

Applied Data Sciences / PCEHSD® Software Specifications

2.1 Subroutine Calls

There are two (2) sets of subroutine calls used for controlling the PCEHSD, where each set consists of six (6) separate subroutines. The HSDOPEN routine is to be used when the PCEHSD is configured for the HSD mode and IBLOPEN when operating in IBL mode. The other entry points (e.g. IBLCLOS/HSDCLOS) are simply alternate names for the same routine; either entry point may be used. Two additional utility routines are provided for translating addresses from segment:offset form (far pointers) to the 32-bit physical addresses required by the PCEHSD.

These routines assume Large or Huge memory model; all calls are far calls and all addresses passed as parameters are assumed to be far pointers (segment:offset) form.

NOTE: If calling these routines from a language other than "C", the specific provisions for mixed-language subroutine calls (e.g. the INTERFACE statement in Microsoft Fortran) should be used.

2.2 HSDOPEN/IBLOPEN Routine

The HSDOPEN or IBLOPEN routine must be called first to install the driver in the MS-DOS interrupt structure and initialize some necessary data structures. The appropriate OPEN routine must be called, depending on whether the PCEHSD is currently configured for HSD or IBL mode. The user passes a File Control Block (FCB) address to the subroutine. This data structure is described in SECTION 3 and in the example programs in APPENDIX A & B. Up to four PCEHSD's may be opened simultaneously, provided the user supplies a unique FCB for each. The user must have filled in set-up information in the FCB. A typical calling sequence would be:

```
HSDOPEN(&fcb_blk,&cerr);
```

```
IBLOPEN(&fcb_blk,&cerr);
```

fcblblk is a File Control Block structure which contains status, set-up information and a pointer to the IOCB list.

cerr is an integer error return status code with the following meanings:

- 0 - successful open
- 1 - device not present.
- 2 - too many PCEHSD's already opened.
- 3 - can't read EISA configuration data.
- 4 - wrong OPEN routine for current PCEHSD mode.
- 5 - invalid EISA slot number.
- 6 - File Control Block already open.

(NOTE: The device inoperable and error code flags are set in the FCB status code if the PCEHSD is determined to not be present or if the HSD device is not connected.)

2.3 HSDSTRT/IBLSTRT Routine

The HSDSTRT or IBLSTRT routine is used to start a data transfer via the PCEHSD. HSDSTRT and IBLSTRT are alternate names for the same subroutine; either one may be used. The address of an FCB containing the address of an IOCB list is passed to this subroutine. In the NOWAIT mode, once the transfer is started, control is passed back to the calling routine. In the WAIT mode, the HSDSTRT routine waits for the IOCB list execution to finish. In the NOWAIT mode the calling routine determines when the transfer is complete by monitoring the device busy bit in the FCB status word 3 (1=busy; 0=not busy). A typical calling sequence would be.

```
HSDSTRT(&fcblblk,&cerr);
```

```
IBLSTRT(&fcblblk,&cerr);
```

fcblnk is a File Control Block structure which contains status, set-up information and a pointer to the IOCB list.

cerr is an integer error return status code with the following meanings:

0 - Successful start.

1 - FCB not open.

2 - Device busy.

5 - Timeout

2.4 HSDTEST/IBLTEST Routine

The HSDTEST/IBLTEST routine is used to wait for completion of a NOWAIT mode operation started by HSDSTRT/IBLSTRT, while observing the timeout imposed by the FCB setup. Its main use is to allow a program to begin a data transfer operation, perform other processing in parallel and later to check the status of the data transfer. A typical calling sequence would be:

```
HSDTEST(&fcblnk,&cerr);
```

```
IBLTEST(&fcblnk,&cerr);
```

fcblnk is a File Control Block structure which was used to initiate the data transfer operation being checked.

cerr is an integer error return status code with the following meanings:

0 - Operation completed.

1 - FCB not open.

2 - I/O still in progress.

