SPECIAL KEYBOARDS
(*.WPK files)

One of the most powerful features of WordPerfect is that it allows you to re-map the entire keyboard to suit your needs. In addition to the "standard" WordPerfect keyboard, you should have received the files SHORTCUT.WPK, MACROS.WPK and ENHANCED.WPK as part of your original WordPerfect 5.1 package. (Keyboard files all have a .WPK extension.) These alternate keyboards assign useful functions to special keys.

To check which keyboard is currently installed, press Shift-F1 (Setup). Line 5 on the menu should give the name of the keyboard. If there is nothing on Line 5, the original or default keyboard is installed. To see which keyboards are available, you can press 5 for a list. If none are listed, you may need to tell WordPerfect where to look for your alternate keyboards with the Location of Files option in Setup.

To specify a location, press Shift-F1, then 6 for Location of Files; keyboards and macros are on line 2 of the Location of Files menu. I highly recommend that you keep your macros and keyboard files in a separate sub-directory, called C:\WP51\MACRO. This will keep your main C:\WP51 directory much cleaner and less cluttered.

For more general information about keyboard definitions, see "Keyboard Layout, Edit" and "Keyboard Layout, Map" in Reference.

After you've tried out some of the WordPerfect-supplied keyboards, you may want to create some of your own. Or, you may want to try some of the new keyboards included with this disk. These are:

SUPERKEY.WPK - my favorite all-around keyboard, works with my most commonly-used macros
BOX.WPK - a very useful special-purpose keyboard for drawing boxes with ASCII characters
EQUATION.WPK - lots of helpful assignments for use with the Equation Palette
MERGE.WPK - a real workhorse for anyone who uses the Merge feature regularly
GREEK.WPK - assigns characters from the Greek character map to shift and control keys
HEBREW.WPK - like GREEK.WPK, but assigns Hebrew characters to shift and control keys

Fuller descriptions of each keyboard are included below. These keyboard definitions were created by Joshua Brown and should accompany a much larger collection of shareware WordPerfect macros and other material on the same disk. This file is copyrighted material and may not be duplicated without permission by the author. For more information about shareware, and about duplicating these files, see the file READ.ME.

WordPerfect (tm) and WordPerfect Library (tm) are a trademarks of the WordPerfect Corporation, and are used with their permission. The WordPerfect Corporation does not endorse or provide support for this or any other non-WordPerfect product, nor is it responsible for any cost, loss, damage, etc. which may arise from use of this product.

To install the keyboards, first make sure that you have specified an appropriate location for your keyboard files (I recommend they go in a subdirectory named C:\WP50\MACRO). Then, simply copy all of the .WPK files on this disk into that directory.

To use them, Shift-F1 for Setup, then hit 5 for Keyboard Layout. Scroll down to the one you want to use, then hit 1 to select it. Exit back out to the main editing screen and you're done. When you press the keys on your keyboard, they will now respond to the new key assignments.

SUPERKEY.WPK

As mentioned above, this is my all-around favorite keyboard, which I use most of the time as a replacement for the standard WordPerfect keyboard definition. I use an AT-style keyboard, which has F11 and F12 keys in addition to the F1-F10 available on the original PC keyboard. I first started playing with keyboard definitions when I decided to map some frequently-used functions to those keys. Among the special features of SUPERKEY.WPK are:

Shft-F11	Italics
Shft-F12	Copy block
Ctrl-F11	Very large
Ctrl-F12	(blank - available for your use)
Alt-F11	Large
Alt-F12	Tab set
Ctrl-Up	Move Up by Sentence
Ctrl-Down	Move Down by Sentence
Alt-Up	Move Up by Paragraph
Alt-Left	Move Left One Column
Alt-Right	Move Right One Column
Alt-Down	Move Down by Paragraph
Ctrl-A	Advance to position entered by user
Ctrl-C	Column definition
Ctrl-D	Set double spacing
Ctrl-G	List graphics in c:\wp51\graphics

Ctrl-J	Control Justification on/off
Ctrl-L	Set left/right margins
Ctrl-M	List macros in c:\wp51\macro
Ctrl-N	Page numbering (no number on page 1)
Ctrl-O	Outline on/off
Ctrl-S	Set single spacing
Ctrl-T	Set top/bottom margins
Ctrl-X	Shadow on/off

BOX.WPK

WordPerfect's Line Draw feature (Control-F3, option 2) allows you to draw very nice little lines at right angles to each other, for making boxes, dividers, accents, simple maps, etc. In reality, this feature is simply a convenient way to access the ASCII characters in the ASCII 176-223 series.

