Odin TeleSystems Inc.

User Guide for THOR-2, Rev. 1.0

Doc. No. 1412-1-HAA-1004-1

Rev. 1.8 (Released)

June 26, 1997

Copyright

Trademarks

Changes

The material in this document is for information only and is subject to change without notice. While reasonable efforts have been made in the preparation of this document to assure its accuracy, Odin TeleSystems Inc., assumes no liability resulting from errors or omissions in this document, or from the use of the information contained herein.

Odin TeleSystems Inc. reserves the right to make changes in the product design without reservation and notification to its users.

Notices

This device has been tested and found to comply with the limits for a Class A digital device pursuant to part 15 of FCC rules. These limits are designed to provide reasonable protection against harmful interference when this equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if it not installed in accordance with the instruction, may cause harmful interference to radio communications. If the operation of this equipment causes harmful interference, the user will be required to correct the interference at his/her expense. This device may not be operated in a residential environment.

This equipment is intended for use in a laboratory test environment only. Federal regulations (FCC Part 68) prohibit connection of a DSX-1 device to the public network without an FCC approved Channel Service Unit (CSU) and protection circuitry.

UNDER NO CIRCUMSTANCE SHALL ODIN TELESYSTEMS INC., ITS OFFICERS, EMPLOYEES, OR AGENTS BE LIABLE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS, PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS PRODUCT AND ITS DOCUMENTATION, EVEN IF ODIN TELESYSTEMS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT WILL ODIN TELESYSTEMS' LIABILITY FOR ANY REASON EXCEED THE ACTUAL PRICE PAID FOR THE PRODUCT AND ITS DOCUMENTATION. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL AND CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY.

Odin TeleSystems Inc.

This document is published by:
Odin TeleSystems Inc.
P. O. Box 59686
Dallas, Texas 75229
U. S. A.

Printed in U. S. A.

1. Table of Contents

2. Introduction

This document describes the procedures to configure and install the Thor-2 board into a PC, to connect it to a T1 or E1 interface, to connect accessories to Thor-2, and to set up the on-board processor software development environment. The document also describes the configuration dip-switches and jumpers, the connectors and their pin assignments as well as the use of Thor-2 utility programs *T2Boot.exe* and *T2Config.exe*.

For more information on the technical capabilities of Thor-2, please refer to the *Thor-2 Technical Description* (Odin TeleSystems Inc. document number 1111-1-HAA-1004-1). For information on how to use these capabilities under software control, please refer to the *Thor Application Programming Interface (API) Reference Guide* (Odin TeleSystems Inc. document number 1211-1-SDA-1001-1).

3. How to Contact Us

For Technical Support, please contact:

Odin TeleSystems Inc.
800 E. Campbell Road, Suite 300
Richardson, Texas 75081
U. S. A.
Tel: +1-972-664-0100
Fax: +1-972-664-0855
Email: Support@OdinTS.com
URL: http://www.OdinTS.com

For Sales and Marketing, please contact:

Odin TeleSystems Inc.
P. O. Box 59686
Dallas, Texas 75229
U. S. A.
Tel: +1-972-664-0100
Fax: +1-972-664-0855
Email: Sales@OdinTS.com
URL: http://www.OdinTS.com

4. PC Requirements

It is recommended that Thor-2 is used in a PC which satisfies the following requirements:

- Industry Standard Architecture (ISA) Bus with at least one free 16-bit slot.
- An Intel 80386 processor. A 486, Pentium or Pentium Pro processor is recommended.
- At least 1 MByte of memory. 32 KBytes free upper memory area.
- 1 Free IRQ (3, 4, 5, 7, 10, 11, 12, or 15)
- At least 20 Free I/O -addresses

5. Configuring the Thor-2 Hardware

5.1 Handling Instructions

The Thor-2 board has been packaged in a sealed anti-static bag for protection during shipping and handling. Follow precautions regarding handling of electrical equipment while configuring the board and while installing it into a PC expansion slot. Be aware of the possibility of damage to the sensitive electrical devices on Thor-2 from static electricity discharge. Please wear anti-static protection devices such as a ground strap connected to a grounded equipment frame while handling the Thor-2 board. To configure the board, remove it from the anti-static bag and place it on to a flat and properly grounded anti-static mat.

5.2 General

The Thor-2 board is configured by setting jumpers (JP) and switches (SW) on the board. The locations and names of the jumpers, switches, and connectors used during configuration and installation can be found from Figure 1.

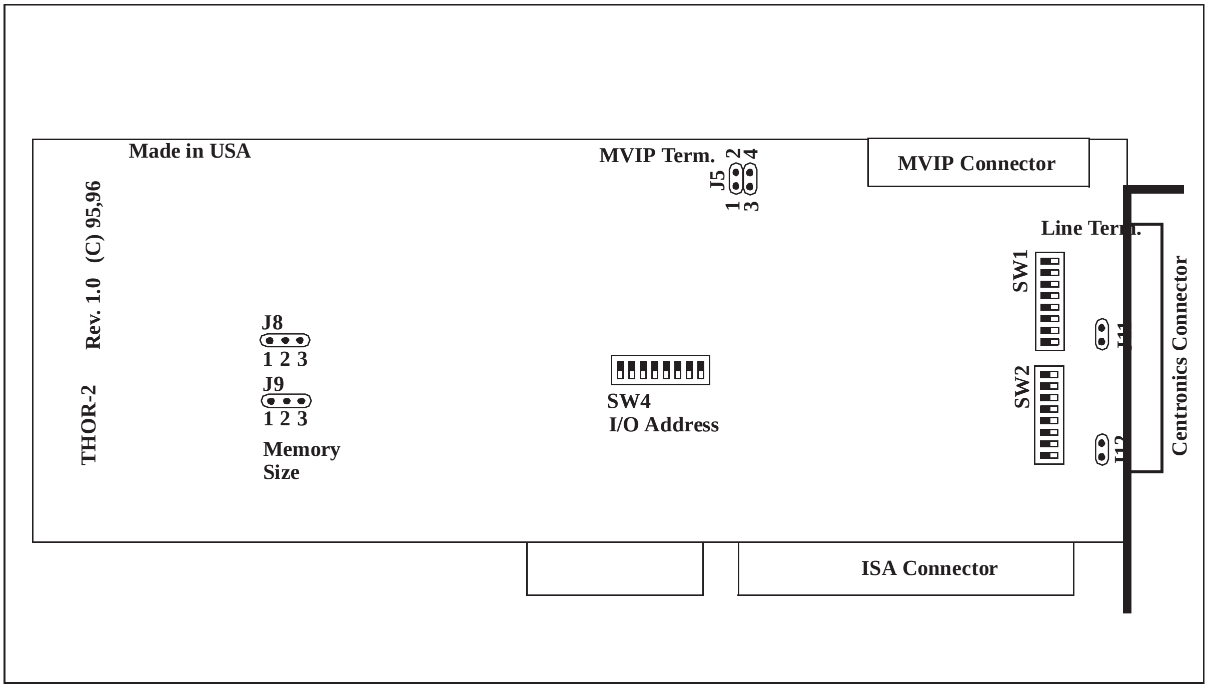


Figure 1. Thor-2 Connectors, Switches, and Jumpers

5.3 Setting the Board Number (Switch SW4)

Up to 4 Thor-2 boards can be placed in one PC. Each installed board has to be assigned a unique board number, which is set with switches 7 and 8 in the dip-switch SW4. Table 1 summarizes the switch SW4 settings for the board number.

