


Chapter 26

Managing memory

In Chapter 2 we introduced the basic concepts of, and operations for, creating and managing variables and directories. In this Chapter we discuss the management of the calculator's memory, including the partition of memory and techniques for backing up data.

Memory Structure

The calculator contains a total of 2.5 MB of memory, out of which 1 MB is used to store the operating system (system memory), and 1.5 MB is used for calculator operation and data storage (user's memory). Users do not have access to the system memory component. To see the way in which the user's memory is partitioned, use the FILES function ( FILES). A possible result is shown below:



File Manager	
0:IRAM	240KB
1:ERAM	127KB
2:FLASH	955KB
HOME	240KB
LCASDIR	

At the bottom of the screen, there are two buttons: CHDIR and a button with CANCL and OK.

This screen indicates the existence of three memory ports, besides the memory corresponding to the HOME directory (see Chapter 2 in this guide). The memory ports available are:

- Port 0, labeled IRAM
- Port 1, labeled ERAM
- Port 2, labeled FLASH

Port 0 and the HOME directory share the same area of memory, so that the more data stored in the HOME directory, for example, the less memory is available for Port 0 storage. The total size of memory for the Port 0/HOME directory memory area is 241 KB.

Port 1 (ERAM) can contain up to 128 KB of data. Port 1, together with Port 0 and the HOME directory, constitute the calculator's RAM (Random Access Memory) segment of calculator's memory. The RAM memory segment requires continuous electric power supply from the calculator batteries to operate. To avoid loss of the RAM memory contents, a CR2032 backup battery is included. See additional details at the end of this chapter.

Port 2 belongs to the calculator's Flash ROM (Read-Only Memory) segment, which does not require a power supply. Therefore, removing the batteries of the calculators will not affect the calculator's Flash ROM segment. Port 2 can store up to 1085 KB of data.

A fourth port, Port 3, is available for use with an SD flash memory card. An example is shown below.



The port appears in File Manager only when an SD card is inserted.


The HOME directory

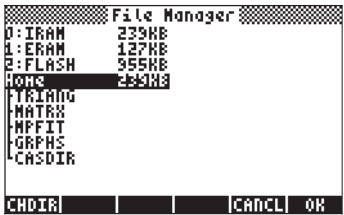
When using the calculator you may be creating variables to store intermediate and final results. Some calculator operations, such as graphics and statistical operations, create their own variables for storing data. These variables will be stored in the HOME directory or one of its directories. Details on the manipulation of variables and directories are presented in Chapter 2.


Port memory



Unlike the HOME directory, the memory in ports 0, 1 and 2 cannot be subdivided into directories, and it can only contain backup objects or library objects. These object types are described below.

Checking objects in memory



To see the objects stored in memory you can use the FILES function ( FILES). The screen below shows the HOME directory with five directories, namely, TRIANG, MATRX, MPFIT, GRPHS, and CASDIR.



Additional directories can be viewed by moving the cursor downwards in the directory tree. Or you can move the cursor upwards to select a memory port. When a given directory, sub-directory or port is selected, press  to see the contents of the selected object.

Another way to access port memory is by using the LIB menu ( LIB , associated with the  key). This action produces the following screen:



If you have any library active in your calculator it will be shown in this screen. One such library is the  (demo) library shown in the screen above. Pressing the corresponding soft-menu key () will activate this library. Pressing the port soft menu keys will open that memory port. Additional information on libraries is presented below.


Backup objects

Backup objects are used to copy data from your home directory into a memory port. The purpose of backing up objects in memory port is to preserve the contents of the objects for future usage. Backup objects have the following characteristics:

- Backup objects can only exist in port memory (i.e., you cannot back up an object in the HOME directory, although you can make as many copies of it as you want)
- You cannot modify the contents of a backup object (you can, however, copy it back to a directory in the HOME directory, modify it there, and back it up again modified)
- You can store either a single object or an entire directory as a single backup object. You cannot, however, create a backup object out of a number of selected objects in a directory.

When you create a backup object in port memory, the calculator obtains a *cyclic redundancy check* (CRC) or *checksum* value based on the binary data contained in the object. This value is stored with the backup object, and is used by the calculator to monitor the integrity of the backup object. When you restore a backup object into the HOME directory, the calculator recalculates the CRC value and compares it to the original value. If a discrepancy is noticed, the calculator warns the user that the restored data may be corrupted.

