

AutoCAD for Electronics Part 2

Tapping the power of AutoCAD by writing your own menus.

BILL MARKWICK

Last month we had a general look at the AutoCAD drafting program, its enormous power, and some of the objections to it. Most of the complaints arise from the apparent awkwardness of entering commands and dealing with all the default settings.

In this issue, we'll look at solving those problems by means of custom menus. AutoCAD lets you rewrite all the menus: the onscreen menu (and submenus), the pulldown menus (Release 9 or 10 only) and the tablet menus (for digitizing tablets). Restructuring the commands and defaults to suit your sense of logic will make AutoCAD buzz along at dizzying speeds. It's safe to say that if you're using the program as it comes from the box, you're only tapping into a small part of its power.

A Recap

As we mentioned, the first thing to do before writing menus is to set the variables to suit you. This means that each new drawing will come onscreen with the snap and grid set to your preference, the blip-mode off (or on), the text size set, and any other variables reset the way you like.

To do this, call up the template drawing ACAD.DWG, set all the variables you want, either directly or with the SETVAR command, and save the file. All new drawings will then have the same defaults as

this drawing. A complete explanation of all the variable settings is in Appendix A.6 of the AutoCAD Reference Manual (with any new variables since publication listed in the Manual Supplement supplied with new releases). Note that a few of the variables display as "Read Only" — these can't be changed with SETVAR; also note that certain variable settings are stored with the individual drawing, not with ACAD.DWG — if you load a drawing and change the Blipmode to "On", for instance, that's what it will be every time for that particular drawing.

Loading Menus

During loading, AutoCAD is normally configured by ACAD.DWG to look for a menu file called ACAD.MNU, the one that comes with the software. If you've written your own, such as MY.MNU, you can make this the new standard by loading ACAD.DWG, typing MENU, entering your new menu name and saving. Or, if you prefer specific menus for specific tasks, you can change them at any time by typing MENU from your drawing file. The three types of menus (screen, tablet, pulldown) can be typed into a single file so they all work at once, or you can make each one a separate file.

The Screen Menu

The screen menu, which appears down the

right-hand side, lets you type in about 20 lines (it depends on the type of monitor). These can be as simple as your favorite 20 commands, or each line can be the title of a submenu, and you can continue nesting the submenus. In general, though, I've found that enormous strings of nested commands get you confused in no time; it's better to pick the most-often-used commands and structure them with no more than one level of submenu. Simplicity pays off. After all, the whole idea here is so that you can go faster with AutoCAD.

Here's the basic syntax for writing screen menus (use a word processor with ASCII output, such as WordStar in the N mode).

```
***SCREEN
LINE
Endp
Perp
ZOOM W
ZOOM A
(etc.)
***BUTTONS
;
```

The three asterisks tell AutoCAD the purpose of the menu (screen, pulldown, buttons, etc.). Note that spaces count as Returns (as do semicolons); though you can make the menu item any length, the onscreen display is truncated to eight

This 3D representation of a sine wave was drawn with the polyline Spline Curve function and plotted from the Vpoint display

characters. The Buttons section makes sure that one of your mouse or digitizer buttons is the same as the Return key (Button #1 is always "Pick"); if you have more buttons available, the others can become other AutoCAD commands. For instance, my 4-button SummaSketch puck uses this:

```
***BUTTONS
;
[CANCEL]^C
GRAPHSCR
```

Button #1 is Pick, #2 is a Return, #3 is Cancel (Control-C), and #4 returns to the drawing editor from the text mode (same as the F1 key).

The above menu is the simplest, and will do if all you want is few common commands. However, the menu structure is actually a simple programming language that can feed default values to AutoCAD's command prompt; here's a look at the menu macros and what you can do with them.

Macros

To create an extensive command for the menu, it's only necessary to give it a title (up to six characters) and enclose it in square brackets. Let's say you wanted to Insert a lot of symbols and didn't care to type in any new default values; we'll call the new command Quick Insert, or Q. INS:

```
[Q.INS]^CINSERT\;\;\;
```

The name Q. INS will appear on the screen menu. When you select it, the Control-C cancels any command in progress (a good idea, putting Control-C before each macro). The left backslash allows user entry, in this case the Block Name and Insertion Point. The three semicolons act as Returns, accepting the next three defaults (size, aspect, rotation).

That's about it for basics. You can make each macro as long or as short as you need.

