



**HUNT ENGINEERING**  
Chestnut Court, Burton Row,  
Brent Knoll, Somerset, TA9 4BP, UK  
Tel: (+44) (0)1278 760188,  
Fax: (+44) (0)1278 760199,  
Email: [sales@hunteng.co.uk](mailto:sales@hunteng.co.uk)  
<http://www.hunteng.co.uk>  
<http://www.hunt-dsp.com>



# ***HUNT ENGINEERING***

## ***Writes API Example***

### ***For LINUX***

***Document Rev A***  
***API Writes Example Rev 1.1***  
***J.Thie 07-12-05***

## **COPYRIGHT**

This documentation and the product it is supplied with are Copyright HUNT ENGINEERING 1999. All rights reserved. HUNT ENGINEERING maintains a policy of continual product development and hence reserves the right to change product specification without prior warning.

## **WARRANTIES LIABILITY and INDEMNITIES**

HUNT ENGINEERING warrants the hardware to be free from defects in the material and workmanship for 12 months from the date of purchase. Product returned under the terms of the warranty must be returned carriage paid to the main offices of HUNT ENGINEERING situated at BRENT KNOLL Somerset UK, the product will be repaired or replaced at the discretion of HUNT ENGINEERING.

Exclusions - If HUNT ENGINEERING decides that there is any evidence of electrical or mechanical abuse to the hardware, then the customer shall have no recourse to HUNT ENGINEERING or its agents. In such circumstances HUNT ENGINEERING may at its discretion offer to repair the hardware and charge for that repair.

Limitations of Liability - HUNT ENGINEERING makes no warranty as to the fitness of the product for any particular purpose. In no event shall HUNT ENGINEERING'S liability related to the product exceed the purchase fee actually paid by you for the product. Neither HUNT ENGINEERING nor its suppliers shall in any event be liable for any indirect, consequential or financial damages caused by the delivery, use or performance of this product.

Because some states do not allow the exclusion or limitation of incidental or consequential damages or limitation on how long an implied warranty lasts, the above limitations may not apply to you.

## **TECHNICAL SUPPORT**

Technical support for HUNT ENGINEERING products should first be obtained from the comprehensive Support section [www.hunteng.co.uk/support/support.htm](http://www.hunteng.co.uk/support/support.htm) on the HUNT ENGINEERING web site. This includes FAQs, latest product, software and documentation updates etc. Or contact your local supplier - if you are unsure of details please refer to [www.hunteng.co.uk](http://www.hunteng.co.uk) for the list of current re-sellers.

HUNT ENGINEERING technical support can be contacted by emailing [support@hunteng.demon.co.uk](mailto:support@hunteng.demon.co.uk), calling the direct support telephone number +44 (0)1278 760775, or by calling the general number +44 (0)1278 760188 and choosing the technical support option.

<b>THE WRITES EXAMPLE .....</b>	<b>4</b>
<b>COMPILING, LINKING AND RUNNING THE EXAMPLE .....</b>	<b>5</b>
COMPILING/LINKING THE EXAMPLE .....	5
RUNNING THE EXAMPLE .....	5
<b>MAKEFILE .....</b>	<b>6</b>
THE MAKEFILE .....	6
INCLUDE FILE .....	6
LIBRARIES .....	6
COMPILE PARAMETERS .....	6
<b>TECHNICAL SUPPORT .....</b>	<b>7</b>

## The writes example

---

The writes example is an example program that tests the PCI FIFO and HSB interface of a HERON carrier board. The example will try to boot a small and simple program onto the processor on a DSP module, such as a HERON4 or HERON2. With the HEPC8, the module must be in slot 1, for the HEPC9 and the HERON-BASE2 any slot is ok – use the /sx (x=1,2,3,4) option to identify the slot you wish to use. This program will absorb a stream of data from the host, sent via the PCI interface. The host example program will send the stream of data and tell you if everything worked or not. It will also give a very rough estimate of the transfer speed.

(This example will **not** work with TIM-40 carrier boards such as the HEPC2E, HEPC3, HEPC4 or HEPC11. It will also **not** work with the HEPC6, a one 'C6x processor board.)

## Compiling, linking and running the example

---

### Compiling/Linking the Example

To compile and link the example, please use the ‘makefile’ that is present in this directory. This makefile is set-up to use the GNU C/C++ 32-bit compiler. To execute the ‘makefile’, go to the ‘linux’ sub-directory and type:

```
make
```

### Running the example

To run the example, in the ‘linux’ sub-directory, type:

```
writes hep8a 0 a 1000 1000 10000
```

for HEPC8, or

```
writes hep9a 0 a 1000 1000 10000 /s3
```

for HEPC9 (assuming a module in slot 3), or

```
writes heb2a 0 a 1000 1000 10000
```

```
writes heb2a 0 b 1000 1000 10000 /s2
```

for HERON-BASE2 (module in slot 1 and 2, respectively).

