

# RShell v0.01 - user guide

## Introduction

While the software built into the *PocketBook II (Psion 3A)* is generally very good, it falls down in instances when certain things need to be done for which there is no real support. Two examples are dumping a file to see what is in it, and basic file management...

- There can be good reasons to want to dump a file. Perhaps you wish to examine a file without loading it into something? Or maybe there is simply no native support – an example being the "Screen.pic" file that is created when you press **Ctrl-Shift-Acorn-S**.
- While you can press **Tab** from the Desktop to "manage" files, you'll quickly find that renaming several files in a subdirectory can mean starting all over again for each file; in other words: **Tab**, select directory, select file, press **Acorn-R...** wouldn't it have been so much better if the file manager hung around until you explicitly dismissed it? Psion must have thought so too, as the series 3C has a proper file manager built-in...

Of course, we must not lose track of the fact that these are *power user* options. But, then, if you weren't a power user, you probably wouldn't be reading this!

There is very little help built in to *RShell*. This is intentional, to further reduce memory requirements.

All of the screenshots within this documentation come from a real *PocketBook II*, and not the *3A emulator*. They are from a very early version of the software and are not representative of the anticipated 'final' state of the software... *This user guide was originally written in Write on my 3A, mostly while in the car, and later converted to a DTP form with OvationPro.*

## Data security

Several commands have the potential to cause data loss, either by deleting files or by killing errant tasks. You are prompted to confirm your action if you are deleting files. A red **!** marks commands to be careful with. *RShell* absolutely will **not** allow you to kill off the background system tasks. There is no 'hidden' option to do such a thing as, well, it'd be a really stupid idea...

## The prompt

When you first start *RShell*, you will see:

```
Welcome to RShell version 0.01 (2006/08/11)
Enter HELP for a list of commands available.
[ Fri, 2006/08/11, 21:07:45 ] (13920 bytes free)
LOC::M:\ >
```

**RShell © 2006 Rick Murray**

This is a brief welcome message, followed by the prompt.

The top line of the prompt shows the current date and time, followed by the amount memory that is available.

The lower line shows the current path followed by a '>' (DOS-like).

At the prompt, just enter your commands. I'm sure you've all used a command line before, be it RISC OS or DOS or some flavour of Unix.

The text size is *necessarily small*, due to the need to squeeze as much as possible into a limited screen. Using this text size allows up to 80 columns by 17 rows.

## Shell commands

Please note the following syntax:

<something> This denotes a parameter that is passed to the command.

[anything] This denotes an *optional* parameter.

Example:

```
MyCmnd <filename> [<format>]
```

This specifies a command "MyCmnd" that *requires* a filename as its first parameter. The second parameter is a format specifier, and it is *optional*, therefore both of the following examples would be valid:

```
MyCmnd \MyFile.dat
```

```
MyCmnd \MyFile.dat DATA
```

If you see a vertical bar '|', that means *OR*.

```
MyCmnd ONE|TWO
```

```
MyCmnd [ONE][TWO]
```

In the first example, you *must* specify "one" or "two". The second example is the same except that the choices are *optional* – you can specify nothing!

The documentation will tell you what information you are supposed to provide.

If you get stuck, typing ? will provide a list of commands, while typing the command without any of its parameters will remind you what the correct syntax is (unless, of course, the command has an action upon no parameters, in which case it will perform that action, i.e. "VER" will display version information).

When you are expected to enter file specifiers, wild cards are generally not accepted.

If you have less than 8K free memory, you will see a warning to this effect before the time and date.

If you have less than 3K free memory, you will instead be informed that *RShell* cannot continue and that you should free up some memory and then press 'Y'.

In the latter case, you can often continue – it depends on what you wish to do – however if *RShell* aborts due to lack of memory... you were warned!

The main memory consumer is the "VIEW" command, which will try to enable the grey plane for drawing into. You may need upwards of 24K free for that.

Due to pre-declared global variables and arrays, the majority of *RShell* has low variance in memory consumption.

## THE COMMANDS:

- ?