Symbols and Submenus

Here are the symbols used in AutoCAD's menu macros:

```
[xxx] onscreen title
; (or Space) Return
' (apostrophe) transparent
*** menu title
** submenu title
$X = menuname activates submenu
$X = deactivates a submenu
```

```
\ pause for text or Pick entry
+ macro continued on next line
^ make following char. ASCII control code
```

The screen title is self-explanatory and should be kept under six characters; spaces are permitted. The semicolon should be used whenever you think AutoCAD may not insert a needed Return; for instance, ERASE \; waits for you to pick an object and then erases it — without the final semicolon it would simply wait after the object selection.

The apostrophe turns certain commands into transparent commands; *ie*, they can be used inside other commands. 'ZOOM and 'PAN will let you move around the drawing in the middle of a LINE command, for instance. Type HELP for a listing of the commands that can be used transparently.

The menu and submenu titles are self-explanatory, though the \$ function gets confusing. It's used to change the display to a submenu and back again. Here's the syntax used to go from a small main menu to a small submenu and back:

```
[LINEMENU]$S = LINE_MENU
SNAP
GRID
etc

**LINE_MENU
Line
Pline
Pedit
Endp
etc
[PREV]$S =
```

If you're at the main menu and select LINEMENU, AutoCAD looks for a submenu called **LINE_MENU and displays it (the underline character is used only for clarity).

If you then select PREV, you'll be back at the main menu.

The \ is inserted whenever you want the program to pause for user entry; for instance, whenever the activated command says "Select Objects:" or "Text:" or similar. The + is used at the end of a line when you want to continue the macro on another line without invoking a Return keystroke.

The ^ (caret) will convert the next character into the equivalent ASCII control code. For instance, if you type ^C, AutoCAD converts this to an actual Con-

trol-C (ASCII 03). Another example: type ^G and it's converted to CTRL-G, the Grid On/Off toggle (same as function key F7).

For more information, see Appendix B.2 in the AutoCAD Reference Manual. To really get the hang of it, as with any programming, you need to write something and then debug any problems. The usual difficulty with AutoCAD macros is that the macro steps may not be in sync with the command prompt. Make sure the proper number of Returns are included, which is the number one cause of problems. Another cause that doesn't show up immediately is putting your text editor into the document mode rather than the ASCII mode, which may add high bits or headers that AutoCAD doesn't understand. To see if this has happened: TYPE your file from the DOS prompt. You should be able to see if funny things have been added. Your word processor may be able to deal with this — in WordStar, for instance, reload the file in the N mode and use CTRL-B to reformat the entire file. This should get rid of the high bits.

Fig. 1 is a listing for a typical screen menu. I've put in the commands I would use, and they probably won't suit you. Feel free to edit the commands all over the place — it's really meant as an example of how the menu syntax works.

Before going on to the next section, here's an example of how you can draw with macros; it produces a rectangle on the screen:

```
[BOX]LIMITS;0,0;12,9;LINE;1,1;1,4;8,4;8,1;C;
```

After it draws the box, the Limits will remain at the default value of 0,0 (lower left) and 12,9 (upper right). If you're using other values, you may want to restore them.

Pulldown Menus

Before wasting any time trying these out, you should make sure that your program and monitor configuration can support pulldowns (versions prior to Release 9 will not, some monitors may not). If you're using the AutoCAD ACAD.MNU, move the mouse or digitizer up into the top line of the screen; you should be able to pull down the default menus. If you don't get these menus, your setup may not support pulldowns. As a last check, type SETVAR POPUPS. If it returns a 1, you're in. If it returns a zero, or an error message, you're

AutoCAD for Electronics, Part 2

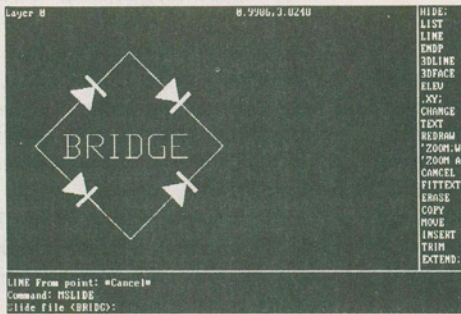


Fig. 3. To make the slide files for your icon display, call your block drawings into AutoCAD, make them fullscreen size and use MSLIDE to make the files.

```
res
cap
elcap
nnp
pnp
opamp
bridge
LED
diode
pot
dot2
dot
jump
arrow
gnd
CANCEL
```

Fig. 4. Now make an ASCII file with a text editor, listing each icon slide file on a separate line. Now use the SLIDELIB utility as described in the text to make a library.

```
***ICON
**IPARTS
[COMPONENTS]^C
[COMPLIB(res)]^Cinsert res
[COMPLIB(cap)]^Cinsert cap
[COMPLIB(elcap)]^Cinsert elcap
[COMPLIB(nnp)]^Cinsert nnp
[COMPLIB(pnp)]^Cinsert pnp
[COMPLIB(opamp)]^Cinsert opamp
[COMPLIB(bridge)]^Cinsert bridge
[COMPLIB(LED)]^Cinsert led
[COMPLIB(diode)]^Cinsert diode
[COMPLIB(pot)]^Cinsert pot
```

Fig. 5. Add this section to your MNU file as described in the text.