5 - Timeout

2.5 HSDTERM/IBLTERM Routine

The HSDTERM/IBLTERM routine is used to terminate any existing transfer in progress. A typical calling sequence would be:

```
HSDTERM(&fcb_blk, trm_cod, &cerr);
```

```
IBLTERM(&fcb_blk, trm_cod, &cerr);
```

fcb_blk is a File Control Block structure which contains status, set-up information and a pointer to the IOCB list.

trm_cod is a binary value defining the type of Reset being issued. There are six types of Reset (some of which may be combined) and have the current binary value.

1 = PCEHSD board Reset.

2 = PCEHSD FIFO Reset.

4 = External Device I/O Reset (Encore signal IOR).

7 = Terminate External Device transfer (Encore signal TDV)

8 = Power-Up Reset of PCEHSD firmware.

16 = Cancel IBL operation which is awaiting link.

Resets 1, 2 and 4 may be combined. If all three are desired, use the value 0 for trm_cod.

cerr is an integer error return status code with the following meanings:

0 - Successful termination.

1 - FCB not open.

2 - Link established; can't cancel operation (code 16 only)

3 - Bad trm_cod value.

5 - Timeout

2.6 HSDCTL/IBLCTL Routine

The HSDCTL/IBLCTL routine is used to control the PCEHSD directly to perform those functions not available through other I/O routines. The passed arguments are placed in the PCEHSD Device Command Block opcode and opcode-dependent fields without checking, the operation is initiated and control returns to the caller if NOWAIT mode was specified. If WAIT mode was specified, HSDCTL/IBLCTL will wait the designated time for an interrupt to occur, indicating completion of the operation. A typical calling sequence would be:

```
HSDCTL(&fcb_blk,op,aux,&cerr);
```

```
IBLCTL(&fcb_blk,op,aux,&cerr);
```

fcb_blk is a File Control Block structure which contains status, set-up information and a pointer to the IOCB list.

op is an integer containing the PCEHSD operation code to be performed.

aux is a longword containing the opcode-dependent information required for the PCEHSD operation to be performed.

cerr is an integer error return status code with the following meanings:

0 - operation complete.

1 - FCB not open.

2 - device busy.

5 - timeout

Note: The user should understand PCEHSD operations thoroughly before attempting to use this routine. Refer to the PCEHSD Technical named, part number 0900045, for a detailed description of PCEHSD operation.

2.7 HSDCLOS/IBLCLOS Routine

The HSDCLOS/IBLCLOS routine is used to restore the MS-DOS interrupt structure back to its normal configuration. A typical calling sequence would be:

```
HSDCLOS(&fcb_blk,&cerr);
```

```
IBLCLOS(&fcb_blk,&cerr);
```

fcb_blk is the File Control Block structure which was used to open the PCEHSD.

cerr is an integer error return status code with the following meanings:

0 - successful close.

1 - FCB not open.

2 - device busy.

5 - timeout

2.8 SEG_TO_ABS Routine

The SEG_TO_ABS function is used to convert addresses in PC memory from the usual segment:offset form (Far Pointer) to the 32-bit physical addresses required by the PCEHSD. The user is responsible for supplying 32-bit physical addresses for data buffers and any IOCB's which is the target of a Transfer in Channel IOCB. A typical calling sequence would be:

```
unsigned long = SEG_TO_ABS(far char*);
```

```
phys_adrs = SEG_TO_ABS(&bfr);
```

&bfr is the address which is to be converted.

phys_adrs is the returned 32-bit physical address

2.9 ABS_TO_SEG Routine

The ABS_TO_SEG function is used to convert addresses in PC memory from the 32-bit physical addresses used by the PCEHSD to the segment:offset form used by the 80x86 family of processors in real mode. The IOCB addresses returned to the FCB by the PCEHSD are 32-bit physical addresses. This routine will convert such a physical address into a far pointer which a program can use to access data. A typical calling sequence would be:

```
far char* ABS_TO_SEG (unsigned long);
```

```
far_adrs = ABS_TO_SEB(phy_adrs);
```

phys_adrs is the 32-bit physical address to be converted.

far_adrs is the far pointer which is returned. If the physical address is outside the range of a 20-bit address (1 Megabyte), a NULL pointer will be returned.