Line Draw is handy, but you can't always get at it when you want to. For example, if you want to create a neatly-boxed prompt in a macro, Line Draw just isn't there. Also, if you want to draw a quick-and-dirty box to check yes/no replies in a survey or form, it's a pain to keep going in and out of Line Draw all the time.

You can access these ASCII characters as part of WordPerfect's Character Set feature (Character Set 3, Box Drawing), and I initially thought I would re-map the keyboard this way, which is basically how I created the HEBREW.WPK and GREEK.WPK definitions.

This, however, didn't work inside a macro, which is one of the reasons I wanted to have BOX.WPK on-line to begin with. Luckily, there is yet a third way to access the ASCII line drawing characters.

Any of these characters can be accessed simply by pressing Alt plus the appropriate reference number simultaneously. Thus, for example, Alt-187 produces ╗, Alt-188 produces ╝, and so on. You can have a lot of fun incorporating all of the ASCII characters into documents, and one benefit is that by doing it this way, you can save your document in DOS (or ASCII) format and not lose the appearance of what you've created. Saving a document in DOS format if you've produced these characters any other way tends to produce strange-looking results.

Anyway, BOX.WPK is a convenient way to use these line drawing characters. The key assignments are as follows:

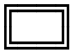
Home	╔	top left corner, double
Up	╦	top tee, double

PgUp	╖	top right corner, double
Left	╟	left tee, double
Right	╢	right tee, double
End	╙	bottom left corner, double
Down	╨	bottom tee, double
PgDn	╜	bottom right corner, double
Ctrl-Left	├	left tee, single
Ctrl-Right	┤	right tee, single
Ctrl-End	└	bottom left corner, single
Ctrl-PgDn	┘	bottom right corner, single
Ctrl-Home	┌	top left corner, single
Ctrl-PgUp	┐	top right corner, single
Ctrl-Up	┬	top tee, single
Ctrl-Num -	─	horizontal line, single
Ctrl-Num +	┼	cross tee, single
Ctrl-Num +	┼	cross tee, single
Ctrl-Down	┴	bottom tee, single
Ctrl-Num /	│	vertical line, single
Num +	╫	cross tee, double
Num -	═	horizontal line, double
Num /	║	vertical line, double

If this organization looks baffling, it're really quite simple. On most keyboards, the number keypad provides an alternate way to position the cursor -- 7 is Home, 1 is End, and so on. (AT-style keyboards have these as separate keys as well.)

I put the "corner" characters at the four corners of the keypad, and the "tee" characters on the arrow keys (e.g. the up arrow key is the "top tee" or ╥, left arrow is the "left tee" or ╟, etc.) Simple vertical lines use the /slash key, and simple horizontal lines use the -minus key, while "cross tees" ╫ use the +plus key.

In practice, you should find this system very easy to use and remember. To draw a box like this:



Make sure that you've loaded BOX.WPK as your active keyboard, then do the following keystrokes:

7 or Home (the top left corner)
Minus, minus (two horizional lines)
9 or PgUp (the top right corner

Return (to move to the next row)
1 or End (the bottom right corner)
Minus, minus (two horizontal lines)
3 or PgDn (the bottom right corner)

You'll note that Home, End, PgUp and PgDn in this case refer to the keys on the numeric keypad -- not to the dedicated keys by the same name if you have an AT-style keyboard. One of the nice quirks of BOX.WPK is that these cursor positioning keys are still available on AT keyboards. Of course, with BOX.WPK active, you've screwed up your numeric keypad for ordinary use -- but you can still access the numbers on the number row (the row above the QWERTYUIOP keys).