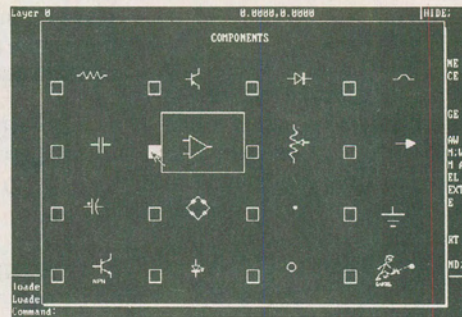


Fig. 6. You've got icon menus!

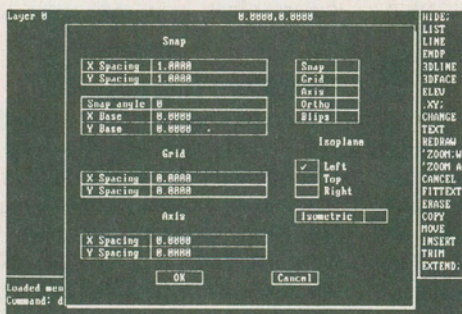


Fig. 7. One of the dialogue boxes provided by AutoCAD and called from a menu, with DDRMODES.

HIDE	LIST	LINE	ENDP	3DLIN	3DFAC	ELEV	.XY	CHG	TXT	REL
BLOCK	WBLOCK	BASE	ORTHO	OSNAP	POLYGON	PURGE	REDO	RENAME	SCRIPT	SKETCH
ON	OFF	CANCEL	SNAP	A	R	S	ISO	STA	GRID	.5
DIM	DIMV	DIMH	ALI	ANG	BAS	CEN	CON	DIA	EXIT	LEADE
C	E	F		L	M	N	O	P	Q	R
VP	VP	FR	REAR	VP	VP	UPPER	UPPER	3D	CON-	DEL
	0,0,1			L	R	LEFT	RT	CIRC	VERT	LAYER
LINE	ENDP	PERP	MID	INS	NEAR	TAN	CEN	INT	C	U
ERA	ERA	L	ERA	1	CANC	MOVE	ERA	W	BREAK	CHANGE
									SEL,C	X
									XTEND	TRIN

Fig. 8. A section of a custom digitizing tablet menu, about actual size. It's taped to the top of the tablet.

Fig. 1. A sample listing that will produce a Screen menu in AutoCAD. It's only one of the many ways you can arrange the commands and command structure; in this case the nesting never goes beyond one submenu, so it's easy to find your way around. Note: do not type in the {REM} markers or the comments that follow them.

```

***SCREEN {REM} sends commands to right side of screen
^CLINE {REM} the ^C cancels any command in progress
^CERASE
OOPS
UNDO
^CMOVE
'ZOOM W {REM} apostrophe lets you zoom inside other commands
^CLIST
^CINSERT
^CCOPY

[CANCEL]^C^C {REM} end of screen main menu

[LINEMENU]$$ = LINEMENU {REM} these five are submenus
[DISPLAY]$$ = DISPLAY {REM} and do not appear onscreen.
[EDIT]$$ = EDIT
[3D]$$ = 3D
[LAYERS]$$ = LAYERS

**LINEMENU {REM} first submenu for Line-related commands
^CLINE
^CPLINE
^CPEDIT
Endp {REM} these three allow one-time Object Snap
Int
Perp

ERASE
OOPS

^CARC
^CCIRCLE
^CELLIPSE
^CSKETCH
^CARRAY
^CLINETYPE
^CPOINT

[PREV]$$ = {REM} returns you to main menu

**DISPLAY {REM} the second submenu
'ZOOM {REM} Zoom commands can be used inside other commands
'ZOOM P
'ZOOM A
'ZOOM W
'PAN
[VU SAVE]^CVIEW S
[VU REST.]^CVIEW R
[VU DEL]^CVIEW D
REDRAW
REGEN