TABLE 1. SW4 Board number settings

Board #	#7	#8
0	On	On
1	Off	On
2	On	Off
3	Off	Off

The boards should be numbered sequentially, starting from 0. If only one board is used, it should be numbered 0. If 2 board are used they should be numbered 0 and 1, etc.

5.4 Setting the I/O Base Address for Direct Access I/O-Registers (Switch SW4)

Thor-2 requires access to a 16 byte long free I/O address space for on-board I/O registers. The base address of the reserved area can be set to any even 16 byte boundary within the range 0000h - 03FFh. Since different PCs are equipped with different hardware, the I/O address area that should be used is PC specific. The factory default for the Thor-2 board is I/O base address 0280h; i.e., Thor-2 will use I/O ports 0280h through 028Fh (16 bytes). This needs to be changed if the PC already contains hardware (e.g. LAN adapters, drive controllers, or serial port boards) which use this address area. Consult the documentation for the corresponding hardware to determine what I/O addresses are free for use by the Thor-2 board.

If more than one Thor-2 boards are installed in one PC, the different boards should be set to the same I/O address. The board number switch setting then determines in which 1024 (400h) byte segment the board places it's I/O address space. For example, if a board is set to number 3 and I/O address 280h, then the board will occupy I/O addressed from (3 * 400h + 280h) to (3*400h + 280h + 16).

Switches 1-6 in the dip-switch SW4 are used to set the I/O base address for the direct access I/O registers. Switch 1 maps to the address bit 4 in the I/O address, switch 2 to the address bit 5, etc. Switch 6 maps to the most significant bit of the I/O-address, bit 9. Table 2 shows some example settings for some commonly used I/O base address settings. Note that dip switch position "On" corresponds to value "0" and position "Off" to value "1".

TABLE 2. Examples of SW4 I/O base address settings

IO Base Address	S6-S1 value	#1	#2	#3	#4	#5	#6	
0200h	20h = 100000b	On	On	On	On	On	Off	Not recommended[1]
0220h	22h = 100010b	On	Off	On	On	On	Off	
0240h	24h = 100100b	On	On	Off	On	On	Off	
0260h	26h = 100110b	On	Off	Off	On	On	Off	Not recommended[2]
0280h	28h = 101000b	On	On	On	Off	On	Off	**Thor-2 Default**
02A0h	2Ah = 101010b	On	Off	On	Off	On	Off	
02C0h	2Ch = 101100b	On	On	Off	Off	On	Off	
02Eh	2Eh = 101110b	On	Off	Off	Off	On	Off	Not recommended[3]
0300H	30H = 110000B	On	On	On	On	Off	Off	Default for many other boards
0320h	32h = 110010b	On	Off	On	On	Off	Off	
0340h	34h = 110100b	On	On	Off	On	Off	Off	
0360h	36h = 110110b	On	Off	Off	On	Off	Off	Not recommended[4]
0380h	38h = 111000b	On	On	On	Off	Off	Off	

TABLE 2. Examples of SW4 I/O base address settings

IO Base Address	S6-S1 value	#1	#2	#3	#4	#5	#6	
03A0h	3Ah = 111010b	On	Off	On	Off	Off	Off	Not recommended[5]
03C0h	3Ch = 111100b	On	On	Off	Off	Off	Off	Not recommended[6]
03E0h	3Eh = 111110b	On	Off	Off	Off	Off	Off	Not recommended[7]

[1] 02001h: Game control adapter

[2] 0278h-02F7h: Second printer port (LPT2)

[3] 02F8h-02FFh: Second serial port (COM2)

[4] 0378h-037Fh: First serial port (COM1)

[5] 03B0h-03B7h: Monochrome and printer adapter

[6] 03D0h-03DFh: Color/Graphics adapter

[7] 03F0h-03F7h: Floppy disk drive adapter card
03F8h-03FFh: Serial port adapter card.

The footnotes for Table 2 describe some commonly used PC hardware and their I/O port addresses. ***Do not use any of these I/O ports for Thor-2 if you have any of this hardware installed.*** Note that this list is not complete, and your particular PC manufacturer may have used other I/O ports than those mentioned here. Also, note that different boards use different number of I/O addresses following the base address. The I/O addresses above only indicate the start address of a range of addresses used (e.g. the Thor-2 board uses 16 bytes following the I/O base address for direct access I/O-registers). Make sure there are no overlaps, or erratic operation may result.

5.5 Setting the T1/E1 Line Termination (Switches SW1 and SW2)

Thor-2 allows the T1 or E1 lines to be terminated with different matching impedances. The different characteristics impedances for the different T1/E1 electrical interfaces can be set with dip-switches SW1 and SW2. The dip-switch SW1 affects the line interface 0 (Li0) and the dip-switch SW2 is connected to the line interface 1 (Li1). The different terminating impedance values and the corresponding dip switch positions are summarized in Table 3.

TABLE 3. Thor-2 T1/E1 Line Termination

Conf. #	Application (Characteristic Impedance)	Dip Switch SW1 and SW2 Positions							
		#1	#2	#3	#4	#5	#6	#7	#8
1	DS1 (100 Ohms as 6dB)	On	On	On	On	Off	Off	Off	Off
2	T1 (100 Ohms as 18 dB)	Off	On	Off	On	Off	Off	Off	Off
3	E1 (120 Ohms as 6 dB)	Off	Off	Off	Off	On	On	Off	On
4	E1 (75 Ohms as 6 dB)	Off	Off	Off	Off	Off	Off	On	Off

TABLE 3. Thor-2 T1/E1 Line Termination

Conf. #	Application (Characteristic Impedance)	Dip Switch SW1 and SW2 Positions							
		#1	#2	#3	#4	#5	#6	#7	#8
5	High-Impedance with Jumpers J11 and J12 Open	Don't Care	Don't Care	Don't Care	Don't Care	On	On	Off	Off

The Thor-2 board can also be used in protocol analyzer applications where the goal is to unintrusively listen both the transmit and receive lines. To accomplish that Thor-2 receivers can be configured into high-impedance mode without termination on the board. The high-impedance mode is set with jumpers J11 and J12 and with switches SW1 and SW2 (See Table 3 and Table 5).

