC match string to be replaced by DEF, the next ABC would become GHI, and the next ABC matched would become JKL.

The replacement string may be longer or shorter than the match string. It may even be a null string (no characters) which results in the match string being deleted from the drawing or macro.

SYMBOL EDITING

Symbols are placed in the Symbol File in a different manner than macros or drawings. They reside in the file as numeric values rather than A.S.A.P. statements. In fact, the entire symbol is really one long statement. As such, it may be added, deleted, or replaced in the file only in its entirety. Changes to a symbol require updating its card deck and replacing it in the Symbol File. The data required to create a symbol is described in Section 8.

Section 6
PROCESSING

A.S.A.P. PROCESSING

From the digitized data describing a drawing and/or macro, the Translator transforms the numeric data into A.S.A.P. language statements. These statements represent exactly what was contained in the digitized data and are placed into the Symbol File by the Editor.

As the Editor initially processes each macro and the drawing, each statement received from the Translator is assigned a sequence number and is either initially placed into the file or merged with an existing entry in the file (i.e., editing a macro or drawing). The Editor maintains a list of all macro names and the drawing name input, and also prints the contents of each as it is placed in the file. The listing contains the A.S.A.P. language statements, their newly assigned sequence numbers, and the sequence number of each statement as it previously existed on the file (equal to 0 for a new entry).

After the macros and the drawing have been initially edited and placed in the file, the Editor starts the final processing for each entry just made. This includes expansion, plot creation, and final editing.

Each macro edited is obtained from the Symbol File and passed to the Drawing Processor. Here the macro is expanded and a plot is optionally created. Concurrently, any windows and continuity connections are resolved with A.S.A.P. language statements being automatically created. These statements (if any) are merged into the macro during final editing of the macro which replaces the initial version on the Symbol File. A listing of this edit is printed which, in content, is the same as described previously.

After final editing of the macros, the drawing is optionally plotted. If plotted, the final edited drawing is placed in the Symbol File. The drawing may be either the expanded version, or the version still containing macro calls and variable annotation characters.

It should be noted that this process is followed whether the input is obtained from the Translator or from hand-coded data.

PLOTTING A DRAWING

Each time a drawing is edited, whether by digitized input or hand-coded input, that drawing will be plotted unless the NDRW (NO DRAW) option has been specified on the $ASAP card. If an existing drawing is to be plotted, it must be accessed by a Begin Edit card, Type -1, with the drawing name in Columns 2 through 9 followed by an End Edit card.

DRAWING AND MACRO EXPANSION

Once a drawing and its associated macros have been digitized and input to A.S.A.P., it is the function of the program to generate the necessary data to plot the drawing. This data is created by expanding all of the macros used within the drawing and replacing any variable annotation characters with their related annotation strings.

Macro expansion is defined as the replacement of the macro call statement (i.e., /MACRO/) with the A.S.A.P. language statements contained within the macro itself. The macro call statement is replaced as /MACRO* followed by the language statements within the macro, and finally, the termination statement T* upon reaching the end of the macro.

Macros may be created in such a manner that one macro may call upon another which, in turn, may call upon another, etc. This is called macro nesting and may be done for a maximum of five levels. The highest level macro call is referred to as an outermost macro and the lowest level macro call within a nest is referred to as an innermost macro call.

During expansion of a drawing or of a nested macro, variable annotation characters (@) are replaced with annotation substitutions. These substitutions are obtained from the macro call statement and replace the variable annotation characters in the same sequence as they were input with the call statement. In addition, the substitutions are obtained starting with an innermost macro proceeding toward the outermost macro. Thus, a substitution could be obtained from the next higher level macro (i.e., 2 being higher than 3) or from the outermost macro. Also, when no more substitutions are available from the next higher level, one will be sought proceeding toward the outermost macro. An example of substitution follows:

Definition of macro MAC1:

```
/R/ "@", "ANNOTATION", "@",*

/MAC2/, "V3", "@", "V5",*

/CC/, "@"*
```

Definition of macro MAC2:

```
/QPNP/, "@", "@", "@", "@",*
```

Macro call statement:

```
/MAC1/, "V1", "V2", "V4", "V6"*
```

Resulting source language statements:

```
/MAC1*

/R/, "V1", "ANNOTATION", "V2",*

/MAC2*

/QPNP/, "V3", "V4", "V5", "V6",*

T*

/CC/," ", *

T*
```

The resulting A.S.A.P. language statements were created in the following manner:

/MAC1/, "V1", "V2", "V4", "V6",* causes

/MAC1*

/R/, "V1", "ANNOTATION", "V2",*

/MAC2/, "V3", "@", "V5",* causes

/MAC2*

/QPNP/, "V3", "V4", "V5", "V6",*

Note that the substitution "V4" is obtained from the outer macro by being passed through the variable annotation character given with the /MAC2/ call statement.