```

(continued)

(Continuation)

```

[PREV]$$ = {REM} returns you to main menu

**EDIT {REM} third submenu
^CERASE
^CERASE W
^CMOVE
^CMOVE W
^CCOPY
[MULTCOPY]^CCOPY;W;\;M {REM} the \ allows user entry
^CINSERT {REM} the ; is the same as Return
^CMINSERT {REM} or Space
^CTRIM
^CEXTEND
^CBREAK
^CSELECT
^CCHANGE
^CROTATE
^CEXPLODE
^CMIRROR
^CSTRETCH
^CPEDIT

[PREV]$$ = {REM} returns you to main menu

**3D {REM} the fourth submenu
^CELEV
^CLINE
^C3DLINE
Endp
.XY
^C3DFACE
^CSOLID
[PLAN VU]^CVPOINT;0,0,1 {REM} the next three send values
[UPPER RT]^CVPOINT;-2,-2.5,2 {REM} to the Viewpoint
[UPPER LT]^CVPOINT;-2,2.5,2 {REM} command

[PREV]$$ = {REM} returns you to main menu

**LAYERS {REM} the fifth submenu
^CLAYER
[LIST]^CLAYER;?;;;
[THAW]^CLAYER T
[FREEZE]^CLAYER F
[NEW]^CLAYER N
[CHANGE]^CLAYER S

[PREV]$$ = {REM} returns you to main menu

***BUTTONS
; {REM} makes Button 1 the PICK button and Button 2
{REM} becomes a Return. Buttons 3 and 4 can be added
{REM} as necessary.

```

AutoCAD for Electronics, Part 2

out of luck.

Assuming that all goes well, and that you'd rather have your own menus than the ones supplied by AutoCAD, you can either add the new pulldown menu onto our Screen menu as described above, or you can write a separate file for pulldowns only.

You can have up to ten main headings; each pulldown can have about 20 lines (it depends on your monitor); each pulldown can also have submenus. Here's the basic structure:

```
***POP1
**P1A
[MAIN MENU]
Line
Pline
Endp
etc
[SUBMENU B]$P1=$P1B $P1=*
[SUBMENU C]$P1=$P1C $P1=*

**P1B
[SUBMENU B]
ZOOM A
ZOOM W
PAN
[EXIT]$P1=P1A

**P1C
[SUBMENU C]
ERASE
OOPS
UNDO
[EXIT]$P1=P1A
```

In this demonstrator I've used Line, Endp, etc. as filler — you can substitute macros of any length just as you did with the Screen menu. Also: you can finally use names of a decent length — the pulldown window will just expand to accommodate long ones.

In the example, the pulldown window that's being displayed is labelled \$P1, so for instance, \$P1=\$P1B means that submenu \$P1B is the one on the screen. When you're finished with the menu, select EXIT to return to the main window; otherwise the submenu disappears but remains active. \$P1=* is a command that means "display the selected menu". Without it, the selected menu becomes active but doesn't immediately appear. Note: if the submenu is small, it will not completely overwrite the main menu, but will merge with it. To prevent this, fill up the submenu with blanks by using a right and left bracket [].

That's about it; the macro structure is

identical to the Screen type. Fig. 2 is a listing of a workable pulldown menu that gives two windows, FILE and DRAW, plus some submenus in the DRAW function. As I mentioned, it's only one of an infinity of ways to structure the commands: edit away until it suits you.

Icon Menus

This is where it gets really interesting. This is where you call your colleagues in to look at your custom-made icons, and word begins to spread that you're King CAD.

Besides that, they're very efficient to use, especially with large symbol libraries. Let's assume that you have a number of DWG files of electronic components that you use with the INSERT command, and that you want them to appear onscreen so you can make your choice (rather than typing the block filename each time).

Note that in the pulldown listing of Fig. 2, I've included the line [COMPONENT ICONS]\$I=IPARTS \$I=*. When selected, this macro sends AutoCAD looking for a Slide file library of my electronic component Blocks. When found, it's displayed full-screen. I can then pick any component from the symbols that appear, and the Icon function runs the Insert command automatically, drawing the component from the DWG library.

The nice thing about it is that you just give AutoCAD any number of slide files — it draws the boxes and figures out how to display them neatly. I've used 16 icons, which AutoCAD arranges in a 4 by 4 grid; you can put more, but the images get smaller and smaller.