This will display brief information on the version of *RShell* that you are using, followed by a list of commands supported.

```
[F:\r, 2008/08/11, 21:08:28] (13920 bytes free)
LOC::M:\ >Help
This is RShell v0.01 (2008/08/11).
The following commands are available:
      ?          about      attrib      batt         cd           cls
      copy       del         delete      dir          dump        exit        help
      info       kill        ls          mem         mkdir       quit        resume
      rmdir      suspend    tasks      type        ver         view        x
```

You *cannot* enter a command after the help for more specific help, however, you *can* enter a command that takes parameters *without* any parameters in order to be reminded what is expected of you. Here is both in action:

```
[F:\r, 2008/08/11, 21:23:48] (12912 bytes free)
LOC::M:\AGN\ >help copy
Sorry, there is no help available for individual commands.
Those commands that take parameters will remind you of their syntax if you
enter the command without specifying any parameter(s).
Use HELP on its own for a list of commands available.
```

```
[F:\r, 2008/08/11, 21:23:54] (12912 bytes free)
LOC::M:\AGN\ >copy
Syntax: Copy <source> <dest>
```

- . [-1] [<filespec>]
- DIR [-1] [<filespec>]
- LS [-1] [<filespec>]

These commands list the contents of the current directory. The default style is a listing of contents only, though you can specify the optional **-1** parameter to display a list with file size, date, etc. Directories are displayed in CAPITALS, while files are displayed with an Initial capital. You can specify a file spec *which may contain wildcards* in order to restrict what is shown, i.e.:

. \*.word

to list only word processor documents. Use **\*** for “anything” and **?** for a single character.

Output will pause after every screenful waiting a key press.

Short form:

```

[Fri, 2006/08/11, 21:11:37] (14304 bytes free)
LOC::M:\ >.
AGN          APP          DAT          OPL          OPD          SCO
SCR          SPR          WDR          WLD          WRD          OPD
Sys#tycl.pdd Sys#rpcs.img Screen.pic

15 objects listed: 3 files (23412 bytes), and 12 directories.

```

Long form screen 1:

```

[Fri, 2006/08/11, 21:12:28] (14048 bytes free)
LOC::M:\ >. -1
AGN          0 ---D- 1993/07/04, 10:17:10
APP          0 ---D- 1993/07/04, 10:17:16
DAT          0 ---D- 1993/07/04, 10:17:48
OPL          0 ---D- 1993/07/04, 10:17:58
OPD          0 ---D- 1993/07/04, 10:18:12
SCO          0 ---D- 1993/07/04, 10:18:20
SCR          0 ---D- 1993/07/04, 10:18:24
SPR          0 ---D- 1993/07/04, 10:18:26
WDR          0 ---D- 1993/07/04, 10:18:30
WLD          0 ---D- 1993/07/04, 10:18:54
WRD          0 ---D- 1993/07/04, 10:18:56
OPD          0 ---D- 2006/08/07, 21:08:54
Sys#tycl.pdd 768 ----M 1994/09/30, 16:41:06
[Paused, press SPACE to continue or ^C to abort]

```

and screen 2:

```

OPL          0 ---D- 1993/07/04, 10:17:58
OPD          0 ---D- 1993/07/04, 10:18:12
SCO          0 ---D- 1993/07/04, 10:18:20
SCR          0 ---D- 1993/07/04, 10:18:24
SPR          0 ---D- 1993/07/04, 10:18:26
WDR          0 ---D- 1993/07/04, 10:18:30
WLD          0 ---D- 1993/07/04, 10:18:54
WRD          0 ---D- 1993/07/04, 10:18:56
OPD          0 ---D- 2006/08/07, 21:08:54
Sys#tycl.pdd 768 ----M 1994/09/30, 16:41:06
Sys#rpcs.img 3712 ----M 1996/10/30, 11:22:38
Screen.pic   16892 ----M 2006/08/11, 21:12:36

15 objects listed: 3 files (21372 bytes), and 12 directories.
[Fri, 2006/08/11, 21:12:57] (16096 bytes free)
LOC::M:\ >

```

- **ATTRIB <file> [+A] | [-A] | [+M] | [-M] [+H] | [-H] [+R] | [-R] [+S] | [-S]**

Files can have four attributions:

- A** Archive – has it been modified?
- M** Same as ‘A’, but in Psion parlance (the “Archive” bit is a DOSism).
- H** Hidden – file is not displayed in the Desktop (System screen).
- R** Read only – file can neither be updated nor deleted.
- S** System – doesn’t seem to have any special significance.