Backing up objects in port memory

The operation of backing up an object from user memory into one of the memory ports is similar to the operation of copying a variable from one sub-directory to another (see details in Chapter 2). You can, for example, use the File Manager ( FILES) to copy and delete backup objects as you would do with normal calculator objects. In addition, there are specific commands for manipulating back up objects, as described next.


Backing up and restoring HOME

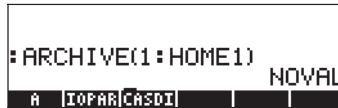
You can back up the contents of the current HOME directory in a single back up object. This object will contain all variables, key assignments, and alarms currently defined in the HOME directory. You can also restore the contents of your HOME directory from a back up object previously stored in port memory. The instructions for these operations follow.

Backing up the HOME directory

To back up the current HOME directory using algebraic mode, enter the command:


ARCHIVE(:Port_Number: Backup_Name)

Here, Port_Number is 0, 1, 2 (or 3, if an SD memory card is available – see below), and Backup_Name is the name of the backup object that will store the contents of HOME. The :: container is entered by using the keystroke sequence ::_. For example, to back up HOME into HOME1 in Port 1, use:



The image shows a calculator screen with the command `:ARCHIVE(1:HOME1)` entered. To the right of the command, the word `NOVAL` is displayed. Below the screen, a row of function keys is visible, including `IOPAR`, `CAS`, `DI`, and others.

To back up the HOME directory in RPN mode, use the command:

: Port_Number : Backup_Name  ARCHIVE

Restoring the HOME directory


To restore the Home directory in algebraic mode use the command:

RESTORE(: Port_Number : Backup_Name)

For example, to restore the HOME directory out of backup object HOME1, use:

RESTORE<: 1:HOME1>

In RPN mode use:


: Port_Number : Backup_Name  RESTORE








Note: When you restore a HOME directory backup two things happen:

- The backup directory overwrites the current HOME directory. Thus, any data not backed up in the current HOME directory will be lost.
- The calculator restarts. The contents of history or stack are lost.

Storing, deleting, and restoring backup objects


To create a backup object use one of the following approaches:

- Use the File Manager ( FILES) to copy the object to port. Using this approach, the backup object will have the same name as the original object.
- Use the STO command to copy the object to a port. For example, in algebraic mode, to back up variable A into a backup object named AA in port 1, use the keystroke sequence:

   ::   A  A 

- Use the ARCHIVE command to create a backup of the HOME directory (see above).


To delete a backup object from a port:

- Use the File Manager ( FILES) to delete the object as you would a variable in the HOME directory (see Chapter 2).
- Use the PURGE command as follows:


In algebraic mode, use: PURGE(: Port_Number : Backup_Name)

In RPN mode, use: : Port_Number : Backup_Name PURGE



To restore a backup object:

- Use the File Manager ( FILES) to copy the backup object from Port memory to the HOME directory.
- When a backup object is restored, the calculator performs an integrity check on the restored object by calculating its CRC value. Any discrepancy between the calculated and the stored CRC values result in an error message indicating a corrupted data.

Using data in backup objects

Although you cannot directly modify the contents of backup objects, you can use those contents in calculator operations. For example, you can run programs stored as backup objects or use data from backup objects to run programs. To run backup-object programs or use data from backup objects you can use the File Manager ( FILES) to copy backup object contents to the screen.

Alternatively, you can use function EVAL to run a program stored in a backup object, or function RCL to recover data from a backup object as follows:

- In algebraic mode:
 - To evaluate a back up object, enter:
EVAL(argument(s), : Port_Number : Backup_Name)
 - To recall a backup object to the command line, enter:
RCL(: Port_Number : Backup_Name)
- In RPN mode:
 - To evaluate a back up object, enter:
Argument(s)  : Port_Number : Backup_Name EVAL
 - To recall a backup object to the command line, enter:
: Port_Number : Backup_Name  RCL

Using SD cards

The calculator has a memory card port into which you can insert an SD flash card for backing up calculator objects, or for downloading objects from other sources. The SD card in the calculator will appear as port number 3.

Inserting and removing an SD card

The SD slot is located on the bottom edge of the calculator, just below the number keys. SD cards must be inserted facing down. Most cards have a label on what would usually be considered the top of the card. If you are holding the HP 50g with the keyboard facing up, then this side of the SD card should face down or away from you when being inserted into the HP 50g. The card will go into the slot without resistance for most of its length and then it will require slightly more force to fully insert it. A fully inserted card is almost flush with the case, leaving only the top edge of the card visible.