Here are the steps:

1. Insert the \$I= macro in your pulldown as described.

2. Decide on what drawings you want displayed in the icon box; call them into AutoCAD one at a time. Expand them to full screen size and type MSLIDE, giving a suitable filename. This captures whatever drawing is on the screen as a file with the extension .SLD. See Fig. 3. Note that AutoCAD does not name the icons — if you need a label, add it to your drawing before making the slide file.

3. Make an ASCII file listing each slide on a new line, with or without the extension .SLD. Call it TEXTFILE or similar. See Fig. 4.

4. Check to see that you have the file SLIDELIB.EXE that comes with Release 9 and later. From DOS, type SLIDELIB COMPLIB <TEXTFILE. Now the SLIDELIB program will follow the list in

TEXTFILE (because of the left arrow, a DOS redirection symbol) and condense the slide files into one library file (COMPLIB.SLB).

Check to see that you have the file COMPLIB.SLB (for Component Library — you can call it anything else you like, as long as it terminates in .SLB). This compact list of the slide files will load rapidly into our pulldown icon menu. It's much better than having AutoCAD load each slide file separately. The SLD slide files are no longer needed and can be deleted once you prove that everything's working.

Lastly, I've added the following lines to the end of my pulldown menu (to make your own icon menu — substitute your slide library filename for COMPLIB, and your component names for mine):

```
***ICON
**IPARTS
[COMPONENTS]
[COMPLIB(res)]insert res
[COMPLIB(cap)]insert cap
[COMPLIB(elcap)]insert elcap
[COMPLIB(npn)]insert npn
[COMPLIB(pnp)]insert pnp
[COMPLIB(opamp)]insert opamp
[COMPLIB(bridge)]insert bridge
[COMPLIB(LED)]insert led
[COMPLIB(diode)]insert diode
[COMPLIB(pot)]insert pot
[COMPLIB(dot2)]insert dot2
[COMPLIB(dot)]insert dot
[COMPLIB(jump)]insert jump
[COMPLIB(arrow)]insert arrow
[COMPLIB(gnd)]insert gnd
[COMPLIB(CANCEL)]^C
```

The filename in round brackets is the one used in the ASCII list and calls the slide file from the library; the block name at the end of each line calls the block from your disk and inserts it in the drawing.

Now when you pull down your DRAW menu and click on Component Icons (or whatever name you're using), the screen will fill with a grid of icons; select the one you want, the menus disappear and the desired block should be onscreen, ready to be inserted.

DDMODES

There's one more feature that can save you a lot of time: the dialogue box. This function is provided with Release 9 and 10 and calls up a screen box full of options that can be selected with the cursor (Snap On/Off, Ortho On/Off, Grid, and many others). You can also type in new values for the

Fig. 2. This is an example listing that will generate pulldown menus in AutoCAD Release 9 or 10. The first SCREEN section allows commonly-used commands to appear at the right edge of the screen; if this menu is merged with the menu of Fig. 1, the SCREEN section below can be deleted.

Note: do not type in the {REM} markers or the comments that follow them.

***SCREEN {REM} commands that appear on the right screen menu.

LINE {REM} Add or replace commands to suit.

Endp
PLINE
ERASE
UNDO
MOVE
CHANGE
COPY
ZOOM W
ZOOM A

***POP1 {REM} the first or leftmost pulldown menu

**P1-FILE

[FILE] {REM} the menu name that will appear at the screen top
[SAVE, MAIN MENU]END
[SAVE AND RESUME]SAVE;;
[SAVE AS...]SAVE
[---] {REM} produces dividing lines for clarity
[ABANDON]QUIT
[---]
[RUN DOS COMMAND]^ CSHELL
[---]
[STATUS]^ CSTATUS
[PLOT]PLOT E {REM} plots Extents; remove the E if desired
[PRPLOT]PRPLOT E {REM} Prplots Extents; remove the E if desired
[INSERT]^ CINSERT
[QUICK INSERT]^ CINSERT \\\;; {REM} accepts all defaults
[BLOCK]^ CBLOCK
[WRITE BLOCK]WBLOCK