You can set attributes by prefixing the attribute letter with a ‘+’, and *unset* them by prefixing ‘-’. You may set multiple flags at one time, i.e. **+A-R** but note that *behaviour is undefined* if you set conflicting flags, like **+A-M** or **+S-S-S+S** etc. Entering *no* attributes will report the current ones.

```

[Fri, 2006/08/11, 21:13:54] (16096 bytes free)
LOC::M:\ >attrib Sys#rpcs.img
Sys#rpcs.img ----M

[Fri, 2006/08/11, 21:14:03] (12944 bytes free)
LOC::M:\ >attrib Sys#rpcs.img +h -m +r +s
Sys#rpcs.img RHS--

[Fri, 2006/08/11, 21:14:41] (12944 bytes free)
LOC::M:\ >attrib Sys#rpcs.img +a -h -r -s
Sys#rpcs.img ----M

```

- **BATT** [**<type>**]

Displays information on the battery/power supply. If you specify the optional 'type' parameter, then you can set how the organiser interprets the battery type:

- 1 – alkaline
- 2 – NiCad, 600mAh
- 3 – NiCad, 1000mAh (or NiMh?)
- 4 – NiCad, 500mAh

```
l-r-i, 2006/08/11, 21:15:25] (12944 bytes free)
LOC::M:\ >batt
Battery status:
  Main battery      = Low (replace soon, or plug in external adapter)
  Currently         = Low
  Current worst     = Low
  Battery type      = NiCad (1000 MAH; also use for NiMh)
  Time on batt.     = 4h, 8m, 22s
  Time inserted     = 2006/08/07, 23:36:12
  Backup battery    = Status unknown

  External supply   = Connected
  Time on ext       = 59h, 23m, 21s

Total consumption = 112 MAh
```

- **CD** **<dir>**

Change to the directory specified, which may be an absolute path:

```
CD LOC::M:\APP
```

or relative:

```
CD APP      (assuming you are currently in the root directory)
```

You can specify certain locations on the current device as follows:

'..' or '^' specifies the parent directory.

'\ ' or '\$' specifies the root directory.

You *must* provide these special specifiers on their own', therefore the following is valid:

```
CD \
CD app
```

but the following is *not* valid:

```
CD \app
```

```
l-r-i, 2006/08/11, 21:15:56] (12848 bytes free)
LOC::M:\ >cd app

l-r-i, 2006/08/11, 21:16:00] (12944 bytes free)
LOC::M:\APP\ >cd ..

l-r-i, 2006/08/11, 21:16:13] (12944 bytes free)
LOC::M:\ >
```

- **CLS**

Clears the screen.

- **COPY** **<source>** **<dest>**

This will copy the specified file. Names can be relative or absolute.

You will be prompted if the copy would cause a file to be overwritten.

```
l-r-i, 2006/08/11, 21:20:10] (32368 bytes free)
LOC::M:\ >cd rem::

l-r-i, 2006/08/11, 21:20:14] (32368 bytes free)
REM::C:\ >cd rickmisc\wwsite\software\pb2

l-r-i, 2006/08/11, 21:20:32] (32320 bytes free)
REM::C:\RICKMISC\WWSITE\SOFTWARE\PB2\ >copy bricks.opa loc::m:\app\bricks.opa
File copied from "REM::C:\RICKMISC\WWSITE\SOFTWARE\PB2\bricks.opa"
to "loc::m:\app\bricks.opa"
```

- **DEL** **<filespec>**



Delete the specified file. You will be prompted before actually deleting.

Note that you cannot use this command to delete directories (use Rmdir for that).

- **DUMP <filespec>**

This will 'dump' the file specified in hex and ASCII, the layout being familiar to anybody who has used a debugger.

Output will pause awaiting a key press after every screenful.

```

[Fri, 2006/08/11, 21:22:48] (12912 bytes free)
LOC::M:\AGN\ >dump diary.agn

00000 : 41 67 65 6E 64 61 46 69 6C 65 54 79 70 65 2A 00 : AgendaFileType*.
00010 : 0F 10 20 00 00 00 00 00 00 00 00 00 00 00 00 00 : ..
00020 : 2A 90 FF 00 54 6F 2D 64 6F 20 6C 69 73 74 00 00 : *É.To-do list..
00030 : 00 00 00 00 00 00 00 E1 0A 68 01 00 54 00 00 1C 02 : .....B.h..T....
00040 : 01 02 00 00 00 00 00 00 00 00 00 00 1F 10 31 34 : .....14
00050 : 38 04 1A 44 1E 00 00 15 4C 69 7A 7A 69 65 20 4D : 8..D....Lizzie M
00060 : 63 47 75 69 72 65 20 28 42 42 43 31 29 0B 50 01 : cGuire (BBC1).P.
00070 : 00 8C 34 01 0E 00 4C 00 00 00 1F 10 31 34 1C 02 : .i4...L.....14..
00080 : 1E 45 B4 00 01 15 45 6E 56 6F 6C 20 2D 20 46 72 : .Ej...EnVol - Fr
00090 : 65 6E 63 68 20 63 6F 75 72 73 65 1A 10 34 34 48 : ench course..44H
000A0 : 03 1E 82 96 00 01 10 45 6E 56 6F 6C 20 6C 27 82 : ..éú...EnVol l'é
000B0 : 63 72 69 74 75 72 65 20 10 3B 34 2A 03 17 41 3C : criture .i4*..A<

[Paused, press SPACE to continue or ^C to abort]