3.1 Introduction

The PCEHSD card emulates the Encore HSDII card and thus all programming constraints are imposed by ensuring compatibility with the Encore HSDII card. The principal data structures used are the File Control Block (FCB) and the Input/Output Control Block (IOCB). The IOCB structure used adheres strictly to the Encore HSDII definition. The FCB used is structured to be efficient for use of the PCEHSD, but retains some similarity to the Encore File Control Block for historical reasons. The file "PCEDEV.H" contains C language structure definitions for the FCB and IOCB as used by the PCEHSD routines. In appendix A you will find an example program with listings of these structure definitions.

3.2 Reference Documents

Refer to the respective Encore MPX-32 Operating System Reference and Technical software manuals for a detailed description of the software requirements for the FCB and IOCB structure and the HSDII board operation. For a comprehensive hardware description of the Encore HSDII compatible card refer to

Encore HSDII Technical Manual, Document # 303-329131-000.

Encore MPX-32 Volume 2, Reference Manual, Document #323-001011-300

Refer to the following documents for programming assistance on the IBM compatible PCEHSD card.

IBM PC/AT Technical Reference Manual 1502494

IBM DOS Technical Reference Manual 6138536

ADS PCEHSD Technical Manual 0900045

Encore MPX-32 Reference Manual, Volume 1 323-001011-300

3.3 File Control Block

The File Control Block (FCB) provides transfer parameters and working storage areas for the device driver routines. The FCB must be set up by the user to describe certain attributes of each logical I/O operation. In addition, certain information collected during each I/O operation is made available to the user via the corresponding FCB.

For purposes of compatibility with PCHSD software available from Applied Data Sciences, the FCB used by the PCEHSD routines maintains the same symbolic names for FCB fields where possible. This should enable the PCHSD user to upgrade software to use the PCEHSD with a minimum of effort. Refer to APPENDIX C for a detailed list of PCHSD-to-PCEHSD conversion considerations.

The FCB format is described in detail in the following sections. The description uses the symbol names from the fcb_struct structure definition found in the header file PCEDEV.H.

3.3.1 PCEHSD FCB Addresses

The various addresses in the FCB used by the PCEHSD are listed below. Refer to the following sections to determine the contents of each field, what is required fill-in information and what is status. The first byte address is 0; contents of areas denoted reserved are undefined. Refer to the example program in APPENDIX A for additional clarification.

name offset byte description

(bytes)(length)

reserved 0 8

nowait 8 4 Wait/nowait [MSB (Encore bit 00)]:

1 = nowait,

0 = wait

sts_flg 12 2 Transfer status flags

mod_flg 14 2 Mode: image of Base Configuration Register

irq_lev 6 2 PCEHSD Interrupt Request Level

hsd_bas 18 2 Base address of PCEHSD I/O registers

sav_iocb 20 4 Address of IOCB causing most recent interrupt

old_vec 24 4 Saved DOS interrupt vector

devid 28 4 Firmware ID (4 ASCII characters)

iocl_ptr 32 4 Address of starting IOCB

tim_out 36 2 Time-out count (units of 0.1 seconds)

reserved 38 6

pchsd_stat 44 4 Not currently used

hsd_stat 48 4 Most recent HSD status

end_act 52 4 Normal end action subroutine

err_end 56 4 Error end action subroutine

spr_cnt 60 2 Count of spurious interrupts

reserved 62 2

port_sel 64 2 Byte/word swap select

slot_nbr 66 2 PCEHSD Slot Number

dcbptr 68 4 Pointer to Device Command Block

3.3.2 PCEHSD FCB Contents

nowait This is an unsigned long integer containing flag bits controlling the PCEHSD routines. The following bits are significant (hex values are shown):

80000000 I/O is to be done in no-wait mode (control returns to caller immediately upon starting I/O transfer).

00800000 Restart IOCB list when complete. This option is effective only in IBL mode and only if the list completes without error or by receipt of an External Terminate.

