Chapter 20
Customizing menus and keyboard

Through the use of the many calculator menus you have become familiar with the operation of menus for a variety of applications. Also, you are familiar with the many functions available by using the keys in the keyboard, whether through their main function, or by combining them with the left-shift (↰), right-shift (↱) or ALPHA (ALPHA) keys. In this Chapter we provide examples of customized menus and keyboard keys that you may find useful in your own applications.

Customizing menus

A custom menu is a menu created by the user. The specifications for the menu are stored into the reserved variables CST. Thus, to create a menu you must put together this variable with the features that you want to display in your menu and the actions required by the soft menu keys. To show examples of customizing menus we need to set system flag 117 to SOFT menu. Make sure you do this before continuing (See Chapter 2 for instructions on setting system flags).

The PRG/MODES/MENU menu

Commands useful in customizing menus are provided by the MENU menu, accessible through the PRG menu (↰ PRG). Setting system flag 117 to SOFT menu, the sequence ↰ PRG NXT produces the following MENU soft menu:

The functions available are:

MENU: Activates a menu given its number

CST: Reference to the CST variable, e.g., ↱ shows CST contents.

TMENU: Use instead of MENU to create a temporary menu without overwriting the contents of CST

RCLMENU: Returns menu number of current menu

Menu numbers (RCLMENU and MENU functions)

Each pre-defined menu has a number attached to it. For example, suppose that you activate the MTH menu (◀ MTH). Then, using the function catalog (▶ CAT) find function RCLMENU and activate it. In ALG mode simple press ENTER after RCLMENU() shows up in the screen. The result is the number 3.01. Thus, you can activate the MTH menu by using MENU(3.01), in ALG, or 3.01 MENU, in RPN.

Most menus can be activated without knowing their numbers by using the keyboard. There are, however, some menus not accessible through the keyboard. For example, the soft menu STATS is only accessible by using function MENU. Its number is 96.01. Use MENU(96.01) in ALG mode, or 96.01 MENU in RPN mode to obtain the STAT soft menu.

Note: The number 96.01 in this example means the first (01) sub-menu of menu 96.

Custom menus (MENU and TMENU functions)

Suppose that you need to activate four functions for a particular application. Say, that you need to be able to quickly access the functions EXP, LN, GAMMA and ! (ALPHA ▶ 2) and you want to place them in a soft menu that you will keep active for a while. You could do this by creating a temporary menu with function TMENU, or a more permanent menu with function MENU. The main difference is that function MENU creates variable CST, while TMENU does not. With variable CST created permanently in your sub-directory you can always reactivate the menu using the specifications in CST by pressing ◀ CUSTOM . With TMENU the menu specifications are lost after you replace the temporary menu with another one.

For example, in RPN mode, a menu is created by using:

{EXP LN GAMMA !} ENTER TMENU ENTER

or

{EXP LN GAMMA !} ENTER MENU ENTER

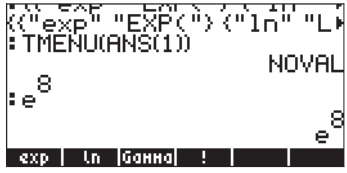
to produce the following menu:

To activate any of those functions you simply need to enter the function argument (a number), and then press the corresponding soft menu key.

In ALG mode, the list to be entered as argument of function TMENU or MENU is more complicated:

{{“exp”,”EXP(“},{“ln”,”LN(“},{“Gamma”,”GAMMA(“},{“!”,”!(“}}

The reason for this is that, in RPN mode, the command names are both soft menu labels and commands. In ALG mode, the command names will produce no action since ALG functions must be followed by parentheses and arguments. In the list shown above (for the ALG mode), within each sub-list you have a label for the key, e.g., “exp”, followed by the way that the function will be entered in the stack so that the argument to the function can be typed at the prompt, e.g., “EXP(“. We need not worry about the closing parenthesis, because the calculator will complete the parentheses before executing the function. The implementation of function TMENU in ALG mode with the argument list shown above is as follows. First, we enter the list, then we produce the temporary menu (see menu key labels) by using function TMENU(ANS(1)). We also show, in the left-hand side, the result of pressing the EXP soft menu key, i.e., the prompt EXP(. After typing 8 ENTER the result of the operation is shown in the right-hand side:



A simpler version of the menu can be defined by using

MENU({{”EXP(“,”LN(“,”GAMMA(“,”!(“}).

Enhanced RPN menu

The list presented above for the ALG mode, can be modified slightly to use in the RPN mode. The modified list will look like this:

{{“exp”,EXP},{“ln”,LN},{“Gamma”,GAMMA},{“!”,!}}

You can try using this list with TMENU or MENU in RPN mode to verify that you get the same menu as obtained earlier in ALG mode.

Menu specification and CST variable

From the two exercises shown above we notice that the most general menu specification list include a number of sub-lists equal to the number of items to be displayed in your custom menu. Each sub-list contains a label for the menu key followed by a function, expression, label, or other object that constitutes the effect of the menu key when pressed. Care must be exercised in specifying the menu list in ALG mode versus RPN mode. In RPN mode, the menu key action can be simply a calculator command (e.g., EXP, LN, etc., as shown above), while in ALG mode it has to be a string with the command prompt whose argument needs to be provided by the user before pressing ENTER and completing the command. The examples above illustrate the difference.