The substitution "V6" is also obtained from the outer macro because the inner macro has run out of substitutions.

terminating macro MAC2 causes

T*

returning to macro MAC1 causes

/CC/, " ",*

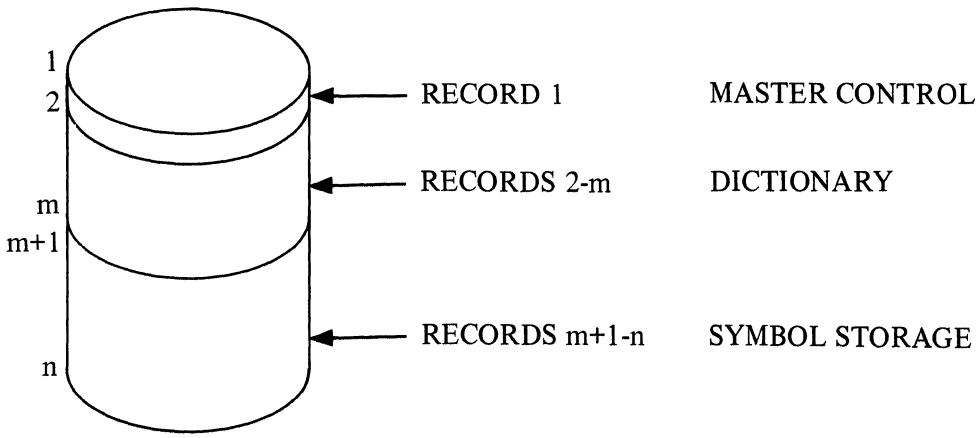
The variable annotation character on this statement is replaced by a blank because neither macro has a substitution left – a warning message is printed when this condition is encountered.

and, terminating macro MAC1 causes

T*

Section 7
SYMBOL FILE

The Symbol File is the data storage media for A.S.A.P. It contains symbol definitions, A.S.A.P. language statements describing macros and drawings, and a table of contents called the dictionary. In order to facilitate rapid storage and retrieval of the data stored in the file, a random access device is used to contain the file which is structured as shown below.



GENERAL DESCRIPTION

Each record contained in the file is 512 words long, and there may be as many records as desired (n) up to the capacity of the random access device. The number of records is normally set to 256, thus providing 131,072 words of storage for the entire file. The master control record uses 512 words, the dictionary (normally set to 8 records) uses 4096 words, leaving 126,464 words for symbol and A.S.A.P. language statement storage.

Using the above numbers as the size of the file, the standard seventy (70) A.S.A.P. symbols use two percent (2%) of the entire symbol storage area. Drawings and macros use various percentages of the file. An accounting of space usage is given after every A.S.A.P. run as shown in the following illustration.

```
SYMBOL FILE IS 8 PERCENT FULL
CONTENTS OF SYMBOL FILE – SYMBOLS
TARGET    ZD     WJ     VO     VI     TVMC   TV     TMC    T      TR
SNL       SMO    SMC    SLT    SL     SG     SD     RV     RCT    R
QFUN      QFP    QFN    PU     PMF    PM     PF     PC     OAMP   LS
GF        GE     GC     GA     FB     FA     F      EO     EC     DS
CRT       CR     CO     CCIR   CC     CB     CAC    CA     C      BH
                        – MACROS
CBR       CSE    SRNC   SRNO   TCDEC  TCDEC  TCEC   TCEO   THERM  TH
                        – DRAWINGS
PMG2      GJH1   HN1    A S A P 1
DICTIONARY CONTAINS 160 ENTRIES (MAXIMUM IS 1024)
```

It should be noted that all operations on the Symbol File during A.S.A.P. processing is performed on a copy of the file. Prior to processing, A.S.A.P. uses the Subroutine SYMIN to copy the permanent file onto a work file, and, when processing is complete, the Subroutine SYMOUT copies the work file back onto the permanent file. This technique helps to insure that the permanent file remains intact even if some type of system error causes A.S.A.P. to fail and possibly destroy the work copy.

During processing, certain error conditions will cause A.S.A.P. to print the contents of the work file and cease processing. In the event this should happen, the work file is not copied back onto the permanent file. Thus, the permanent Symbol File is exactly as it was prior to the run having the failure.

Section 8
SYMBOL CREATION

Symbols are the basic means of creating macros and drawings. Symbols themselves are not, however, created in the same fashion as macros and drawings. They are input to A.S.A.P. in the form of alphanumeric values on punched cards. These values are obtained from a scaled drawing of the desired symbol. This symbol description is placed into the Symbol File exactly as input, ready for use within macros and drawings. The following discussion explains how to create symbols.

Three data types are permitted in the description of a symbol: numeric, component designator (alphanumeric), and associated alphanumeric annotation. Numeric data is used to describe the shape of the symbol; alphanumeric component designator data is used to define the component class of the symbol which is required for automatic numbering of the components within a drawing (optional item); and alphanumeric annotation data is used for any annotation which is an integral part of the symbol.