```

- **EXIT**

- **QUIT**

Close down *RShell*.

If you think you may need it again later, remember that your organiser multitasks so you can use the application keys underneath the screen to switch to a different task, and come back to *RShell* later...

- **INFO <spec>**

If the object specified is a file, this will display information on the file:

File size, attributes, modification time.

If the object specified is parsed as a device (i.e. no "\"), this will display device information:

Device type, total/free space, battery status, etc.

Info on a file:

```

[Fri, 2006/08/11, 21:24:39] (15328 bytes free)
LOC::M:\WRD\ >info thcfilms.wrd
Information on LOC::M:\WRD\Thcfilms.wrd
Object name = LOC::M:\WRD\Thcfilms.wrd
Object type = File
Object size = 2566 bytes
Is Read Only? No
Is Hidden? No
Is 'System'? No
Is Modified? Yes
Date stamp = 2006/08/10, 00:37:14

```

Info on a device...

```

[Sun, 2006/08/13, 21:46:31] (12720 bytes free)
LOC::M:\ >info m:
Information on device m:
Device type = RAM
Properties = formattable system-internal dynamically-sizeable
Removeable? = No
Capacity = 113152 bytes
Free space = 0 bytes
Volume name = ""
Battery = Status is not available.

```

- **KILL <PID>**

**!** This kills off the task specified *without confirming*. It is always better to try to exit a task cleanly... *You are not permitted to kill the system tasks* (or any task with a '\$' in the first part of the task name) because doing so will invariably mess up the functioning of your organiser – consider it like having a car and arbitrarily pulling out the brakes...

- **MEM**

Provides some basic information on your organiser's memory.

```

[Fri, 2006/08/11, 21:27:49] (36672 bytes free)
LOC::M:\WRD\ >mem
Memory statistics:

```

Device	Total	Free
Installed memory	262144 ( 256Kb)	34368 ( 33Kb)
Internal disc 'M:'	139520 ( 136Kb)	28672 ( 28Kb) dynamic
Left SSD 'A:'	Device is not present	or not responding.
Right SSD 'B:'	Device is not present	or not responding.
Expansion port 'C:'	131072 ( 128Kb)	52930 ( 51Kb)

- **MKDIR <name>**

Makes a new directory.

For an example picture, refer to *RmDir*.

- **RESUME <PID>**

! Resumes a task (presumably marked as **SUSPENDED**) *without confirming*.

- **RMDIR <spec>**

! Removes the specified directory.

You cannot delete directories which contain files, delete the files first.

```
[Fri, 2006/08/11, 21:28:33] (13520 bytes free)
LOC:M:\WRD\ >mkdir mystuff

[Fri, 2006/08/11, 21:28:35] (13520 bytes free)
LOC:M:\WRD\ >.
Thcfilms.wrd Vampire.wrd Mirbrisé.wrd Bugs.wrd Ideas.wrd MYSTUFF

11 objects listed: 10 files (34091 bytes), and 1 directory.

[Fri, 2006/08/11, 21:28:39] (13520 bytes free)
LOC:M:\WRD\ >rmdir mystuff
NOTE - you cannot remove a directory until it is empty.
Remove directory "LOC:M:\WRD\mystuff", sure? (y/N) Yes
Directory removed.
```

- **SUSPEND <PID>**

! Suspends a task *without confirming*. This command follows the same rules as **KILL**. Note that many tasks are *normally* in a suspended state due to the efficient multitasking.