TABLE 4. Setting Thor-2 Receivers into High-Impedance Mode

Jumper	Open	Closed
J11	Set Li0 Receiving lines in high-impedance (No termination). Set switch SW1 to configuration #5.	Terminate Li0 receiving lines according to the switch SW1 configuration settings 1-4. **(Thor-2 Default)**
J12	Set Li1 Receiving lines in high-impedance (No termination). Set switch SW2 to configuration #5.	Terminate Li1 receiving lines according to the switch SW2 configuration settings 1-4. **(Thor-2 Default)**

5.6 Setting the Memory Size (Jumpers J8 and J9)

The Thor-2 Board utilizes Dynamic Random Access Memory (DRAM) in 30-pin Single In-line Memory Modules (SIMMs). The board can be equipped with two 512 KByte, 1 MByte, or 4 MByte SIMMs. Note that both SIMMs must always be installed and that installed SIMMs must be of the same size and speed. The speed of the used memory should be 70 ns or faster.

Only use Odin TeleSystems Inc. approved SIMMs. If you are installing new memory, please contact Odin TeleSystems Inc. for a list of approved SIMMs.

The size of the used SIMMs is set using jumpers J8 and J9. Table 5 summarizes the setting of the memory size.

TABLE 5. Thor-2 Memory Size Setting

Installed SIMMs	Total Memory	Jumper J8				Jumper J9			
		Pins 1 - 2		Pins 2 -3		Pins 1 - 2		Pins 2 -3	
2 x 512 K	1 MByte	Closed		Open		Closed		Open	
2 x 1 M	2 MBytes	Open		Closed		Closed		Open	
2 x 4 M	8 MBytes	Open		Closed		Open		Closed	

5.7 Setting the MVIP Termination (Jumper J5)

Thor-2 provides an MVIP interface compliant to the MVIP-90 standard. The standard stipulates that MVIP compliant boards must provide the capability to terminate the MVIP clock lines with 1000 Ohm / 1000 pF series termination to ground. For systems with five or fewer MVIP Bus connections, the board that provides the master clock to the bus should be placed at one end of the cable and the board at the other end should provide the termination. On systems with more than five MVIP connections, both ends of the cable should provide the electrical termination.

The jumper J5 (pins 1-4) on Thor-2 are used to configure the MVIP termination. Table 5 summarizes the J5 (pins 1-4) settings for the MVIP termination. Note that J5 pins 5 and 6 are reserved test points and should not be connected with a jumper.

TABLE 6. Thor-2 MVIP Termination

Jumper J5	Open	Closed
Pins 1-2	Do not terminate MVIP Clock C4\ **(Thor-2 Default)**	Terminate MVIP Clock C4\ with 1000 Ohm - 1000 pF in series to ground
Pins 3-4	Do not terminate MVIP Clock C2 **(Thor-2 Default)**	Terminate MVIP Clock C2 with 1000 Ohm - 1000 pF in series to ground

6. Installing the Thor-2 Driver Software

In addition to the hardware settings described in the previous chapter, Thor-2 requires resources that are software configurable. These resources include an I/O address area for indirect access to the on-board I/O mapped devices, a memory area for indirect access to the on-board memory, and an IRQ line.

6.1 Selecting an I/O address area for indirect access I/O registers

In addition to the 16 I/O registers mapped above the I/O base address (See Chapter 5.4: "Setting the I/O Base Address for Direct Access I/O-Registers (Switch SW4)"), Thor-2 contains a large number of I/O-mapped devices. These devices are accessed indirectly through a sliding I/O window. For more information on the operation of the sliding I/O window, please refer to the *Thor-2 Technical Description* (Odin TeleSystems Inc. document number 1111-1-HAA-1004-1). The sliding I/O-window requires a free I/O address area. The base address and the size of the address area for the sliding I/O-window can be set with software. The minimum size of the window is 4 Bytes and the maximum is 1 KByte. The indirect I/O area does not have to be adjacent to the direct access I/O area. However, to ease the administration of the I/O space, a recommended value for the indirect access I/O area is 16 Bytes to be located immediately after the 16 Bytes used for the direct I/O access registers. Thus, making the total Thor-2 free I/O address requirement to 32 Bytes.

Make note of the free area available for the indirect I/O access and provide this information to the installation program *Setup.exe* and to the configuration program *T2Config.exe* (See Chapter 7: "Configuring the Thor-2 board with T2config.exe").

6.2 Selecting a memory area for indirect access to the Thor-2 on-board memory

The Thor-2 on-board memory is mapped to the PC's main memory area. The access to the on-board memory is performed through a sliding memory window. For more information on the operation of the sliding memory window, please refer to the *Thor-2 Technical Description*. The memory window requires a free memory area. The starting address of the area and the size of the area can be set with software. The minimum size of the memory window is 1 KBytes and the maximum size is 16 MBytes. The area for the sliding memory window must be reserved for Thor-2 by, for example, instructing the memory manager in use to exclude the area from the free memory pool. In DOS this is performed in the config.sys file by instructing the EMM386 to exclude an area. Figure 2 shows an example how to exclude 16 kbytes starting from address D0000.

```
DEVICE=C:\DOS\EMM386.EXE NOEMS X=D000-D3FF
```

Figure 2. Example on how to reserve a memory area in DOS config.sys file

Consult your PC’s memory manager documentation and your Thor-2 application software package documentation on how to reserve a memory area and how to set the sliding memory window starting address and size.

With DOS applications the sliding memory window is typically set into the upper memory area between 640 K and 1 M. Typical values are 32 K reserved between addresses D0000h - D7FFFh or 64 K reserved between addresses D0000h - DFFFFh. With Windows’95 or Windows NT the sliding memory window can be placed at any location within the first 16 MBytes of memory.

Make note of the free area available for the indirect memory access and provide this information to the installation program *Setup.exe* and to the configuration program *T2Config.exe* (See Chapter 7: "Configuring the Thor-2 board with T2config.exe").

6.3 Selecting an IRQ

Thor-2 requires an ISA bus IRQ line. If more than one Thor-2 boards are installed in one PC, all of the boards can share the same IRQ. However, allocating a separate IRQ to each board will improve the performance. Thor-2 can also share an interrupt with other “well-behaving” peripherals. The sharing of interrupts with other devices is, nevertheless, not generally recommended because the other devices may not have been properly designed to support IRQ sharing.