Numeric data consists of an optional sign (+ or −) and an integer sequence of digits or a floating point number used to represent a linear dimension in units of 0.0001 inch or an angular dimension in units of 0.0001 degree. Thus, a value of 10000 or 1. represents 1 inch or 1 degree.

The alphanumeric component designator is composed of a character count (maximum of 4) followed by the letter P and then the character string itself.

Associated alphanumeric annotation consists of a character count followed by the letter H and then the character string. There is no limit to the number of characters allowed and all characters including blanks will be plotted.

When the above data types are punched onto cards, they are separated from one another by one of two delimiter characters: the slash (/) or the comma (,). The difference between the two delimiters is whether the next data entry is punched in the column following the delimiter (intervening blanks are ignored), in which case, the comma is used, or starts in Column 1 of the next card, which requires using the slash. Any data punched after the slash is ignored and may be used for comments.

If no slash is used in a card, Column 72 of that card is considered as the immediate predecessor of Column 1 of the next card. Thus a data entry may be split over the two cards.

Note that two adjacent delimiters (,, or ,/) will generate a null entry that is equivalent to zero. Care should be taken that this does not occur, as it will cause erroneous data within the symbol definition. The combination /, will not generate a zero, as the comma is considered to be a comment.

The format for describing symbols is shown in the following paragraphs. These cards must be bound by the Begin Edit and End Edit cards (see Section 5).

The first data entry used in describing a symbol is the component designator used in Automatic Numbering, Section 11. The component designator is input as zero if the symbol is not eligible for automatic component numbering (even if this option is not available). Otherwise, it is input as a character count, the letter P, and the component class (string of characters from one to four). A maximum of 72 component classes (i.e., R-Resistors, Q-Transistors, etc.) may be used.

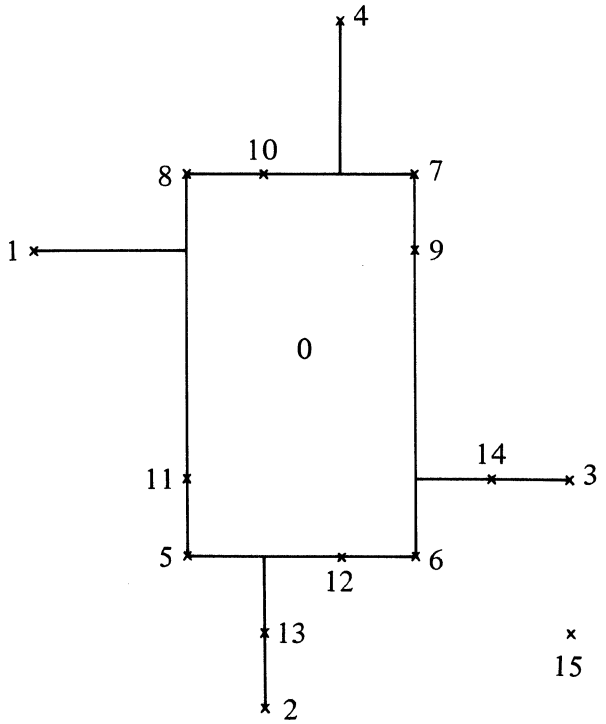
The second and third data entries must be the X and Y coordinates, respectively, of the exit node of the symbol in relation to its origin point (0, 0). Normally, the entry node of the symbol is placed at the origin, so that the exit node position is defined relating to the entry node.

Entry and exit nodes, and any other nodes (points at which an interconnection may be made) can be designated as junction points. Junction points are listed on the printer by the report generator, if requested. Note that symbols do not require junction points, but that any symbol not having at least one cannot be automatically numbered or interconnected using the Continuity Connection command.

The remainder of entries following the component designator and exit node position consist of a numeric operation code followed by related data or zero. These entries are referred to as instructions.

For all instructions except REPEAT, CIRCLE, and ANNOTATION, a negative operation code indicates that the final position of the pen, after executing the instruction, will become a junction point. If a negative operation code is used, it must be followed with a numeric connection designator. This designator is used to indicate the allowable directions from which this junction point may be connected. The connection designators are:

0	no direction
1	north, west, south
2	west, south, east
3	south, east, north
4	east, north, west
5	west, south
6	south, east
7	east, north
8	north, west
9	east
10	north
11	west
12	south
13	east, west
14	north, south
15	any direction



The number of junction points defined must agree with the value punched in Columns 9 and 10 of the Begin Edit card.

Instructions used in the definition of symbols are described in the following paragraphs. Entries representing linear measurements must be made in units of 0.0001 inches (thus, 10000 or 1. equals 1 inch), and angular measurement in units of 0.0001 degrees (10000 or 1. equals 1 degree). All entries must be separated by a delimiter and all instructions are punched sequentially on cards.