To remove an SD card, turn off the HP 50g, press gently on the exposed edge of the card and push in. The card should spring out of the slot a small distance, allowing it now to be easily removed from the calculator.

Formatting an SD card

Most SD cards will already be formatted, but they may be formatted with a file system that is incompatible with the HP 50g. The HP 50g will only work with cards in the FAT16 or FAT32 format.

You can format an SD card from a PC, or from the calculator. If you do it from the calculator (using the method described below), make sure that your calculator has fresh or fairly new batteries.

Note: formatting an SD card deletes all the data that is currently on it.

1. Insert the SD card into the card slot (as explained in the previous section).
2. Hold down the **ON** key and then press the **F4** key. Release the **F4** key and then release the **ON** key. The system menu is displayed with several choices.
3. Press 0 for FORMAT. The formatting process begins.
4. When the formatting is finished, the HP 50g displays the message "FORMAT FINISHED. PRESS ANY KEY TO EXIT". To exit the system menu, hold down the **ON** key, press and release the **F3** key and then release the **ON** key.

The SD card is now ready for use. It will have been formatted in FAT32 format.

Alternative method

When an SD card is inserted, **FORMAT** appears as an additional menu item in File Manager. Selecting this option reformats the card, a process which also deletes every object on the card.

Accessing objects on an SD card

Accessing an object from the SD card is similar to when an object is located in ports 0, 1, or 2. However, Port 3 will not appear in the menu when using the LIB function (\rightarrow LIB). The SD files can only be managed using the Filer, or File Manager (\leftarrow FILES). When starting the Filer, the Tree view will appear as follows if you have an SD card inserted:



Long names of files on an SD card are supported in the Filer, but are displayed as 8.3 characters, as in DOS, i.e., displayed names will have a maximum of 8 characters with 3 characters in the suffix. The type of each object will be displayed, unless it is a PC object or an object of unknown type. (In these cases, its type is listed as String.)

In addition to using the File Manager operations, you can use functions STO and RCL to store objects on, and recall objects from, the SD card, as shown below. You can also use the PURGE command to erase backup objects in the SD card. Long names can be used with these commands (namely, STO, RCL, and PURGE).

Storing objects on an SD card

To store an object, use function STO as follows:

- In algebraic mode:

Enter object, press $\text{STO} \rightarrow$, type the name of the stored object using port 3 (e.g., I3:VAR1), press ENTER .

- In RPN mode:

Enter object, type the name of the stored object using port 3 (e.g., I3:VAR1), press $\text{STO} \rightarrow$.

Note that if the name of the object you intend to store on an SD card is longer than eight characters, it will appear in 8.3 DOS format in port 3 in the Filer once it is stored on the card.

Recalling an object from an SD card

To recall an object from the SD card onto the screen, use function RCL, as follows:

- In algebraic mode:
Press \leftarrow **RCL** , type the name of the stored object using port 3 (e.g., **#3:VAR1**), press **ENTER** .
- In RPN mode:
Type the name of the stored object using port 3 (e.g., **#3:VAR1**), press \leftarrow **RCL** .

With the RCL command, it is possible to recall variables by specifying a path in the command, e.g., in RPN mode: **#3: (path) ENTER RCL**. The path, like in a DOS drive, is a series of directory names that together specify the position of the variable within a directory tree. However, some variables stored within a backup object cannot be recalled by specifying a path. In this case, the full backup object (e.g., a directory) will have to be recalled, and the individual variables then accessed in the screen.

Note that in the case of objects with long files names, you can specify the full name of the object, or its truncated 8.3 name, when issuing an RCL command.

Evaluating an object on an SD card

To evaluate an object on an SD card, insert the card and then:

1. Press \leftarrow **:** . This puts a double colon on the edit line with the cursor blinking between the colons. This is the way the HP 50g addresses items stored in its various ports. Port 3 is the SD card port.
2. Press **3** **▶** **'** **ALPHA** **ALPHA** [name of the object] **ENTER** . This will place the name and path of the object to be evaluated on the stack.
3. To evaluate the object, press **EVAL** .

Note that in the case of objects with long files names, you can specify the full name of the object, or its truncated 8.3 name, when evaluating an object on an SD card.