The IRQs supported by Thor-2 are listed in Table 7:

TABLE 7. IRQs supported by Thor-2

IRQ	Comment
3	Normally used by COM2 (if installed)
4	Normally used by COM1 (if installed)
5	Normally used by LPT2 (if installed)
7	Normally used by LPT1 (if installed)
10	Sometimes used by Video, LAN or Sound Card
11	Sometimes used by Video, LAN or Sound Card
12	Sometimes used by Mouse Port
15	Sometimes used by Secondary IDE Controller

Make note of the free area available for the indirect memory access and provide this information to the installation program *Setup.exe*

6.4 Installing the Windows Device Driver

In DOS based hosts systems the Thor-2 driver is linked into the Thor-2 application software. To communicate with the Thor-2 board, each DOS application needs to be provided the information on the IRQ, the I/O-address area, and the memory address area to be used. Please consult your DOS based application software documentation on how to provide this information to the application.

In Windows 95 and Windows NT based systems, the information on the IRQ, the I/O-address area, and the memory address area is stored in the Windows Registry. All the applications can access the information from the registry, and thus the information need to be provided only once during the installation. To install the Windows 95 or Windows NT driver, run the *Setup.exe* program from the distribution diskette, and answer the questions asked by the installation Wizard.

7. Configuring the Thor-2 board with *T2config.exe*

The Thor-2 board contains 512 KBytes of Flash memory for persistent storage. This memory can be used to persistently store T1/E1 configuration parameters. All the application programs can then read the parameters from the on-board flash and application specific configuration files are not needed.

Thor-2 is delivered with a utility program *T2Config.exe* which can be used for the management of the configuration parameters. The use of *T2Config.exe* and the syntax of the Thor-2 configuration file is described in this chapter. Note that *T2Config.exe* does not have to be used if the Thor-2 application software has been designed to handle the management of configuration data in some other fashion. *T2Config.exe* is merely provided to allow an easy utilization of the on-board flash for persistent storage of the configuration information. Consult your Thor-2 application software documentation to determine whether *T2Config.exe* need to be executed.

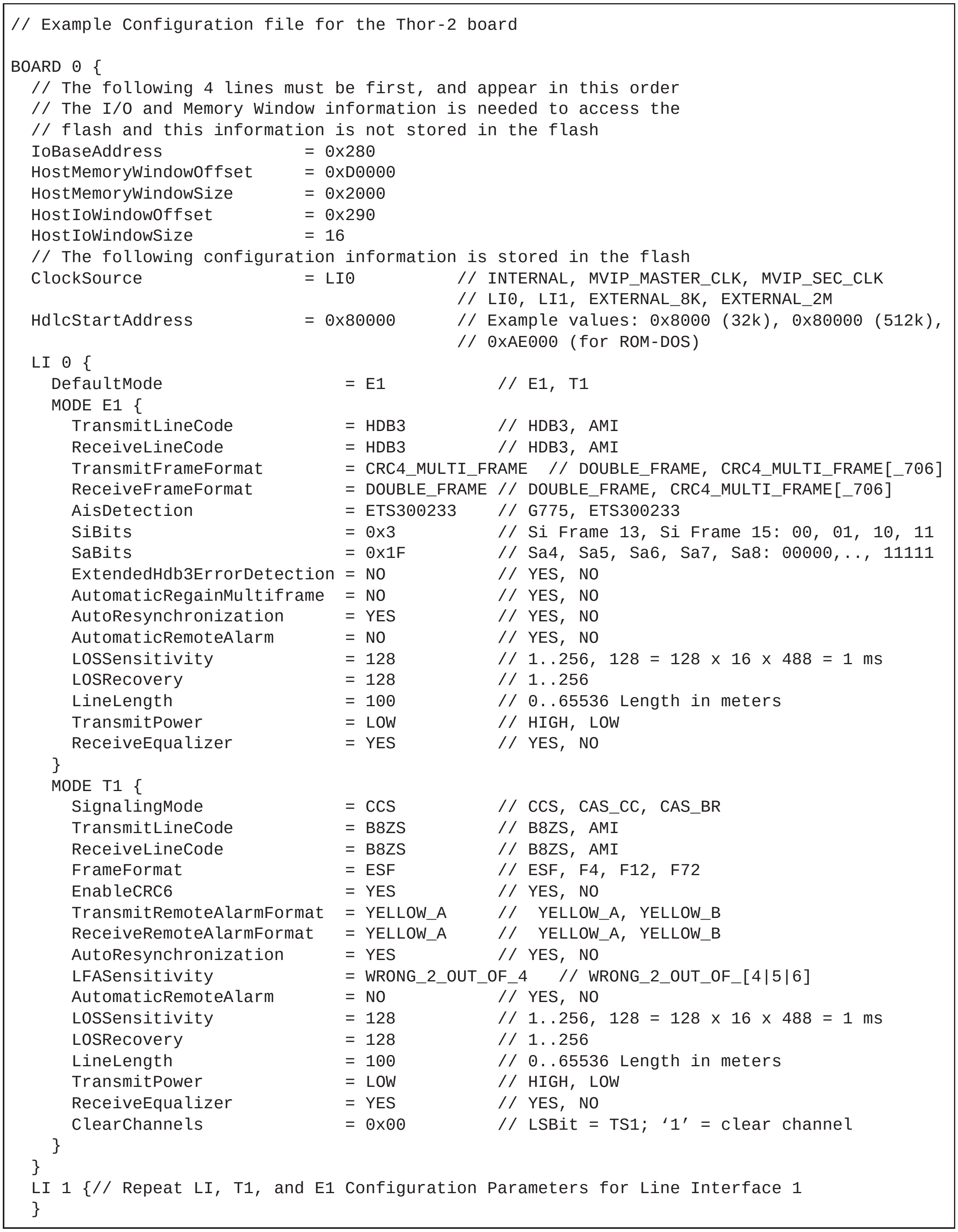
7.1 Thor-2 Configuration File

The parameters are passed to *T2Config.exe* by the means of a configuration file. The *T2Config.exe* program parses the configuration file and stores the configuration parameters in the flash memory sector #9. A Thor-2 Application software can then read the configuration parameters from the flash and configure the on-board devices accordingly. The overall structure of the configuration file is shown in Figure 3.

```
// Use C or C++ type comments
BOARD 0 {
  // Board Configuration Parameters for Board 0:
  boardParameter_1             = <boardParameter_1>
  boardParameter_2             = <boardParameter_2>
    :                               :
  LI 0 {
    // Line Interface (LI) Configuration Parameters for Li0:
    liParameter_1                = <liParameter_1>
         :                              :
    MODE E1 {
      // E1 Mode Configuration Parameters:
      E1Parameter_1              = <E1Parameter_1>
        :                        =    :
    }
    MODE T1 {
      // T1 Mode Configuration Parameters:
      T1Parameter_1              = <T1Parameter_1>
        :                        =    :
    }
  }
  LI 1 {
    :  // Line Interface (LI) Specific Configuration Parameters for Li1:
    MODE E1 {
      :  // E1 Mode Configuration Parameters:
    }
    MODE T1 {
      :  // T1 Mode Configuration Parameters:
    }
  }
}
BOARD 1 {
  : // Board Configuration Parameters for Board 1:
}
```