Instruction	Entries
PEN DOWN, ABSOLUTE	1. 2 or -2, connection designator
	2. X and Y coordinates relative to symbol origin of terminal point
PEN UP, ABSOLUTE	1. 3 or -3, connection designator
	2. X and Y coordinates relative to symbol origin of terminal point
PEN DOWN, DELTA	1. 4 or -4, connection designator
	2. X and Y coordinates relative to current pen position of terminal point
PEN UP, DELTA	1. 5 or -5, connection designator
	2. X and Y coordinates relative to current pen position of terminal point
CIRCLE	1. 6
	2. Radius of circle to be drawn about current pen position
MINOR ARC	1. 7 or -7, connection designator
	2. Radius of arc
	3. Terminal point of arc relative to current pen position

The arc will be drawn counterclockwise from the current pen position to the terminal point.

Instruction	Entries
MAJOR ARC	1. 8 or -8, connection designator
	2. Radius of arc
	3. Terminal point of arc relative to current pen position

The arc will be drawn counterclockwise from the current pen position to the terminal point.

Instruction	Entries
REPEAT	1. 9 2. Repeat count

The sequence of instructions, following the repeat instruction, up to a matching REPEAT TERMINATOR instruction is performed the number of times specified in Entry Number 2. A negative or zero repeat count is equivalent to a count of 1. Repeating sequences may be nested within larger repeating sequences; the one requirement being that the interior repeating sequence be completely contained within the outer sequence.

Instruction	Entries
REPEAT TERMINATOR	1. 10

This operation code signals the end of a sequence of instructions to be repeated. REPEAT TERMINATORS are paired with REPEAT instructions in a nested manner; that is, if, in a sequence of instructions, two REPEAT instructions are encountered before a REPEAT TERMINATOR is encountered, the first REPEAT TERMINATOR encountered will be paired with the second REPEAT, and the second REPEAT TERMINATOR encountered will be paired with the first REPEAT.

Instruction	Entries
OPTIONAL SYMBOL LABELING PLACEMENT	1. 11 or -11, connection designator 2. X and Y coordinates relative to symbol origin

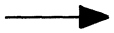
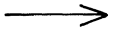
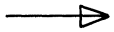
If this instruction appears within a symbol deck, the specified coordinates will override any calculated position for placement of symbol labeling. The symbol labeling will be centered horizontally and vertically about the point.

Instruction	Entries
ANNOTATION	1. 12 2. Annotation height 3. Angle at which annotation is to be drawn 4. Alphanumeric string to be drawn

The annotation is drawn centered about the current pen position.

Instruction	Entries
SOLDER DOT	1. 13

A solder dot is drawn at the current pen position.

Instruction	Entries
ARROW	1. 14 or -14, connection designator 2. Type of arrow Type Number — Arrow 1  2  3  4  5  3. Height of arrowhead 4. Length of arrowhead 5. X and Y coordinates relative to current pen position of tip of arrowhead.
VARIABLE PIN LABEL DESIGNATION (optional feature, see Section 13)	1. 15 2. Pin Label Annotation Placement Designator. This designator indicates the direction Pin Label Annotation is to be placed relative to the junction point defining the pin label. Designator — Direction 1 East 2 Northeast 3 North 4 Northwest 5 West 6 Southwest 7 South 8 Southeast 3. Pin Label Annotation Orientation Code. 0 – horizontal 1 – vertical 4. Pin Label Annotation Plotting Indicator. -1 – never plot pin label 0 – plot pin label only if A.S.A.P. statement PPL is in effect 1 – always plot pin label

FIXED PIN LABEL DESIGNATION
(optional feature, see Section 13)

1. 16
2. Pin Label Annotation Placement Designator. This designator indicates the direction Pin Label Annotation is to be placed relative to the junction point defining the pin label.

Designator	Direction
1	East
2	Northeast
3	North
4	Northwest
5	West
6	Southwest
7	South
8	Southeast

3. Pin Label Annotation Orientation Code.
   0 – horizontal
   1 – vertical
4. Pin Label Annotation Plotting Indicator.
   -1 – never plot pin label
   0 – plot pin label only if A.S.A.P. statement PPL is in effect
   1 – always plot pin label
5. Pin Label Characters.
   2HAA (AA represents the pin label characters)

EXAMPLE: The instructions to create a symbol called OAMP with seven junction points, associated annotation, and a member of the component class A is shown in Figure 8-1. The symbol is drawn on a one-quarter inch grid.