The general form of the argument list for commands TMENU or MENU in ALG mode is

{"label1","function1(","ls1(","rs1("}, {"label2", "function2(","ls2(","rs2("},…}

While, in RPN mode, the argument list has this format

{"label1", function1, ls1, rs1}, {"label2", function2, ls2, rs2},…}

In these specifications, function1, function 2, etc., represent the main operation of the key, while ls1, ls2, …, etc., represent the left-shift operation of the key. Similarly, rs1, rs2, …, etc., represent the right-shift operation of the key. This list will be stored in variable CST if command MENU is used. You can have a different CST variable in each sub-directory, and you can always replace the current contents of CST with those of other variables storing the properly formatted list to produce another custom menu.

Note: You can use a 21x8 GROB (See Chapter 22) to produce an icon in the soft menu keys. As an example, try, in RPN mode:

{{GROB 21 8 00000EF908FFF900FFF9B3FFF9A2FFF9A3FFF9A0FFF388FF "hp" }}
ENTER MENU

This will place the hp logo on key F1 . Pressing F1 places the text 'hp' in the command line.

Customizing the keyboard

Each key in the keyboard can be identified by two numbers representing their row and column. For example, the VAR key ([VAR]) is located in row 3 of column 1, and will be referred to as key 31. Now, since each key has up to ten functions associated with it, each function is specified by decimal digits between 0 and 1, according to the following specifications:

.0 or 1, unshifted key	0.01 or 0.11, not applicable
.2, key combined with [↰]	.21, key simultaneous with [↰]
.3, key combined with [↱]	.31, key simultaneous with [↱]
.4, key combined with [ALPHA]	.41, key combined with [ALPHA]
.5, key combined with [ALPHA] [↰]	.51, [ALPHA] key simultaneous with [↰]
.6, key combined with [ALPHA] [↱]	.61, [ALPHA] key simultaneous with [↱]

Thus, the VAR function will be referred to as key 31.0 or 31.1, while the UPDIR function will be key 31.2, the COPY function will be key 31.3, the upper-case J is key 31.4, and lower case j is key 31.5. (Key 31.6 is not defined). In general, a key will be described by the arrangement XY.Z, where X = row number, Y = column number, Z = shifting.

We can combine a given key with the USER key (left-shift associated with the [ALPHA] key, or [↰] [USER]) to create a customized key action. In principle, the entire keyboard can be re-defined to perform a number of customized operations.

The PRG/MODES/KEYS sub-menu

Commands useful in customizing the keyboard are provided by the KEYS menu accessible through the PRG menu ([↰] [PRG]). Setting system flag 117 to SOFT menu, the sequence [↰] [PRG] [NXT]

produces the following KEYS soft menu:

The functions available are:

ASN: Assigns an object to a key specified by XY.Z
STOKEYS: Stores user-defined key list
RCLKEYS: Returns current user-defined key list
DELKEYS: Un-assigns one or more keys in the current user-defined key list, the arguments are either 0, to un-assign all user-defined keys, or XY.Z, to un-assign key XY.Z.

Recall current user-defined key list

Use command RCLKEYS to see the current user-defined key list. Before any user-defined key assignments, the result should be a list containing the letter S, i.e., {S}.

Assign an object to a user-defined key

Suppose that you want to have access to the old-fashioned PLOT command first introduced with the HP 48G series calculator, but currently not directly available from the keyboard. The menu number for this menu is 81.01. You can see this menu active by using

ALG mode: `MENU(81.01)`
RPN mode: `81.01` ENTER `MENU` ENTER

If you want to have a quick way to activate this menu from the keyboard, you could assign this menu to the GRAPH key (F3) whose reference number is 13.0, i.e., first row, third column, main function. To assign an object to a key use function ASN, as follows:

ALG mode: `ASN(<<MENU(81.01)>>,13.0)`
RPN mode: `<< 18.01 MENU >>` ENTER `13.0` ENTER `ASN`

Another useful menu is the original SOLVE menu (described at the end of Chapter 6 in this Guide), which can be activated by using ↱ (hold) 7.

Operating user-defined keys

To operate this user-defined key, enter ↰ USER before pressing the F3 key. Notice that after pressing ↰ USER the screen shows the specification 1USR in the second display line. Pressing for ↰ USER F3 for this example, you should recover the PLOT menu as follows:

If you have more than one user-defined key and want to operate more than one of them at a time, you can lock the keyboard in USER mode by entering ↰ USER ↰ USER before pressing the user-defined keys. With the keyboard locked in USER mode, the specification USR will be shown in the second display line. To unlock the keyboard press ↰ USER once more.

Un-assigning a user-defined key

To remove the assignment performed above, use function DELKEYS, as follows:

ALG mode: `DELKEYS(13.0)`

RPN mode: `13.0` ENTER `DELKEYS` ENTER

Assigning multiple user-defined keys

The simplest way to assign several user-defined is to provide a list of commands and key specifications. For example, suppose that we assign the three trigonometric functions (SIN, COS, TAN) and the three hyperbolic functions (SINH, COSH, TANH) to keys F1 through F6, respectively, as user-defined keys. In RPN mode use:

`« SIN 11.0 COS 12.0 TAN 13.0 SINH 14.0 COSH 15.0 TANH 16.0 »` ENTER `STOKEYS` ENTER

In ALG mode use:

`STOKEYS({"SIN(" , 11.0, "COS(", 12.0, "TAN(", 13.0, "SINH(", 14.0, "COSH(", 15.0, "TANH(", 16.0})` ENTER

Operate these keys by using, for example, in RPN mode:

5 ↰ USER F1 4 ↰ USER F2 6 ↰ USER F3

2 ↰ USER F4 1 ↰ USER F5 2 ↰ USER F6

To un-assign all user-defined keys use:
ALG mode: DELKEYS(0) RPN mode: 0 DELKEYS

Check that the user-key definitions were removed by using function RCLKEYS.