Figure 3. The overall Structure of the Thor-2 Configuration File

An example configuration file is shown in Figure 4. For detailed description of the Configuration file parameters, please see See “Thor-2 Configuration File Reference” on page 36.



```
// Example Configuration file for the Thor-2 board

BOARD 0 {
  // The following 4 lines must be first, and appear in this order
  // The I/O and Memory Window information is needed to access the
  // flash and this information is not stored in the flash
  IoBaseAddress              = 0x280
  HostMemoryWindowOffset     = 0xD0000
  HostMemoryWindowSize       = 0x2000
  HostIoWindowOffset         = 0x290
  HostIoWindowSize           = 16
  // The following configuration information is stored in the flash
  ClockSource                = LI0          // INTERNAL, MVIP_MASTER_CLK, MVIP_SEC_CLK
                                            // LI0, LI1, EXTERNAL_8K, EXTERNAL_2M
  HdlcStartAddress           = 0x80000      // Example values: 0x8000 (32k), 0x80000 (512k),
                                            // 0xAE000 (for ROM-DOS)
  LI 0 {
    DefaultMode                  = E1           // E1, T1
    MODE E1 {
      TransmitLineCode           = HDB3         // HDB3, AMI
      ReceiveLineCode            = HDB3         // HDB3, AMI
      TransmitFrameFormat        = CRC4_MULTI_FRAME  // DOUBLE_FRAME, CRC4_MULTI_FRAME[_706]
      ReceiveFrameFormat         = DOUBLE_FRAME // DOUBLE_FRAME, CRC4_MULTI_FRAME[_706]
      AisDetection               = ETS300233    // G775, ETS300233
      SiBits                     = 0x3          // Si Frame 13, Si Frame 15: 00, 01, 10, 11
      SaBits                     = 0x1F         // Sa4, Sa5, Sa6, Sa7, Sa8: 00000,.., 11111
      ExtendedHdb3ErrorDetection = NO           // YES, NO
      AutomaticRegainMultiframe  = NO           // YES, NO
      AutoResynchronization      = YES          // YES, NO
      AutomaticRemoteAlarm       = NO           // YES, NO
      LOSSensitivity             = 128          // 1..256, 128 = 128 x 16 x 488 = 1 ms
      LOSRecovery                = 128          // 1..256
      LineLength                 = 100          // 0..65536 Length in meters
      TransmitPower              = LOW          // HIGH, LOW
      ReceiveEqualizer           = YES          // YES, NO
    }
    MODE T1 {
      SignalingMode              = CCS          // CCS, CAS_CC, CAS_BR
      TransmitLineCode           = B8ZS         // B8ZS, AMI
      ReceiveLineCode            = B8ZS         // B8ZS, AMI
      FrameFormat                = ESF          // ESF, F4, F12, F72
      EnableCRC6                 = YES          // YES, NO
      TransmitRemoteAlarmFormat  = YELLOW_A     //  YELLOW_A, YELLOW_B
      ReceiveRemoteAlarmFormat   = YELLOW_A     //  YELLOW_A, YELLOW_B
      AutoResynchronization      = YES          // YES, NO
      LFASensitivity             = WRONG_2_OUT_OF_4   // WRONG_2_OUT_OF_[4|5|6]
      AutomaticRemoteAlarm       = NO           // YES, NO
      LOSSensitivity             = 128          // 1..256, 128 = 128 x 16 x 488 = 1 ms
      LOSRecovery                = 128          // 1..256
      LineLength                 = 100          // 0..65536 Length in meters
      TransmitPower              = LOW          // HIGH, LOW
      ReceiveEqualizer           = YES          // YES, NO
      ClearChannels              = 0x00         // LSBit = TS1; ‘1’ = clear channel
    }
  }
  LI 1 {// Repeat LI, T1, and E1 Configuration Parameters for Line Interface 1
  }
```

Figure 4. An example of the Thor-2 Configuration file

7.2 Running *T2Config.exe*

The configuration file to be loaded is given as a command line parameter to *T2Config.exe*. The command line syntax of *T2Config.exe* is specified in Figure 5 and an example of the use of *T2Config.exe* is illustrated in Figure 6.

```
T2CONFIG <command_file_name>
```

Figure 5. T2Config Command Line Syntax

```
T2CONFIG   T2.CFG
```

Figure 6. An example of using *T2Config.exe*

8. Booting up the Thor-2 board with *T2Boot.exe*

The Thor-2 board contains two Field Programmable Gate Arrays (FPGAs) and an on-board processor (Intel 386EX, which is called the Local Processing Unit, the LPU in Odin documentation). Before the board can operate, the FPGAs have to be loaded with their configuration data and the on-board processor, the LPU, must be booted. In Windows 95 and Windows NT based host systems, the Thor-2 device driver will perform these tasks automatically upon start-up. In DOS based systems, the Thor-2 driver cannot automatically start up the board. Instead, the supplied *T2Boot.exe* utility program must be executed to start up the board. Or alternatively, the Thor-2 application software must be designed to handle the start-up of the FPGAs and the LPU. If you are using the Thor-2 board in a DOS based host system, please consult your Thor-2 application software documentation to determine whether *T2Boot.exe* needs to be executed.

Note that *T2Boot.exe* is supplied with Windows 95 and Windows NT drivers as well, but it does not have to be used. It can, however, be used to diagnose the board or to load a new boot disk image to the on-board flash memory.

8.1 Using *T2Boot.exe* to load the FPGAs and to boot the on-board processor (DOS only)

In its simplest form, *T2Boot.exe* can be used to load the FPGAs and to boot up the on-board processor with Odin's proprietary operating system. **As was mentioned above, this operation is only needed in DOS based systems. With Windows 95 and Windows NT based systems, the FPGAs are loaded and the LPU booted automatically upon Windows start-up**.