00400000 Do not attempt to reset PCEHSD on HSDOPEN or IBLOPEN call. Normally, if the board status cannot be read at open time, a reset will be

issued and another status request issued before the device is declared non-existent. This option prevents the OPEN routine from performing the reset.

00200000 This option enables the PCEHSD to detect interrupts from an external device and to call the user's error or end action routine on such occurrence.

sts_flg This word contains status flag bits returned by the PCEHSD routines. The following bits are significant (hex values are shown):

8000 Device Busy. Current I/O request is in progress. A program using No-Wait I/O should check this bit to determine completion of the request.

4000 Error. A device error has occurred. The PCEHSD status word in hsd_stat should be examined to determine the cause of the error.

2000 External Device Interrupt occurred. This bit is set when an External Interrupt is received if External Interrupts were enabled at OPEN time by setting the appropriate bit in the FCB nowait field.

1000 Open. This FCB is currently open and I/O operations to the corresponding PCEHSD are enabled.

400 External Mode Active. External mode was enabled by calling HSDCTL/IBLCTL to enable external mode. This bit is cleared when the connected device issues an external terminate and exits external mode.

mod_flg This unsigned integer contains a copy of the PCEHSD Base Configuration Register set up at power-up. This value is returned on an HSDOPEN/IBLOPEN call. See the PCEHSD Technical Manual, document number 0900045, for a detailed description of the contents.

irq_lev This integer contains the interrupt request level assigned to the PCEHSD by EISA configuration information. This value is returned on an HSDOPEN/IBLOPEN call.

hsd_bas This integer contains the base address in slot-specific EISA bus I/O space of the PCEHSD board. The value should be 1000 hex times the slot number. This value is returned on an HSDOPEN/IBLOPEN call.

sav_iocb This unsigned longword contains the 32-bit physical EISA bus address of the IOCB causing the last PCEHSD interrupt.

old_vec This field contains a far pointer (segment:offset format) to the interrupt service routine originally assigned to the PCEHSD's interrupt request level. The original vector is saved here by the HSDOPEN/IBLOPEN call and restored on a CLOSE.

devid This is a four-character array containing the firmware identification (revision level) for the PCEHSD. The format of this data is "VMnn"; where 'V' is an ASCII character, M is the hex major revision level, and nn is the hex

iocl_ptr This field is a far pointer to the first IOCB of a list to be executed. Note that in the fcb_struct definition, this is defined to be a pointer to a structure of type iocb_struct. This structure definition for the Encore IOCB is also found in the PCEDEV.H header file. conversion of the far pointer to a 32-bit physical address is performed by the PCEHSD routines.

tim_out This unsigned integer contains the user's desired time-out interval in units of 0.1 seconds. (i.e. a value of 300 represents 30 seconds.) A value of zero represents an infinite time-out interval.

hsd_stat This unsigned longword contains the last status stored in the Device Command Block by the PCEHSD.

end_act This field contains a far pointer to the caller's end action routine. This routine is called as a subroutine by the PCEHSD interrupt service routine, on an interrupt caused by 1) an IOCB with the "Interrupt on End of Block" bit set or 2) completion of the IOCB list with no errors.

err_act This field contains a far pointer to the caller's error end action routine. This routine is called as a subroutine to the PCEHSD interrupt service routine, on an interrupt caused by termination of the IOCB list due to error.

spr_cnt This unsigned integer contains a count of spurious interrupts detected while the FCB was open.

port_sel This field contains the additional bits to be merged with the Base Configuration Register for the desired PCEHSD operation. Normally, the only significant bits in this unsigned integer are the following (hex values are shown):

0001 Swap bytes in data transferred

0002 Swap words in data transferred

Any value in this word on a call to HSDCTL/IBLCTL will be copied to the Device Command Block and used verbatim. See the PCEHSD Technical Manual, document number 0900045 for detailed information on the use of the PCEHSD configuration register.

slot_nbr This integer contains the EISA bus slot number of the PCEHSD being accessed on a call to HSDOPEN/IBLOPEN.

dcbptr This field contains a far pointer to the Device Command Block (DCB) used internally by the PCEHSD I/O routines to control the device. The user may freely read the data in the DCB, but must be very cautious when altering its contents. See the PCEHSD Technical Manual, part number 0900045 for additional information about the PCEHSD Device Command Block.