- **TASKS**

This lists the tasks running on your organiser, and displays information on them.

```
[Fri, 2006/08/11, 21:28:49] (13776 bytes free)
LOC:M:\ >tasks
1 : SYS$NULL.#01 1138 [8472] suspended RAM code Inactive
2 : SYS$MANG.#02 1200 [84B0] suspended RAM code Inactive
3 : SYS$FSRV.#03 1262 [84EE] suspended RAM code Inactive
4 : SYS$WSRV.#04 1324 [852C] waiting[S] RAM code Inactive
5 : SYS$SHLL.#05 1386 [856A] waiting[S] RAM code Inactive
6 : TIME.#06 1448 [85A8] ready ROM code Active
7 : LINK.#07 21990 [85E6] ready ROM code Inactive
8 : SYS$NCP.#08 17956 [8624] suspended RAM code Inactive
9 : SYS$RFSV.#09 22114 [8662] waiting[S] RAM code Inactive
10 : SYS$RPCS.#10 30368 [86A0] waiting[S] RAM code Inactive
11 : R$HELL.#11 22238 [86DE] suspended RAM code Inactive
```

On the left is the process name.

This is followed by the PID (Program ID) that you should use to refer to the task during **SUSPEND**, **RESUME**, or **KILL**.

The following number, in hex, is the address of the task information block.

This is followed by the state of the task: *suspended*, *waiting*, or *ready*.

This is followed by whether the application is executing code in RAM or ROM.

Finally, if the application is *active* or *inactive*.

Due to the efficient multitasking, you may well find that many tasks (except "*Time*") are inactive and most likely suspended... including *RShell* itself.



## Licence

**This software, and source, are copyright © 2006 Rick Murray.**

You may freely use this software provided that all copyright attributions are preserved.

You are *not* permitted to release modified versions of this software without my permission. You may, however, distribute the original provided that you are distributing an exact duplicate of that which is available from my website. Please, no poking in "*file\_id.diz*" and the like.

If you include this software on the likes of a compilation CD, I would appreciate a gratis copy of that CD in return (doesn't have to be a fancy copy, just burn off a cheap CD-R, so long as it is readable, it'll do...).

***This software is provided as-is with no guarantees. No liability will be accepted for any loss or damage, howsoever caused. You use this software entirely at your own risk.***

There is no support offered on this software. However, generally I will endeavour to correct any issues raised if you send me a polite and descriptive email (note my limited access to the Internet, below).

This software is *entirely* my own work. I wish to credit and give thanks to those people who have released on-line the OPL documentation, as well as the collection of OPL examples, plus Psionics for the system calls documentation as well as brief by useful information on the kernel memory structures.

## My organiser's specification

My organiser is an *Acorn PocketBook II*. This is, essentially, a rebadged *3a* with slightly different software provided as-standard. Most things are the same, however, *Word* is called *Write* (but it is "Word" in the task list!) and *Agenda* is called *Schedule* (I've renamed the default file to call it "*Diary*" anyway)... that sort of thing.

The system memory is a mere 256K (which has all-too-quickly reached a state of being mostly-full!), slightly alleviated once a friend loaned me a 128K SRAM SSD.

I live in France so the default time zone is CET (UTC +1 (Winter) or +2 (Summer)), not that this should matter? My preferred time format is 24hr, date is YYYY/MM/DD. Numbers are in the format "12 345,67". Again, this shouldn't matter...

## Contact details

Email: **heyrick1973 at yahoo dot co dot uk**

Please note that I am only (usually) able to access the internet for half an hour a week at my local library, so my responses will not be as quick as I'd like.

Website: **<http://www.heyrick.co.uk/software/pb2/>**

## Thank you

...to mom, John & Irene, Frédérique, Anne, Françoise B, Sandrine, Françoise, Tiphaine, Bob & Barbara, Glenn, Ewen, Psionics, Google, and various Series 3a data repositories still around a decade later.

Music supplied by *Classic FM TV* (SkyDigital #359), including: *Catherine Jenkins*, *Trinity*, *Hayley Westenra*, *Andrea Bocelli*, *Adiemus*, *Lara St. John*, *Bond*, *Enya*, *Amici Forever*, *Ennio Morricone*, and some others both good and horribly screechy. :-)

Evidently I'm now one of 9 million people (figure from the channel's own promo advert) who probably find most of *CFMTV*'s output to be a heck of a lot more restful than that so-called music they broadcast on "*ChartShow TV*" (SkyDigital #355).

Right now as I write this, *CSTV* is playing "*God This Hurts*" by *Betty Curse* (and what is she wearing? that's weirder than Kelly Osbourne's normal attire!) while *CFMTV* is playing *Enya*'s "*Amarantine*". Nuff said, really!