In order to access the board, the *T2Boot.exe* must be provided with the board number, the I/O-base address, and the sliding memory window parameters. This information can be passed to the program as command line parameters (See Figure 7 and Figure 8).

```
T2Boot [BN=<board_number>] [IBA=<i/o_base_address>]
        [MO=<host_memory_offset>] [MS=<memory_window_size>]
```

Figure 7. *T2Boot.exe* (DOS) Command Line Syntax for loading FPGAs and booting the LPU

```
T2Boot BN=0 IBA=0x280 MO=0xD0000 MS=0x2000
T2Boot
```

Figure 8. Examples of using *T2Boot.exe* (DOS)

TABLE 8. T2Boot.exe Command Line Parameters for loading the FPGAs and booting the LPU with Odin Operating System

Parameter	Meaning	Possible Values	Default Value
BN	Board Number	0 .. 3	0
IBA	I/O Base Address	0x200 .. 0x3FF	0x280
MO	Host Memory Window Offset	0x100 .. 0x1000000	0xD0000
MS	Host Memory Window Size	0x100 .. 0x1000000	0x2000

If no command line parameters are specified, *T2Boot.exe* uses the default values. If *T2Boot.exe* does not find a Thor-2 board from the provided address, it will scan the I/O-address space 0x200 - 0x3FF for a Thor-2 board. If a board is found in another address, the *T2Boot.exe* program will attempt to start-up the board at that address.

Note: The default I/O-base address and memory window parameter values are provided as a convenience and they should only be used if they match the values in use by the Thor-2 board. If the Thor-2 board is configured for other values, provide the correct values as command line parameters. Use of incorrect values may cause erratic results.

8.2 Using *T2Boot.exe* to boot the on-board processor with ROM-DOS

The Thor-2 board on-board processor may also execute ROM-DOS^{TM} 6.22 operating System. ROM-DOS^{TM} 6.22 is a MS-DOS^{TM} 6.22 compatible operating system designed for embedded systems. The *T2Boot.exe* program can be used to boot up the on-board processor with the ROM-DOS operating system.

The Thor-2 boards equipped with ROM-DOS have a ROM-DOS license label from Datalight Inc. attached to the processor chip. If the Thor-2 board does not have the ROM-DOS label, it cannot run the ROM-DOS operating system. In that case please contact Odin TeleSystems Inc. to upgrade the board for ROM-DOS and to obtain a ROM-DOS license.

ROM-DOS operates within the first 1 Mega byte of on-board memory. This may conflict with the memory used by the HDLC controller. The memory area used by the HDLC controller can be set with the *T2Config.exe* program (HdlcStartAddress Parameter in the configuration file). Before using ROM-DOS, make sure that the HDLC start address has been moved beyond the memory area used by ROM-DOS. (For more information, please refer to chapter Chapter 11.1.7: "HdlcStartAddress")

In DOS based host systems, the *T2Boot.exe* must be provided with the I/O-base address and the sliding memory window parameters so that the program can access the board. In Windows 95 and Windows NT based host systems that is not necessary as the information is available in the Windows Registry. The *T2Boot.exe* command line syntax for booting ROM-DOS is illustrated in Figure 9 and Figure 10.

```
T2Boot A=DOS [BN=<board_number>] [IBA=<i/o_base_address>]
              [MO=<host_memory_offset>] [MS=<memory_window_size>]
```

Figure 9. T2Boot.exe (DOS) Command Line Syntax for Booting the LPU with ROM-DOS

```
T2Boot A=DOS [BN=<board_number>]
```

Figure 10. T2Boot.exe (Windows) Command Line Syntax for Booting the LPU with ROM-DOS

```
T2Boot A=DOS BN=0 IBA=0x280 MO=0xD0000 MS=0x2000
```

Figure 11. Examples of using *T2Boot.exe* (DOS) to Boot up ROM-DOS

```
T2Boot A=DOS BN=0
T2Boot A=DOS
```

Figure 12. Examples of using *T2Boot.exe* (Windows) to Boot up ROM-DOS

8.3 Using *T2Boot.exe* to load a new DOS disk-image in to the Flash

When the LPU is booted for ROM-DOS, the BIOS, the ROM-DOS operating system, and the Boot Disk Image are copied from the flash memory into the RAM memory. The user can create custom boot disks with customized *CONFIG.SYS* and *AUTOEXEC.BAT* files, as well as customized applications. *T2Boot.exe* can be used to store a new boot disk image in to the flash memory. The BIOS, the ROM-DOS operating System, and the Boot Disk are stored in sectors 0 - 4 of the Flash. For more information on the on-board flash, please refer to the Thor-2 Technical Description.

In DOS based systems, *T2Boot.exe* must be provided with information on the sliding host memory window so that program can access the on-board memory. In Windows 95 and Windows NT systems that is not necessary as the information is available in the Windows Registry. The T2Boot.exe command line syntax for loading a new disk image and booting the LPU with ROM-DOS is illustrated in Figure 9 and Figure 10.

```
T2Boot A=DOS [BN=<board_number>]
        [MO=<host_memory_offset>] [MS=<memory_window_size>]
        DI=<disk_image_file>
```

Figure 13. ***T2Boot.exe* (DOS) Command Line Syntax for loading a new LPU boot disk image**

```
T2Boot A=DOS [BN=<board_number>]
        DI=<disk_image_file>
```

Figure 14. ***T2Boot.exe* (Windows) Command Line Syntax for loading a new LPU boot disk image**

TABLE 9. T2Boot.exe Command Line Parameters for loading a new boot disk image

Parameter	Meaning	Possible Values	Default Value
A	Action Indicator	DOS	
BN	Board Number	0 .. 3	0
MO	Host Memory Offset	0x100 .. 0x1000000	DOS: 0xD0000 WIN: from Registry
MS	Memory Window Size	0x100 .. 0x1000000	DOS: 0x2000 WIN: from Registry
DI	Boot Disk Image	File name string	

8.3.1 Creating a new boot disk image

To create a new boot disk image, perform the following steps:

1. Place all the files to be included on the boot disk in a directory. E.g.; create a directory called TMPDISK and copy the files to be included into that directory.

2. Run the provided utility program ROMDISK.EXE. Provide the name of the directory containing the files and the name of the disk image to be loaded as parameters. E.g.:

   ```
   ROMDISK TMPDISK ROMDISK.IMG
   ```

3. The created file ROMDISK.IMG can now be loaded into the flash memory using *T2Boot.exe*, as explained in Chapter 8.3: "Using T2Boot.exe to load a new DOS disk-image in to the Flash".