Double lines are produced by the numeric keypad alone. If you want single lines (e.g., ┌└│─┼┬┘ instead of ╔╚║═╬╦╝), simply hold down the Control/Ctrl key while pressing the keys you want.

As mentioned above, you can use BOX.WPK to allow you to draw neatly-boxed prompts in macros, which is a real trick to accomplish any other way. (You can do it by pressing Alt-ASCII number combinations, but you'll go cross-eyed looking up the ASCII numbers in Appendix A of your manual!)

I have reproduced the necessary keystrokes for a sample boxed macro prompt below, which is how they will have to appear on your Macro Editor screen:

```
{PROMPT}
························╔═══════════╗{Down}
························║···········║{Down}
························║·this·is·a·║{Down}
························║···test····║{Down}
························║···········║{Down}
························╚═══════════╝{Down}
{Down}{Down}{Down}{Down}{Down}{Down}~
{PAUSE}
{Screen}{Screen}
{QUIT}
```

(periods ..... represent spaces produced by the spacebar)
This will roll up a boxed prompt onto the screen, which will wait patiently there until you press ENTER, at which point it will disappear. If you're unfamiliar with creating macros of this level of sophistication (though this one is quite simple) using the Macro Editor screen, start digging in the

reference manual. It's a lot of fun.

Note in this case that the box was positioned using the space bar, and that you must specify an advance to the next line with a {Down}. Don't forget to end your {PROMPT} statement with a tilde (~), which in the example above is after the last {Down}.

That's about all there is to BOX.WPK -- it's pretty simple, and it does a job which is hard to do any other way, which is why I created it.

EQUATION.WPK

The Equation Palette in WP 5.1 is a major new feature, which allows mathematical and scientific expressions to be displayed in a visually attractive format. It's basically a very simple tool to use, but it takes some getting used to.

One of the nuisances of using the Equation Palette is that it takes several keystrokes to enter many commonly-used characters and command expressions. Regular users of this feature will welcome EQUATION.WPK's many shortcuts. If you haven't used this feature yet, simply reading the following assignments may give you some idea of the power of this new keyboard:

Alt-E	epsilon
Alt-R	rho
Alt-T	theta
Alt-I	INF (Infinity)
Alt-O	omega
Alt-P	pi
Alt-A	alpha
Alt-S	sigma
Alt-D	delta
Alt-F	phi
Alt-G	gamma
Alt-L	lambda
Alt-`	SIMEQ (Similar or Equal To)
Alt-\	LINE (Line)
Alt-B	beta
Alt-N	eta
Alt-M	mu
Alt-,	<= (Less Than or Equal To)
Alt-.	>= (Greater Than or Equal To)
Alt--	CONG (Congruent)
Alt-=	!= (Not Equal To)
Ctrl-Tab	← (Left Arrow)

Alt-Tab	→ (Right Arrow)
Ctrl-A	SUP (Superscript)
Ctrl-B	BAR (Bar)
Ctrl-D	DELTA (upper case)
Ctrl-E	IN (Member, Element)
Ctrl-F	FROM TO (x From a, To b)
Ctrl-G	GRAD (Nabla, Gradient)
Ctrl-I	INT (Integral)
Ctrl-L	OVERLINE (Overline)
Ctrl-N	GRAD (Nabla, Gradient)
Ctrl-O	OVER (Over)
Ctrl-P	PARTIAL (Partial)
Ctrl-Q	SQRT (Square Root)
Ctrl-S	SUM (Sum)
Ctrl-Z	SUB (Subscript)

If you use the Equation Palette regularly, you may want to make some editorial adjustments to the keyboard definition yourself. Please don't distribute your version with this documentation, though, or folks will get confused...

MERGE.WPK

There is no denying that WP 5.1 has added some powerful new Merge commands to the repertoire. There is also no denying that it's a pain to access some of them. Anyone who has played with WP 5.1's Merge feature has quickly become tired of cutting through a couple of layers of menus to get to the desired command.