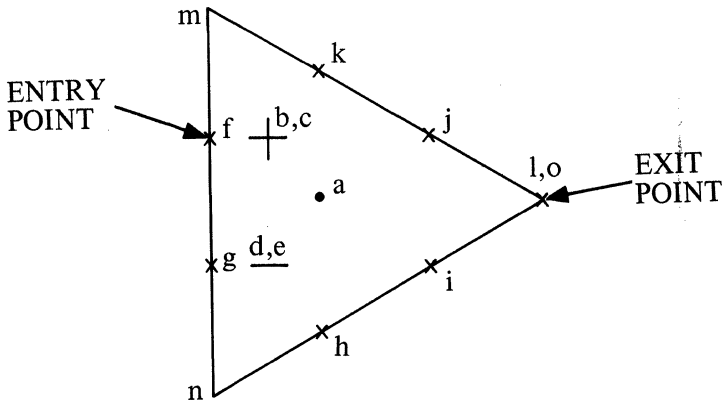


Figure 8-1. OAMP

EXAMPLE OF OAMP INPUT

	|OAMP		7			Begin Edit card
	1 PA,	15000,	-2500		/	component designator and relative position of exit node
a.	11,	5000,	-2500		/	optional labelling postion
b.	3,	2000,	0		/	pen up – absolute
c.	12,	2000,	0,	1 H+	/	associated annotation
d.	3,	2000,	-5000		/	pen up – absolute
e.	12,	2000,	0,	1 H-	/	associated annotation
f.	-3,	11,	0,	0	/	junction point – west
g.	-3,	11,	0,	-5000	/	junction point – west
h.	-3,	6,	5000,	-7500	/	junction point – south, east
i.	-3,	6,	10000,	-5000	/	junction point – south, east
j.	-3,	7,	10000,	0	/	junction point – east, north
k.	-3,	7,	5000,	2500	/	junction point – east, north
l.	-3,	15000,	-2500		/	pen up – absolute
m.	2,	0,	5000		/	pen down – absolute
n.	2,	0,	-10000		/	pen down – absolute
o.	-2,	3,	15000,	-2500	/	junction point – south, east, north
	–					End Edit card

Note that the above data could have been input in a more compact form such as:

1PA, 15000,-2500,11,5000,-2500,3, ...
. . . . . 0, -10000,-2,3,15000,-2500/

Section 9
ERROR MESSAGES

DESCRIPTIVE ERROR MESSAGES

MESSAGE	MEANING
ABORT TERMINATE TRANSLATOR	Due to one or more major errors in digitizing the drawing, the program terminates processing of this drawing. Correct the errors with digitizer correction cards and rerun the digitizer deck.
ALTER MENU ACTION INCORRECT	The method used in trying to modify a symbol menu was incorrect. No menu alterations will be made.
AN INVALID MENU NO. nn	Menu number nn is not in the range of allowable menu numbers (1 to 99). All subsequent ALTER MENU(s) will be ignored.
ANNOTATION NOT SPECIFIED	While in ANNOTATION Mode, the work area has been accessed before any alphanumeric information has been specified (via the keyboard menu). The placement location will be retained but without annotation.
ATTEMPT TO ADD SYMBOL name FAILED	Information only – reason for failure is in message preceding this one.
CAN NOT ALTER UNASSIGNED MENU NO. nn	Menu number nn cannot be modified unless the MODM option is specified on the $ASAP card. No menu modifications will be made.
CAN NOT APPLY TO OPTION RESET	OPTION RESET was accessed prior to accessing something that cannot be reset or defaulted via OPTION RESET. The command is ignored.
COMPONENT DESG. TABLE FULL, AAAAAA IGNORED	There are 71 component designator classes allowed per drawing when using automatic numbering. Component designator class AAAAAA is ignored.

DESCRIPTIVE ERROR MESSAGES (Cont)

MESSAGE	MEANING
COULD NOT FIND JUNCTION POINT FOR XX.XXX, YY.YYY nnn1 nnn2	In continuity connections, a junction point was not within an inch of the specified point XX.XXX, YY.YYY. No continuity connection will be made for the A.S.A.P. statement on drawing sequence No. nnn1, macro sequence No. nnn2.
DELETE-FROM ENDS EDIT FILE – edit card image and its sequence number, sequence number of card image on Symbol File	No Delete-to follows this card prior to running out of edit cards.
DELETE-FROM FOLLOWED BY SAME – edit card image and sequence number of card image on Symbol File.	Delete-from followed by Delete-from. Ignore first.
DELETE-FROM NOT IN SOURCE – edit card image and its sequence number, sequence number of card image on Symbol File	Delete-from sequence number not found in Symbol File data. End of edit.
DELETE-FROM OUT OF SEQUENCE – edit card image and its sequence number, sequence number of card image on Symbol File	Delete-from sequence number less than sequence number of present card in Symbol File. Ignore it and its associated Delete-to.
DELETE-FROM SAME AS PREVIOUS – edit card image and its sequence number, sequence number of card image on Symbol File	Delete-from has same or lower sequence number than previous. Ignore it and its associated Delete-to.
DELETE-TO NOT IN SOURCE – edit card image and its sequence number, sequence number of card image on Symbol File	Delete-to sequence number not found in Symbol File data. End of edit.
DELETE-TO OUT OF SEQUENCE – edit card image and its sequence number, sequence number of card image on Symbol File	Delete-to sequence number less than its associated Delete-from. Ignore pair.
DIGITIZER CARD nnn1 FOLLOWING CARD nnn2 OUT OF SEQ.	The digitizer output deck is out of sequence. Digitizer *card* nnn1 should precede digitizer card nnn2. The program will continue and process the digitizer access as found.
DIGITIZING SCALE OF nn.nnn IS ILLEGAL	A digitizing scale of nn.nnn was specified, but is illegal. Digitizing scale will be set to 1.0