8.4 Using *T2Boot.exe* to diagnose the board

T2Boot.exe can also be used to run diagnostics on the board. To perform the diagnostics, *T2Boot.exe* must know the settings for the sliding memory and I/O windows as well as the IRQ. In DOS systems, this information must be provided to *T2Boot.exe* as command line parameters. In Windows 95 and Windows NT systems, this information is available to T2Boot.exe from the Windows registry. The T2Boot command line syntax for Diagnostics is shown in Figure 15 and Figure 16.

```
T2Boot  A=DIAG [BN=<board_number>] [IBA=<i/o_base_address>]
          [IRQ=<interrupt_#_used>]
          [MO=<host_memory_offset>] [MS=<memory_window_size>]
          [IO=<host_i/o_offset>] [IS=<io_window_size>]
```

Figure 15. ***T2Boot.exe* (DOS) Command Line Syntax for Diagnostics**

```
T2Boot A=DIAG [BN=<board_number>]
```

Figure 16. ***T2Boot.exe* (Windows) Command Line Syntax for Diagnostics**

```
T2Boot A=DIAG BN=0 IBA=0x280 IRQ=10 MO=0xD0000 MS=0x2000
       IO=0x290 IS=16
```

Figure 17. An Example of using *T2Boot.exe* (DOS) for Diagnostics

```
T2Boot A=DIAG BN=0
```

Figure 18. An Example of using *T2Boot.exe* (Windows) for Diagnostics

TABLE 10. *T2Boot.exe* Command Line Parameters for diagnosing the board

Parameter	Meaning	Possible Values	Default Value
A	Action Indicator	DIAG	
BN	Board Number	0 .. 3	0
IBA	I/O Base Address	0x200 .. 0x3FF	DOS: 0x280 WIN: from Registry
IRQ	Interrupt Used	3, 4, 5, 7, 10, 11, 12, 15	DOS: 5 WIN: from Registry
MO	Host Memory Window Offset	0x100 .. 0x1000000	DOS: 0xD0000 WIN: from Registry
MS	Host Memory Window Size	0x100 .. 0x1000000	DOS: 0x2000 WIN: from Registry
IO	Host I/O Window Offset	0x4 .. 0x400	DOS: 0x290 WIN: from Registry
MS	Host I/O Window Size	0x4 .. 0x400	DOS: 0x10 WIN: from Registry

The various tests performed by the *T2Boot.exe* diagnostics are listed in Table 11.

TABLE 11. Thor-2 Diagnostics tests

Test Area	Test	Result
I/O Access Test	Reading and Writing to the Line Interface (LI) #0	Passed/Failed
	Reading and Writing to the Line Interface (Li) #1	Passed/Failed
	Reading and Writing to the Time-Space Switch (TSS)	Passed/Failed
	Reading and Writing to Codec #0	Passed/Failed
	Reading and Writing to Codec #1	Passed/Failed
Memory Test	Write 0x0000 to every memory location. Read back and verify the value.	Passed/Failed
	Write 0xFFFF to every memory location. Read back and verify the value.	Passed/Failed
	Verify reading from the Flash	Passed/Failed
Device Test	Line Interface Device #0 Configuration.	Passed/Failed
	Line Interface Device #1 Configuration.	Passed/Failed
	HDLC Controller. Configuration of Pipes.	Passed/Failed
	On-board processor (LPU). Verify that LPU can be interrupted by the host and that the LPU can access the I/O devices and the memory.	Passed/Failed
	Clock Switching using the Digital Configuration Switch.	Passed/Failed
	Sending of DTMF digits from one transceiver #0 and receiving by transceiver #1. And the other way around.	Passed/Failed

T2Boot.exe reports passed or failed for every test it runs. If any of the tests fail, check your PC for potential conflicts with other hardware utilizing the same I/O address area or the same IRQ. Also check that the memory area reserved for the Thor-2 sliding memory window is properly excluded from the use of the CPU or any other hardware in the PC.

9. Connecting Thor-2

9.1 Connecting the “Harmonica”

The back panel of Thor-2 contains a Centronics connector with 50 contacts. Thor-2 is delivered with a telco-type connector cable and a Harmonica module which converts from Centronics connector to 8 RJ-11connectors. The Harmonica module allows the connection of T1/E1 lines, handsets, speakers, external synchronization clocks, and serial lines to Thor-2 using RJ-11 connectors (See Figure 19).

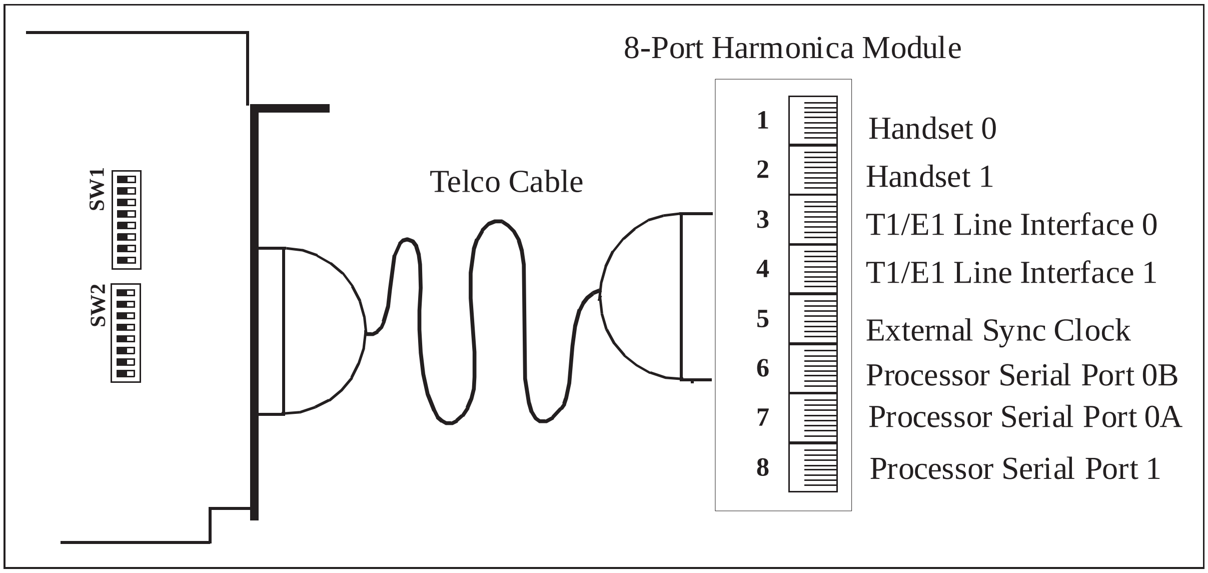


Figure 19. Thor-2 Eight Port Harmonica

The different connectors provided in the Harmonica are listed in Table 12.

TABLE 12. Harmonica Connector Assignment

Harmonica Connector #	Harmonica Label	Signal
1	☎ Hs 0	Handset for Codec #0
2	☎ Hs 1	Handset for Codec #1

TABLE 12. Harmonica Connector Assignment

Harmonica Connector #	Harmonica Label	Signal
3	T1/E1 Li 0	T1 or E1 for Line Interface #0
4	T1/E1 Li 1	T1 or E1 for Line Interface #1
5	Sync 🕒	External Synchronization Clock
6	🔧 Com 0B	The On-board Processor Serial Line #0 (LPU Com1) Modem Control Signals
7	🔧 Com 0A	The On-board Processor Serial Line #0 (LPU Com1)
8	🔧 Com 1	The On-board Processor Serial Line #1 (LPU Com2)

The Centronics Connector pin-out is show in Figure 20 and the pin assignments are listed in Table 13.