***POP2 {REM} the second pulldown menu

**P2A {REM} the main part of the second menu

[DRAW] {REM} menu name that will be displayed
[ERASE WINDOW]^ CERASE W
[UNERASE LAST]OOPS
[LINE]^ CLINE
[LINE ENDP]^ CLINE ENDP
[LINETYPE MENU]\$P2 = P2D \$P2 = * {REM} this displays P2D
[POLYLINE]^ CPLINE
[EDIT POLYLINE]^ CPEDIT
[COMPONENT ICONS]\$I = IPARTS \$I = * {REM} display icon
[POINT]POINT {REM} if it exists; see text
[ARC]^ CARC
[CIRCLE]^ CCIRCLE
[ELLIPSE]^ CELLIPSE
[UNDO]^ CUNDO
[ARRAY]^ CARRAY
[SKETCH]^ CSKETCH
[FILLET MENU]\$P2 = P2C \$P2 = * {REM} this displays P2C
[HATCH MENU]\$P2 = P2B \$P2 = * {REM} this displays P2B

**P2B {REM} this is the P2B (Hatch) submenu (continued)

[HATCH] {REM} this is the submenu name that appears on the top line
[HATCH]HATCH
[PATTERN LIST]^ CHATCH ?
[HATCH, LINE]^ CHATCH LINE
[DOTS]^ CHATCH DOTS
[METAL SECTION]^ CHATCH ANSI32
[WOOD]^ CHATCH ANSI33
[BRICK]^ CHATCH BRICK
[DASH]^ CHATCH DASH
[HOUNDSTOOTH]^ CHATCH HOUND
[EXIT]\$P2 = P2A {REM} this recalls the main Draw menu
[] {REM} these blanks ensure overwriting of the
[] {REM} main menu by the submenu

[]
[]
[]
[]
[]
[]
[]
[]
[]

**P2C {REM} this is the P2C (Fillet) submenu

[FILLET] {REM} this is the submenu name that appears on the top line
[FILLET]^ CFILLET
[MULT FILLET]^ C ^ C ^ CFILLET;
[FILLET PLINE]^ CFILLET P
[SET FILLETRAD]^ CFILLET R
[GET FILLETRAD](GETVAR "FILLETRAD")
[CHAMFER]CHAMFER
[EXIT]\$P2 = P2A {REM} this recalls the main Draw menu
[]
[] {REM} blanks ensure overwriting of main menu by the submenu

[]
[]
[]
[]
[]
[]
[]
[]
[]
[]
[]

**P2D {REM} this is the P2D (Linetype) submenu

[LINETYPE] {REM} the submenu name that appears on the top line
[LINESCALE]^ CLTSSCALE
[DOTTED LINE]^ CLINETYPE SET DOT;;LINE
[DASHED LINE]^ CLINETYPE SET DASHED;;LINE
[CONT. LINE]^ CLINETYPE SET CON-
TINUOUS;;LINE
[SMALL DASH]^ CLINETYPE SET HIDDEN;;LINE
[PHANTOM]^ CLINETYPE SET PHANTOM;;LINE
[CENTERLINE]^ CLINETYPE SET CENTER;;LINE
[BYLAYER]^ CLINETYPE SET BYLAYER;;LINE
[SEE LINETYPES]^ CLINETYPE ?;;
[EXIT]\$P2 = P2A {REM} this recalls the main Draw menu
[]
[] {REM} blanks ensure overwriting of main menu by the submenu

[]
[]
[]
[]

Continued on page 45

AutoCAD for Electronics, Part 2

Continued from page 19

variables if you want. The operation is very convenient, much more so than using the function key toggles.

There are a number of useful dialogue boxes, which can be called from the command line or put in as a menu item:

DDRMODES — this box contains drawing aid selections, such as Snap, Grid, Ortho, Isoplane, etc. It can be used transparently by starting the command with an apostrophe. It's shown in Fig. 7.

DDEMODES — the Entities box lets you change Color, Linetype, Elev, etc.

DDLMODES — the Layers box gives you convenient control of setting layers, something that usually involves a lot of confusing typing.

DDATTE — this one lets you edit the Attributes of various blocks (more on this in a future issue).

Tablet Menus

The digitizing tablet offers two major ad-

vantages to the CAD user. First, you can copy paper drawings into your computer just by taping them to the tablet and going over the drawing with the cursor (digitizer cursors usually have crosshairs for precise alignment); AutoCAD allows you to adjust the scale so that the copied drawing fits the screen the way you want it. Second, you can design up to four menus and tape them wherever you want them on your tablet. Mine, part of which is shown in Fig. 8, has 30 columns by 8 rows. The 240 selection points let me put in just about everything I need, including the alphabet and various macros.

The menu is written exactly as described for Screens and Pulldowns, except that you preface the sections with *****TABLET1**, *****TABLET2**, etc.

Next Month

In a future issue, we'll be looking at drawing a complex schematic in nothing flat: tips, tricks, and attributes.