DESCRIPTIVE ERROR MESSAGES (Cont)

MESSAGE	MEANING
DRAWING AAAAAA HAS NO ORIENTATION COORDINATES	EDIT DRAWING has been accessed and drawing name AAAAAA has no orientation coordinates. The X and Y dimensions digitized this time will be used and saved as the orientation coordinates for drawing AAAAAA.
DRAWING AAAAAA TO BE EDITED NOT ON FILE END AFTER TRANSLATOR	Major Error. Drawing AAAAAA to be edited or modified is not found on the Symbol File. Correct the file or the name specified and rerun the digitizer deck.
DRAWING SPECIFICATIONS INCORRECT, TERMINATE JOB	*FATAL ERROR* The drawing orientation information has been incorrectly specified. The relative position of the drawing on the digitizer table is therefore unknown to the program. Processing terminates on this drawing.
EDIT CARDS DO NOT AGREE WITH SYMBOL TYPE FOR name	Symbol type on the Begin Edit card does not match the symbol type in the Symbol File dictionary for the entry named. Editing not performed.
ERASE COUNT LIMIT EXCEEDED AT nnn	Digitizer access nnn exceeds a maximum of 20 consecutive ERASE commands. This ERASE is ignored.
ERROR 001	An attempt to nest a macro call beyond 5 levels has been encountered.
ERROR 002	A macro calls a macro which, in turn, calls the first macro (circular macro path).
ERROR 003	The tables which hold annotation substitutions for variable annotation are full – either too many substitutions (maximum of 250 in a nest) have been input or too many substitution characters have been input (maximum of 6000 characters in a nest).
ERROR 004	A substitution for variable annotation is required, but none is available – a blank is output as " ".
ERROR 005	More substitutions for variable annotation were input with a macro call than were necessary.

DESCRIPTIVE ERROR MESSAGES (Cont)

MESSAGE	MEANING
ERROR 006	A Symbol label or labeling annotation contained more than 24 characters per line.
ERROR 007	A decimal point was used in other than a numeric field.
ERROR 008	Alphabetic data was encountered in a numeric field.
ERROR 009	The amount of data used between asterisks (an A.S.A.P. logical record) caused the program to run out of storage area.
ERROR 010	A symbol or macro being called was not found on the Symbol File.
ERROR 011	A code word was specified that does not exist in the program.
ERROR 012	A code word which requires numeric data was encountered without the numeric data.
ERROR 013	Numeric data was encountered without a code word preceding it.
ERROR 014	Code word AL or AR was not immediately followed by the appropriate character (") used to enclose annotation data.
ERROR 015	An illegal character was encountered in the data.
ERROR 016	An illegal instruction was encountered in a symbol.
ERROR 017	More than five repeat nestings were encountered in a symbol or an end repeat was encountered before a repeat was started.
ERROR 018	More than 50 junction points were encountered in a symbol.
ERROR 019	Too many junction points were defined on the drawing.
ERROR 020	Code word CIRC=n.n was not immediately followed by: ,X=n.n, Y=n.n*
ERROR 021	A window was specified on the drawing where an unexpanded macro was placed.

DESCRIPTIVE ERROR MESSAGES (Cont)

MESSAGE	MEANING
ERROR 022	Code word CJ, CJC, or JC was not immediately followed by: ,X=n.n, Y-n.n, X=n.n, Y=n.n*
ERROR 023	Code word ARW was not immediately followed by: ,X=n.n, Y=n.n, X=n.n, Y=n.n*
ERROR 024	Code word WIN was not immediately followed by at least three, but not more than 24 pairs of ,X=n.n, Y=n.n instructions. More than 50 windows were being defined in a drawing or macro at one time.
ERROR 025	Code word REC was not immediately followed by: ,X=n.n, Y=n.n, X=n.n, Y=n.n*
ERROR 026	Code word MAJA or MINA was not immediately followed by three ,X=n.n, Y=n.n coordinate pairs; or Code word MAJA or MINA was followed by three ,X=n.n, Y=n.n coordinate pairs which were colinear; or Code word MAJA=n.n or MINA=n.n was not immediately followed by: ,X=n.n, Y=n.n, X=n.n, Y=n.n*; or The radius specified by MAJA=n.n or MINA=n.n was less than one-half the distance between the end points of the arc.
ERROR 027	Code word PNT was followed by data other than ,X=n.n, Y=n.n or ENDP.
ERROR 028	The last coordinate of a polygon was not equal to the first coordinate. (see PAINTING.)
ERROR 029	Less than four coordinates were specified to define a polygon. (see PAINTING.)
ERROR 030	More than 1000 points were specified in a set of polygons. (see PAINTING.)
ERROR 031	A radius of a circle equal to or less than zero was encountered in a symbol.
ERROR 032	A PIN LABEL DESIGNATION instruction in a symbol did not immediately follow an instruction defining a junction point.