MERGE.WPK helps speed things up by assigning frequently-used codes to Control-letter combinations. For those familiar with the older WP 5.0 codes, you will notice that most of the assignments in this keyboard are similar to the codes used in WP 5.0. Anyway, the keyboard assignments are as follows:

Alt-M	retrieve code reminder prompt - press ENTER when ready to clear the prompt and continue working
Ctrl-Left	Item left
Ctrl-Right	Item right
Ctrl-Up	Item up
Ctrl-Down	Item down
Ctrl-B	bell tone code
Ctrl-C	add comment to merge
Ctrl-D	date
Ctrl-E	end record
Ctrl-F	field
Ctrl-I	pause for keyboard input

Ctrl-M	merge code menu
Ctrl-N	next record
Ctrl-O	page off
Ctrl-Q	quit merge
Ctrl-T	print
Ctrl-X	field names

GREEK.WPK

This keyboard is another one which illustrates the principle that there is more than one way to skin a cat than by buttering it with parsnips.

You can access the various characters of the Greek alphabet individually if you like with the Compose feature by pressing Compose (Control-2 or Control-V) and then entering the appropriate Extended Character Set code (for example, 8,1 for alpha or α). You'll go cross-eyed and thumbfingered looking up the codes in Appendix P of the manual if you do this often.

An easier alternative is to use GREEK.WPK, which assigns all of the upper-case letters of the Greek alphabet to easy-to-remember Control-letter combinations, and all of the lower-case letters to Alt-letter combinations. Admittedly, this screws up any Alt-letter macros you've created temporarily, but only temporarily -- you'll get them back when you change to another keyboard definition, such as SUPERKEY.WPK.

The key assignments for GREEK.WPK are as follows:

Alt-A	LC alpha
Alt-B	LC beta
Alt-C	LC chi
Alt-D	LC delta
Alt-E	LC epsilon
Alt-F	LC phi
Alt-G	LC gamma
Alt-H	LC sigma #2
Alt-I	LC iota
Alt-J	LC eta
Alt-K	LC kappa
Alt-L	LC lambda
Alt-M	LC mu
Alt-N	LC nu
Alt-O	LC omicron
Alt-P	LC pi
Alt-Q	LC theta
Alt-R	LC rho
Alt-S	LC sigma #1

Alt-T	LC tau	
Alt-U	LC upsilon	
Alt-V		no character
Alt-W	LC omega	
Alt-X	LC xi	
Alt-Y	LC psi	
Alt-Z	LC zeta	
Ctrl-A	UC alpha	
Ctrl-B	UC beta	
Ctrl-C	UC chi	
Ctrl-D	UC delta	
Ctrl-E	UC epsilon	
Ctrl-F	UC phi	
Ctrl-G	UC gamma	
Ctrl-H		no character
Ctrl-I	UC iota	
Ctrl-J	UC eta	
Ctrl-K	UC kappa	
Ctrl-L	UC lambda	
Ctrl-M	UC mu	
Ctrl-N	UC nu	
Ctrl-O	UC omicron	
Ctrl-P	UC pi	
Ctrl-Q	UC theta	
Ctrl-R	UC rho	
Ctrl-S	UC sigma	
Ctrl-T	UC tau	
Ctrl-U	UC upsilon	
Ctrl-W	UC omega	
Ctrl-X	UC xi	
Ctrl-Y	UC psi	
Ctrl-Z	UC zeta	

You might note that I have reserved Alt-V on my own system for my View Document/preview macro. This is because not all of the Greek characters will display on all monitors. They will print on any printer with WordPerfect 5.1, however, and you can look at what you've done with the View Document feature (Shift-F7, option 6) or with my Alt-V macro, which does the same thing.

Greek purists will complain that I haven't accounted for all of the accent marks which are available in Character Set 8. If you want to fiddle around with the keyboard definition, I'm sure it can be accomplished, but all I've ever wanted to do was to transliterate Greek words from time to time.

