

COMPILER PATHS & STANDARD LIBRARY INFORMATION

You ***MUST*** have a valid PSYQ_PATH environment variable. This variable should point to the directory containing PSYQ.INI

If you use CCPSX to compile you no longer require the environment variables for compiler paths and library files but you must put this information into your PSYQ.INI file instead. Please check this PSYQ.INI file now as your compiler and library paths may differ from ours.

THE NEW MESSAGE WINDOW for PSX printf() output

Some developers complained that printf() output received by DBUGPSX was too slow and had too small a buffer. To fix this I have implemented a character based (rather than block) message stream which is much faster and makes better use of buffer space. The new system also allows you to override the default buffer size.

Unlike the original message handler this version is designed to handle single character I/O as produced by PSX printf().

Because it is designed for a single character stream it makes much more efficient use of buffer space. A full 4K characters can be buffered by default. Other buffer sizes can be specified on the command line up to a max of 65534 bytes (but see notes below).

The debugger now auto-detects which message system is being used and takes input from either the new one or the old one.

To use the printf() facility you must have installed a PSX STDOUT handler on the PC. The standard STDOUT handler is MESS1.COM. There are no longer default message facilities in DEXBIOS or PSYBIOS. By separating this into two programs we can make more efficient use of PC memory. If you install DEXBIOS but do not install MESS?.COM then stdout from PSX will be discarded by the PC.

To confirm that new stdout to PC is working correctly you can execute the following:-

```
RESETPS 1
RUN MAIN
TESTMESS
```

If all is working correctly you should see the text output from the main program scrolling up your PC screen. Press a key to exit.

That's all there is to it. You now have a larger buffer and a snappier response for your printf() output.

Additional notes:

=====

Note that for message support you must have the second TSR MESS1.COM installed in your PC.

MESS1 allows an optional command line parameter to specify buffer size. If no size is specified then the default is to use a 4K buffer. Be warned that if you increase the buffer size then you will consume more DOS memory.

If you use QEMM then you should re-run the QEMM OPTIMIZE program if you update the TSRs or adjust the buffer size.

PSYBIOS no longer contains message buffering code. Instead it will detect MESS1 and forward messages to that program if it is installed.

The combined size of MESS1 and PSYBIOS is actually less than the old PSYBIOS.

MESS1 is optional. If you do not install MESS1 then PSYBIOS will still function correctly but it will discard any messages received from the target. MESS1 is not required for normal PC fileserver operation.

DEBUGPSX now does not poll for messages unless there is a message window on the screen.

DEXBIOS 1.22 now does not check for DOS re-entrancy on message writes as this is only necessary for genuine fileserver writes. This means that your PSX printf() output does not slow down if Brief is running.

For help on the DELAY program enter DELAY /?

DELAY program for DOS (from SN Systems Ltd)

Syntax:

delay <count>

Count is seconds/18, default is .5 seconds.

MESS3.COM - message handler to divert output to PC serial port

=====

This alternative version may be handy if you have a spare laptop PC or Unix workstation (!?) sat around near you development machine. It also works quite well with a Psion Series 3 + serial adapter. It sounds a bit silly to use another machine just for character output but there has been quite a demand for this as it means you can see the printf() output even if you aren't running the PSX debugger.

MESS3 works just like MESS1 but instead of making message output available to the debugger it sends the messages back out of the serial port. Unlike the old MW3 serial output this method will not hold up the PSX whilst serial output is happening because the characters are buffered by MESS3 and sent out of the PC serial port in the background using a serial port interrupt.

Any strange individuals who are considering tying together 2 PC serial ports and using Windows terminal to monitor PSX output may prefer to wait for the upcoming Windows version of MESS1 which will not require any serial ports.

Note that if PSX printf() sends data faster than the serial port can transfer it then the PC buffer will fill up and eventually you will lose characters when it overflows. This problem is more noticeable with slower baud-rate settings.

Default settings (i.e. MESS3 with no parameters) are to use a 4K buffer and direct output to COM2 (IRQ 3) at 9600 Baud.

Psy-Q RS232 output Message Server TSR ver 3.01

Syntax:

mess1 <switches> <buffsize> <switches>

Buffsize is in Kbytes, default is 4K

valid switches are:-

b## set baud rate (first 2 characters of baud rate)

c# set COM port (1,2,3 or 4)

i# override default IRQ number (0 to 7)

e.g. MESS3 /b19200 /c1

Sets the TSR to re-direct output to COM1 at 19200 baud.

Note that the baud rate specification needs only specify the first two characters - the rest are ignored. Also note that if you have a serial port with a non-standard IRQ (as long as it is 0 to 7) you can also override the default IRQ setting for that COM port.

e.g. MESS3 /b11 /c3 /i5

causes output to be sent to COM3 (i.e. port address 3E8) at 115 Kbaud using IRQ 5. A standard COM3 would have used IRQ 4 (same as COM1).

IF IT DOESN'T WORK

=====

1) if you do not get any message text at all then check:-

a) that you haven't run the NOPRINT module fro SCEJ

b) that you really have installed the correct version of DEXBIOS and that you have the MESS1 tsr loaded.

c) If you install PATCHX in a batch file then see next section.

2) If it works but is slow (18cps) then you have an IRQ problem:-

a) you have not specified the DTL-H interrupt on DEXBIOS cmdline

or b) the DTL-H IRQ is in use by another card or IRQ handler.

or c) you have not set the IRQ jumper on the DTL-H card correctly

or d) on a PCI local-bus PC you may need to use BIOS set-up program to select the IRQ as for ISA bus rather than PCI

or e) the IRQ on your DTL-H board is faulty.

Note: Please try MESS1.COM first as this will allow you to be sure that the PSX side of the set-up is working. MESS3.COM also requires that you specify the serial parameters correctly.

PROBLEMS USING PATCHX

=====

This is really nothing to do with this update but I'm mentioning it here because printf() can show the symptoms of this problem:-

The PATCHX kernel update may be unstable for a very short while after it is installed. Even though PATCHX may have appear to have completed execution OK, if you connect to the PSX during this short period of instability it seems that certain kernel functions may not initialise correctly. Printf() output is one of the things which may not work if the kernel does not start-up undisturbed.

This problem seems to be caused by cache coherency problems in the current version of PATCHX. Some people have attempted to insert delays into their batch file to work around the problem. This is flaky at best and a better solution is to obtain the SNPATCH replacement for PATCHX from SONY. SNPATCH has the added advantage that it updates your debug stub code (to 4.03) at the same time. Note that 4.03 is necessary if you wish to have a polhost call in a callback.