You should see something like:

```
Start at 1000, inc 1000, end at 10000, BlockSize=250 on hep8a (0: Comporta)
Resetting...
Serial bus: slot 1: HERON1-C6201, rom version 4.
Resetting...
Booting /home/johan/api/etc/c6x/examples/writes/writes.out...
Testing...
Writes Transfer size 1000 DWORDS in 1 ticks, Speed: 3906.25 KBytes/sec
Writes Transfer size 2000 DWORDS in 1 ticks, Speed: 7812.50 KBytes/sec
Writes Transfer size 3000 DWORDS in 1 ticks, Speed: 11718.75 KBytes/sec
Writes Transfer size 4000 DWORDS in 1 ticks, Speed: 15625.00 KBytes/sec
Writes Transfer size 5000 DWORDS in 1 ticks, Speed: 19531.25 KBytes/sec
Writes Transfer size 6000 DWORDS in 1 ticks, Speed: 23437.50 KBytes/sec
Writes Transfer size 7000 DWORDS in 1 ticks, Speed: 27343.75 KBytes/sec
Writes Transfer size 8000 DWORDS in 1 ticks, Speed: 31250.00 KBytes/sec
Writes Transfer size 9000 DWORDS in 1 ticks, Speed: 35156.25 KBytes/sec
Writes Transfer size 10000 DWORDS in 1 ticks, Speed: 39062.5 KBytes/sec
Check whether any interrupts were used: read 0, write 0, master mode 0.
```

If you have any other response than this, and you have a HEPC8 or HEPC9, please first test if the ‘testint’ example works. If this example doesn’t work as well, there is likely an interrupt problem. With the HEPC8, check the ‘routing jumpers’ on the HERON module in slot 1. These jumpers need to be set to select ‘FIFO 0’ for both the ‘in’ and ‘out’ FIFO. Please refer to the ‘Troubleshooting’ section in the LINUX Installation & User Manual.

## The Makefile

What changes need to be made to a ‘standard’ LINUX makefile? This section will explain what needs to be changed (or added) in a makefile to compile/link successfully the Hunt Engineering API programs

## Include file

All Hunt Engineering API programs must include ‘heapi.h’. This file is located in directory ‘/usr/local/include’. The ‘installme’ script should have copied it there. If this hasn’t happened, or if you wish to install this file manually, it can be copied from the API installation directory’s ‘include’ sub-directory into ‘/usr/local/include’. The compiler automatically searches in the local include directory as well; you shouldn’t need to have to add it to the makefile.

## Libraries

The Hunt Engineering API is delivered as a shared library (‘libhel.so’). This file is located in directory ‘/usr/local/lib’. The ‘installme’ script should have copied it there. If this hasn’t happened, or if you wish to install this file manually, it can be copied from the API installation directory’s ‘lib’ sub-directory into ‘/usr/local/lib’. It must be linked before other (GNU) libraries. Example:

```
writes: main.o cload.o
    $(CC) $(CFLAGS) main.o cload.o -o writes
        /usr/local/lib/libhel.so /usr/lib/librt.so
```

The bold italic part is the part added by us.

For the HEPC9, we need to use HeartConf. This is part of the Server/Loader library. In the makefile the Server/Loader library ‘liblinuxsl.so’ should be added. The Server/Loader uses ‘hrn\_fpga’ (the FPGA programmer) and that library needs to be linked in as well.

```
reads: main.o cload.o
    $(CC) $(CFLAGS) main.o cload.o -o reads
        /usr/local/lib/liblinuxsl.so
        /usr/local/lib/libhrnfpga.so
        /usr/local/lib/libhel.so /usr/lib/librt.so
```

## Compile Parameters

The Hunt Engineering API supports several different types of Operating Systems. To select LINUX support, you need to #define a variable ‘\_LINUX’. The easiest way to do this is in the makefile.

```
CFLAGS=-O2 -Wall -D_LINUX=1
```

The bold italic part is the part added by us.

1. Technical support for HUNT ENGINEERING products should first be obtained from the comprehensive Support section [www.hunteng.co.uk/support/index.htm](http://www.hunteng.co.uk/support/index.htm) on the HUNT ENGINEERING web site. This includes FAQs, latest product, software and documentation updates etc. Or contact your local supplier - if you are unsure of details please refer to [www.hunteng.co.uk](http://www.hunteng.co.uk) for the list of current re-sellers.
2. HUNT ENGINEERING technical support can be contacted by emailing [support@hunteng.co.uk](mailto:support@hunteng.co.uk), calling the direct support telephone number +44 (0)1278 760775, or by calling the general number +44 (0)1278 760188 and choosing the technical support option.