DESCRIPTIVE ERROR MESSAGES (Cont)

MESSAGE	MEANING
ERROR 033	Code word P was not immediately followed by: "
ERROR 034	A Pin Label was encountered in the data without an associated Symbol Label or Macro Label.
ERROR 035	More than 16,384 D,X=nn.nnn, Y=nn.nnn, A.S.A.P. statements occurred on the drawing.
ERROR 036	Invalid Pin Label Annotation Placement Designator instruction in symbol.
ERROR 101 to 136	Error occurred in unexpanded macro. Refer to corresponding Errors 001 through 036.
FACTOR OF nn.nnn IS ILLEGAL	A factor of nn.nnn was specified, but is illegal. A factor of 1.0 is assumed.
FOLLOWING CORRECTION CARD nnnn OUT OF SEQUENCE – IGNORED	A digitizer *correction* card is out of sequence. The card nnnn is ignored.
GROUND STORAGE TABLE OVERFLOW	More than 100 "GROUND" symbols occurred on the drawing. All subsequent grounds are ignored.
ILLEGAL CHARACTER IN NUMERIC FIELD – data field	A character other than +, –, or a number was found in a field where it is illegal. The character is set to zero.
INCORRECT EDIT AA. . .A at nnn	Digitizer *correction* card nnn contains AA. . .A which cannot be recognized. The card is ignored.
INIT, AREA CAN NOT BE ACCESSED AT nnn	Digitizer access nnn is in the initialization control area. This area can be accessed at the beginning of a drawing or at the beginning of macro creation or editing. The initialization function is ignored.
INTERCONNECT STRINGING TABLE OVERFLOW	More than 100 line segments are used to connect two pins.
INVALID ACCESS AT nnn	Digitizer access nnn was made on the control menu but not within a defined area. The access is ignored.

DESCRIPTIVE ERROR MESSAGES (Cont)

MESSAGE	MEANING
INVALID CHARACTER IN SYMBOL	Digitizer access is a character (keyboard entry) that is invalid in drawing, symbol, and macro names. The character is ignored.
INVALID END MACRO AT nnn	Digitizer access nnn is an END MACRO access. However, there is no START MACRO or EDIT MACRO to associate with the END MACRO. The command is ignored.
INVALID KEYBOARD ACCESS AT nnn	Digitizer access nnn is a keyboard access and there is no control function to associate with it. The keyboard entry is ignored.
KEYBOARD BUFFER EXCEEDED AT nnn	Digitizer access nnn exceeds one of 4 keyboard buffer areas. The type of keyboard entry and the maximum number of characters are: (1) 6 characters for numeric entries (2) 8 characters for alphanumeric names (3) 250 characters for alphanumeric labeling (4) 67 characters for remarks The character is ignored.
KEYBOARD ENTRY EXPECTED AT nnn	Digitizer access nnn is a function that expected a keyboard entry. Processing continues.
MACRO AAAAAA TO BE EDITED NOT ON FILE, END AFTER TRANSLATOR	Major error. Macro AAAAAA to be edited or modified is not found on the Symbol File. Correct the file or the name of the macro and rerun the digitizer deck.
MACRO EDIT TABLE OVERFLOW – macro name	A maximum of 100 macros may be edited in any one run – macros in excess of this must be rerun.
MACRO IN DRAWING NOT ENDED, END AFTER TRANSLATOR	Major error. START MACRO or EDIT MACRO was accessed during drawing creation but END MACRO was not. Include the END MACRO command as a digitizer correction card and rerun the digitizer deck.