3.3.3 PCEHSD FCB Required Fill-in

Prior to an HSDOPEN or IBLOPEN call the caller must fill in the FCB values for the following:

slot_nbr PCEHSD EISA bus slot number (1 through 15)

nowait Options on OPEN:

No-Reset (Hex value 00400000)

External Interrupt (Hex value 00200000)

end_act Normal end action subroutine address (required if External Interrupts enabled).

(see Appendix A for FCB "C" structure).

Prior to an HSDSTRT or IBLSTRT call the caller must fill in the following additional fields of the FCB.

nowait Wait/nowait [Encore bit 00 or PC bit 31 of a long word];

hex 00000000 = wait

hex 80000000 = nowait

iocl_ptr Starting IOCB list pointer

tim_out Time out count in 1\10 seconds (decimal value)

end_act Normal end action subroutine address (optional)

err_end Error end action subroutine address (optional)

port_sel Byte/word swap:

0000 = No swap

0001 = Byte swap

0002 = Word swap

0003 = Byte & Word swap

3.3.4 Status Information

Various status information is available to the user. This information is presented in the following places:

sts_flg Current state of I/O operation (busy, error)

mod_flg Contents of PCEHSD Base Configuration Register

irq_lev Interrupt level in use by PCEHSD.

hsd_bas PCEHSD base I/O address.

devid PCEHSD firmware revision level.

sav_iocb Most recently interrupting IOCB address

hsd_stat PCEHSD status word

3.4 Input/Output Control Block Required Fill-in

Prior to an HSDSTRT or IBLSTRT call the calling routine must first fill in the pointer to the IOCB list iocl_ptr. (Notice that this pointer must be a far pointer.)

The IOCB list is an array of structures which the calling routine must fill in. The normal HSD bit definitions are adhered to strictly. The following HSD functions, which are described by IOCB Word 1, opcode bits 00 - 07, are supported:

Bit 00 - HSD I/O Transfer.

Bit 01 - HSD Command Transfer.

Bit 02 - HSD Status Request.

Bit 04 - HSD Interrupt on End-of-Block (IEOB).

Bit 05 - HSD Transfer-In-Channel (TIC). Subtract one and branch non-zero.

Bit 06 - HSD Command Chain.

Bit 07 - HSD Data Chain.

Two additional restrictions apply to IOCB's used by the PCEHSD:

1. All addresses used by the PCEHSD are 32 bit physical EISA bus addresses. The user should provide 32 bit physical address for data buffers and for target IOCB addresses in TIC's.
2. All IOCB's must be doubleword aligned. That is, all IOCB addresses must be multiples of 4 bytes.

See section 2.8 & 2.9 information on routines to perform translation between far addresses (segment:offset form) and 32-bit physical addresses.

3.5 Special Features

There are some features of this program that are not required for basic PCEHSD operation. Information is presented in the following sections to help better describe these enhancements.

3.5.1 IBL Mode Special Features

Special features when the PCEHSD board is being used in the IBL mode are described in the following paragraphs.

In the FCB Status Flag area Bit 8 of Word 3 [or the most significant bit of byte 13 (i.e., Encore bit 08 or PC long word bit 23)] is used to indicate that an IOCB list should restart a sequence following completion of the current IOCB list.

In the first word of the first IOCB of a list, device dependent bit 14 has a special meaning. This is Encore bit 14 or PC bit 17.

Bit 14 - if bit 15 is set then a Link Request will be issued, otherwise the device will wait for a Link Request from the external device.