Purging an object from the SD card

To purge an object from the SD card onto the screen, use function PURGE, as follows:

- In algebraic mode:
Press **TOOL** **PURGE**, type the name of the stored object using port 3 (e.g., **#3:VAR1**), press **ENTER**.
- In RPN mode:
Type the name of the stored object using port 3 (e.g., **#3:VAR1**), press **TOOL** **PURGE**.

Note that in the case of objects with long files names, you can specify the full name of the object, or its truncated 8.3 name, when issuing a PURGE command.

Purging all objects on the SD card (by reformatting)

You can purge all objects from the SD card by reformatting it. When an SD card is inserted, **FORMAT** appears an additional menu item in File Manager. Selecting this option reformats the entire card, a process which also deletes every object on the card.

Specifying a directory on an SD card

You can store, recall, evaluate and purge objects that are in directories on an SD card. Note that to work with an object at the root level of an SD card, the **⏮** key is used. But when working with an object in a subdirectory, the name containing the directory path must be enclosed using the **⏭** **⏮** keys.

For example, suppose you want to store an object called PROG1 into a directory called PROGS on an SD card. With this object still on the first level of the stack, press:

⏮ **:** **3** **⏭** **⏮** **"** **ALPHA** **ALPHA** **P** **R** **O** **G** **S** **⏭** **÷** **P** **R** **O** **G** **/** **ENTER** **STO▶**

This will store the object previously on the stack onto the SD card into the directory named PROGS into an object named PROG1. Note: If PROGS does not exist, the directory will be automatically created.

You can specify any number of nested subdirectories. For example, to refer to an object in a third-level subdirectory, your syntax would be:

:3:"DIR1/DIR2/DIR3/NAME"

Note that pressing $\boxed{\text{ALPHA}}$ $\boxed{\rightarrow}$ $\boxed{\div}$ produces the forward slash character.

Using libraries

Libraries are user-created binary-language programs that can be loaded into the calculator and made available for use from within any sub-directory of the HOME directory. In addition, the calculator is shipped with two libraries that together provide all the functionality of the Equation Library.

Libraries can be downloaded into the calculator as a regular variable, and, then, installed and attached to the HOME directory.



Installing and attaching a library

To install a library, list the library contents in the stack (use $\boxed{\rightarrow}$ variable soft-menu key, or function RCL) and store it into port 0 or 1. For example, to install a library variable into a port use:

- In algebraic mode: $\text{STO}(\text{Library_variable}, \text{port_number})$
- In RPN mode: $\text{Library_variable} \boxed{\text{ENTER}} \text{port_number} \boxed{\text{STO}\blacktriangleright}$

After installing the library contents in port memory you need to attach the library to the HOME directory. This can be accomplished by rebooting the calculator (turning the calculator off and back on), or by pressing, simultaneously, $\boxed{\text{ON}}$ $\boxed{\text{F3}}$. At this point the library should be available for use. To see the library activation menu use the LIB menu ($\boxed{\rightarrow} \text{LIB}$). The library name will be listed in this menu.

Library numbers

If you use the LIB menu ( ) and press the soft menu key corresponding to port 0, 1 or 2, you will see library numbers listed in the soft menu key labels. Each library has a three or four-digit number associated with it. (For example, the two libraries that make up the Equation Library are in port 2 and are numbered 226 and 227.) These numbers are assigned by the library creator, and are used for deleting a library.

Deleting a library

To delete a library from a port, use:

- In algebraic mode: `PURGE(:port_number: lib_number)`
- In RPN mode: `: port_number : lib_number PURGE`

Where `lib_number` is the library number described above.

WARNING: Libraries 226 and 227 in port 2 constitute the Equation Library. You can delete these libraries just as you can a user-created library. However, if you are thinking of deleting these libraries but there is some likelihood that you will need to use the Equation Library in the future, you should copy them to a PC, using the HP 48/50 Calculator Connectivity Kit, before deleting them on the calculator. You will then be able to re-install the libraries later when you need to use the Equation Library.

Creating libraries

A library can be written in Assembler language, in System RPL language, or by using a library-creating library such as LBMKR. The latter program is available online (see for example, <http://www.hpcalc.org>). The details of programming the calculator in Assembler language or in System RPL language are beyond the scope of this document. The user is invited to find additional information on the subject online.

Backup battery

A CR2032 back up battery is included in the calculator to provide power backup to volatile memory when changing the main batteries. It is recommended that you replace this battery every 5 years. A screen message

will indicate when this battery needs replacement. The diagram below shows the location of the backup battery in the top compartment at the back of the calculator.