DESCRIPTIVE ERROR MESSAGES (Cont)

MESSAGE	MEANING
MATCH STRING LONGER THAN 72 CHARACTERS – edit card image	Match string contains too many characters. Ignore it.
MAXIMUM NO. OF PIN LABELS EXCEEDED	More than 50 pin labels were specified. All labels over the 50 maximum will be ignored.
MENU ENTRY FOR SYMBOL NAME name ON MENU number AT LOCATION postion DELETED	Information only – this occurs when a symbol is deleted from the Symbol File which had a corresponding menu entry.
MORE PINS PLACED THAN SPECIFIED	While creating a macro, more pin labels were placed than were specified. The last pin specified will be used to fulfill the placement requirements.
MORE WORK AREA REQUIRED at nnn	Digitizer access nnn is a mode control that required more work area accesses in the placement specifications. Placement will not occur and any unused work area accesses will be ignored.
NAME AAAAAA EXISTS ON FILE, END AFTER TRANSLATOR	Major error. Trying to create a new drawing or macro with a name (AAAAAA) that already exists on the Symbol File. Change the name specified via digitizer correction cards and rerun the digitizer deck.
NO DRAWING TO PROCESS FOR DRAWING PROCESSOR	No drawing was requested and the No Draw option was not set.
NULL MATCH STRING – edit card image	Match string does not contain any characters – ignore it.
NUMERIC EXPECTED AT nnn	Digitizer access nnn is a keyboard entry that is not a numeric character, but should be. The non-numeric character will be ignored.
OPTION INVALID IN CURRENT MODE AT nnn	Digitizer access nnn is a mode option control that was selected but that has no meaning in the current mode control. The option is ignored.
PAINTING ENDED FOR nnn	An end painting instruction ENDP is generated for the painting instruction PNT at access nnn.

DESCRIPTIVE ERROR MESSAGES (Cont)

MESSAGE	MEANING
PIN ASSIGNMENTS NOT IN MACRO	PIN LABELS accessed in ANNOTATION but not while creating a macro. Pin labels are ignored.
PIN LABEL NOT SPECIFIED – IGNORED	PIN LABELS was accessed but no label was given. The blank pin label will be ignored.
PINS SPECIFIED NOT EQUAL TO PIN ALLOWED	The number of pins specified does not equal the number of pins allowed for the symbol. The number of pins allowed for the symbol will be used.
POINTS ARE THE SAME – NO CONNECTION nnnn1 nnnn2	In continuity connections, the two points specified were snapped to a common junction point. No continuity connection will be made for the A.S.A.P. statement on drawing sequence No. nnn1, macro sequence No. nnn2.
PREVIOUS MACRO NOT ENDED	An attempt to create or edit a macro has been made while in the process of creating or editing a previously defined macro. The previous macro is ended and the new macro creation or modification begins.
STRING EDIT TABLE OVERFLOW	Maximum of 100 string edits allowed in one run – ignore remaining string edit input.
SYMBOL AAAAAA NOT FOUND ON SYMBOL FILE	The name AAAAAA requested for placement in SYMBOL OR MACRO Mode is not on the Symbol File at this time. If symbol or macro AAAAAA is created in this run, it will occur on the drawing. If not, another error message will be generated by the Drawing Processor.
SYMBOL FILE DICTIONARY OVERFLOW	Dictionary has no more room – program will stop. A new file must be created or unwanted entries deleted. This message is also followed by a listing of the entire Symbol File. If it is ascertained that the dictionary is not full, contact CalComp.
SYMBOL FILE FULL	No more room in file – program will stop. A new file must be created or unwanted entries deleted.
SYMBOL name ADDED TO SYMBOL FILE	Information only.

DESCRIPTIVE ERROR MESSAGES (Cont)

MESSAGE	MEANING
SYMBOL name DELETED FROM FILE	Information only.
SYMBOL NOT FOUND IN MENU	The symbol menu has been accessed, but there is not a symbol or macro at the location accessed. The dummy symbol XXXXXX is substituted and processing continues.
SYMBOL OR MACRO AAAAAA IS INVALID	The name AAAAAA has been requested in SYMBOL OR MACRO Mode, but is not the name of a symbol or macro. The dummy symbol XXXXXX is substituted and processing continues.
TOO MANY POINTS ON WINDOW	Major error. There is a maximum of 24 points allowed in defining a windowing polygon. Digitizer correction cards can be used to correct the polygon and then rerun the digitizer deck.
UNRECOGNIZABLE EDIT CARD – edit card image	Begin Edit card has illegal punch in Column 1. No further editing performed.
UNRECOGNIZABLE FIELD – edit card image	Data describing a symbol is punched incorrectly – symbol not added to file.
WINDOW ACCESS AT nnn INVALID	Digitizer access nnn was WINDOW which is in the initialization control area. This area can only be accessed at the beginning of an edit to a drawing or at the beginning of an edit to a macro. The windowing information will be ignored.

The following error messages are due to possible A.S.A.P. Symbol File maintenance errors by the program. These messages are followed by a listing of the Symbol File and should be brought to the attention of CalComp Applications Programming immediately. The program will stop after this listing is done.

DUPLICATE SYMBOL FILE NAME name, location

SYMBOL FILE CHAINING ERROR, HASH = number, CHAIN = number

SYMBOL FILE ENTRY NOT FOUND FOR name

CALIFORNIA COMPUTER PRODUCTS, INC.
2411 West La Palma Anaheim, California 92801
Tel. (714) 821-2011 TWX 910-591-1154

1056 5C 7/74