Note that this usage is compatible with the PCHSD; the bit specifying that a link must be established is unnecessary. If this PCHSD is not establishing the link, it will await a link request and acknowledge it. In any case, a link will be established once for each IOCB list executed in IBL mode.

3.5.2 External Device Interrupt

There are certain requirements that must be met if using the external device to interrupt the PCEHSD.

1. The External Device Interrupt is enabled by the HSDOPEN routine when the proper bit (hex00200000) is set in the JFCB nowait field. Refer to SECTION 2.2.
2. The External Device Interrupt is disabled when the HSDCLOS routine is called. Refer to SECTION 2.7.

Servicing of this external interrupt is handled differently if the PCEHSD is executing an IOCB, or if it is not.

If the applications program is not currently executing an IOCB, then when an External Device Interrupt is received, the File Control Block (FCB) Word 03, Bit 02 status bit will be set. This bit is described in SECTION 3.3.2. A check is then made to determine if a Normal End Action Routine Address is present in the File Control Block Word 13 (offset bytes 52 - 55). If no End Action Address is present then the External Device Interrupt is ignored. If an End Action Address is present, then the End Action routine is executed. The External Device Interrupt Status bit in the File Control Block Word 3, bit 02 is reset when the program returns from the End Action routine.

If the applications program is currently executing an IOCB, then when an External Device Interrupt is received, the next status stored by the PCEHSD will have the Link Request In (LRI) bit set and IOCB list execution will continue.

3.5.3 Normal End Action/Error End Action Routines

Normal End Action occurs when the PCEHSD completes the operation specified by an IOCB with the Interrupt-on-End-of-Block (IEOB) bit set, or with none of Data Chain, Command Chain, or Transfer-In-Channel set. If there is a Normal End Action address set in File Control Block Word 13, the routine at that address will be called.

Error End Action occurs when the PCEHSD determines that one of two conditions has occurred.

- 1 - The signal External Terminate (EXT) is set or
- 2 - Device-End-of-Block (DEOB) with Transfer Count not zero is set.

If there is an Error End Action address set in File Control Block Word 14, the routine at that address will be called.

An end action routine is called as a Far Procedure with a single parameter passed on the stack. The passed parameter is a Far Pointer to the FCB for the PCEHSD causing the end/error action call. The parameter is passed according to the C language calling convention (it is popped from the stack by the calling routine), so it may be ignored by the end action routine, if desired.

To set the File Control Block (FCB) end action address perform:

```
Declare as "void far ENDACT ();"
```

```
fcf.end_act = ENDACT;
```

NOTE: Refer to the File Control Block Structure. Also the name ENDACT is not required.

3.5.4 Operation in External Mode

When using the PCEHSD in external mode, the attached device takes control of the PCEHSD and commands reads and writes via the EISA bus independently of the host CPU. Since the EISA bus has a 32-bit wide address and the external device supplies only a 24-bit address, the PCEHSD uses a 32-bit absolute base address which it adds to addresses from the external device when accessing the bus.

To enable external mode, call HSDCTL/IBLCTL with opcode 4, Enable External Mode, and with the absolute base address for EISA bus reads and writes as the opcode-dependent data (aux). See section 2.6 for additional information. The FCB status (sts_flg) will show "External Mode Active" and "Device Busy". When the PCEHSD has processed the Enable External

Mode operation, "Device Busy" will be cleared and only "External Mode Active" will be set. To send an IOCB list to the external device, call HSDSTRT/IBLSTRT in the usual manner. The FCB status will again show "Device Busy" until the IOCB list has been processed and all statuses stored. If the external device issues an External Terminate and does not release external mode, the "External Device Interrupt" bit will be set in sts_flg and any declared end action routine will be called. When the external device issues an external terminate and releases external mode, the "external mode active" bit in sts_flg will be cleared. The PCEHSD should be completely reset by calling HSDTERM/IBLTERM with trm_cod set to 8 before using the PCEHSD for non-external-mode operations.