If you find yourself writing long documents in Greek for some reason, you will probably find that printing them out takes a bodacious amount of time. That's because most printers will only accept Greek letters as individual graphics, not as individual letters.

If you happen to be using a laser printer with soft fonts, you may not have this problem, but I only have a dot-matrix printer, and large chunks of Greek take a LONG time to print -- but they WILL print, which they wouldn't under WordPerfect 5.0.

HEBREW.WPK

This keyboard definition is like unto GREEK.WPK, except that it produces the Hebrew characters in Character Set 9.

If you're seriously into writing Hebrew, you will have to figure out a way to write from right-to-left (or else write a macro which will reverse the order of whatever you've written). "I leave that as an exercise to the student", which means that the professor hadn't figured out such a macro at the time this documentation was written. Also, if you're seriously into writing Hebrew on a regular basis, you probably won't be using WordPerfect to do it!

The key assignments for HEBREW.WPK are:

Alt-A	Aleph
Alt-B	Beth/Bet
Alt-C	Koph
Alt-D	Daleth
Alt-E	He
Alt-G	Gimel
Alt-H	Tzade/Sade
Alt-I	Iod/Yod
Alt-J	Cheth/Heth/Het
Alt-K	Caph/Kaph
Alt-L	Lamed
Alt-M	Mem
Alt-N	Nun
Alt-O	Ain/'Ayin
Ctrl-O	Overstrike function -- see below
Alt-P	Pe
Alt-Q	Teth/Tet
Alt-R	Resh
Alt-S	Shin
Control-S	Sin
Alt-T	Tau/Taw
Alt-V	Preview macro
Alt-W	Waw/Vau

Alt-X	Samech/Samek
Alt-Z	Zain/Zayin

My local rabbi informs me that vowel pointers are seldom used except in writing prayers or in writing for young children. However, I have made key assignments for them anyway, figuring that if they're there in the character set, they might as well be accessible. They are accessed by using the following keys on the number keypad (on AT-style keyboards):

1	Cheerek
2	Tzareh
3	Seegol
4	Koobootz
5	Kamatz
6	Patach
7	Shevah
8	Chataf Patach
9	Chataf Seegol
0	Chataf Kamatz
*	Cholom
-	Shoorook
/	Alternate Yud/Iod?

In order to place the vowel pointers correctly, you will need to use the Overstrike feature, which is normally accessed by pressing Format (Shift-F8), 4 (other), 5 (overstrike) and then 1 (create), then entering the characters to overstrike. It's something of a nuisance to do it this way.

An shortcut alternative is to use my macro, OVRSTRIK.WPM, which automates the process of overstriking two characters. If you find yourself using this macro often, you might notice that I have assigned it to the Ctrl-O function in HEBREW.WPK.

If you're looking at this documentation on your display, the characters may not appear properly. If you want to see the characters and try to catch me in a goyische mistake, you can either a) print out this documentation or b) look at it with the Preview screen (Shift-F7, option 6, or use my Alt-V macro).

MORE GENERAL NOTES

You may prefer to juggle the various key assignments around to make them easier for you to remember. Feel free -- but please don't distribute this documentation along with your altered version, or my documentation will be even less accurate than it probably is already.

Some day, I may get around to writing keyboard definitions for the Cyrillic and Japanese character sets (sets 10 and 11), but by that time they'll probably have figured a simpler way to do things, not to mention a way to make the extended characters print faster.

If you find yourself using these alternate keyboard definitions often, as I do, you may want to speed up the switching process by using the macros which I have written. These are found on this same subdirectory, and should be stashed wherever you keep your macro files (I highly recommend putting them in a subdirectory called C:\WP51\MACRO).

The macros are, for quick reference:

SUPERKEY.WPM - loads SUPERKEY.WPK
BOXKEY.WPM - loads BOX.WPK
GREEK.WPM - loads GREEK.WPK
HEBREW.WPM - loads HEBREW.WPK

Each of these macros will notify you on the prompt line which one has just been loaded. For more information about macros, see the file, MACRO.DOC.