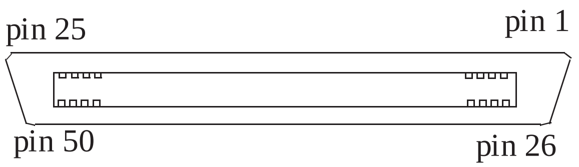


Figure 20. Thor-2 Centronics Connector

TABLE 13. Thor-2 Centronics Connector Pin Assignments

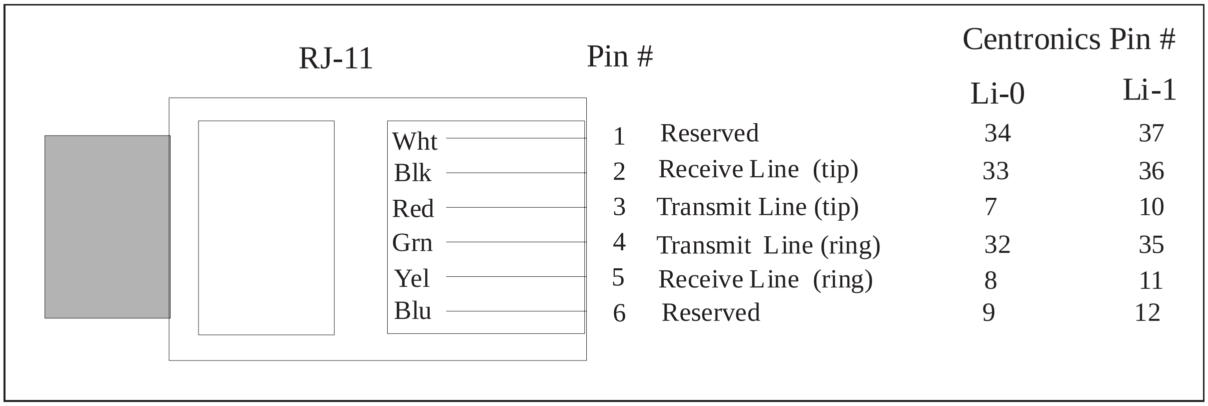
Pin	Signal	Pin	Signal
1	Codec-0, Handset Earpiece (-)	26	Codec-0, Handset Earpiece (+)
2	Codec-0, Handset Microphone (+)	27	Codec-0, Handset Microphone (-)

TABLE 13. Thor-2 Centronics Connector Pin Assignments

Pin	Signal	Pin	Signal
3	Codec-0, Speaker (+)	28	Codec-1, Speaker (-)
4	Codec-1, Handset Earpiece (-)	29	Codec-1, Handset Earpiece (+)
5	Codec-1, Handset Microphone (+)	30	Codec-1, Handset Microphone (-)
6	Codec-1, Speaker (+)	31	Codec-0, Speaker (-)
7	Line Interface 0, Transmit Line (tip)	32	Line Interface 0, Transmit Line (ring)
8	Line Interface 0, Receive Line (ring)	33	Line Interface 0, Receive Line (tip)
9	Reserved	34	Reserved
10	Line Interface 1, Transmit Line (tip)	35	Line Interface 1, Transmit Line (ring)
11	Line Interface 1, Receive Line (ring)	36	Line Interface 1, Receive Line (tip)
12	Reserved	37	Reserved
13	External Synch clock 1.544 or 2.048 MHz	38	External Synch Clock, 8 kHz
14	Ground	39	Ground
15	Reserved	40	Reserved
16	LPU Com0, Data Terminal Ready (DTR)	41	LPU Com0, Data Set Ready (DSR)
17	LPU Com0, Data Carrier Detected (DCD)	42	LPU Com0, Ring Indicator (RI)
18	Reserved	43	Ground
19	LPU Com0, Transmit Data (TXD)	44	LPU Com0, Receive Data (RXD)
20	LPU Com0, Clear to Send (CTS)	45	LPU Com0, Request to Send (RTS)
21	Reserved	46	Ground
22	LPU Com1, Transmit Data (TXD)	47	LPU Com1, Receive Data (RXD)
23	Reserved	48	Reserved
24	Reserved	49	Ground
25	Reserved	50	Reserved

9.2 Connecting to T1 or E1 Interface

The T1 or E1 lines can be connected to connectors 3 and 4 in the Harmonica. The RJ-11 pin-outs for the line interfaces are shown in Figure 21.



Pin #		Centronics Pin # Li-0	Centronics Pin # Li-1
1	Reserved	34	37
2	Receive Line (tip)	33	36
3	Transmit Line (tip)	7	10
4	Transmit Line (ring)	32	35
5	Receive Line (ring)	8	11
6	Reserved	9	12

Figure 21. RJ-11 Connector for T1 or E1 lines

9.3 Connecting to MVIP

Thor-2 supports the Multi-Vendor Integration Protocol (MVIP) and provides a 40-pin double row connector (3M #2540-5002UB or 3M #2440-5122) for the MVIP bus. The bus is carried in a 40-wire ribbon cable (3M #3365 or equivalent). The maximum length of the ribbon cable is 22 inches.

The MVIP connector pin-out is illustrated in Figure 22 and the pin assignment is listed in Table 13.

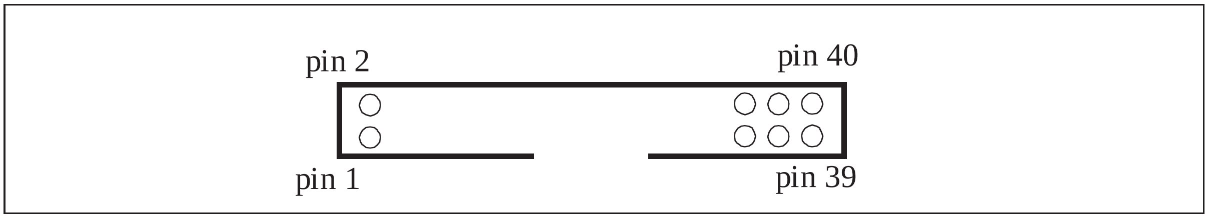


Figure 22. Thor-2 MVIP Connector

TABLE 14. Thor-2 MVIP Connector Pin Assignments

Pin	Signal	Pin	Signal
1	Reserved	2	Reserved
3	Reserved	4	Reserved

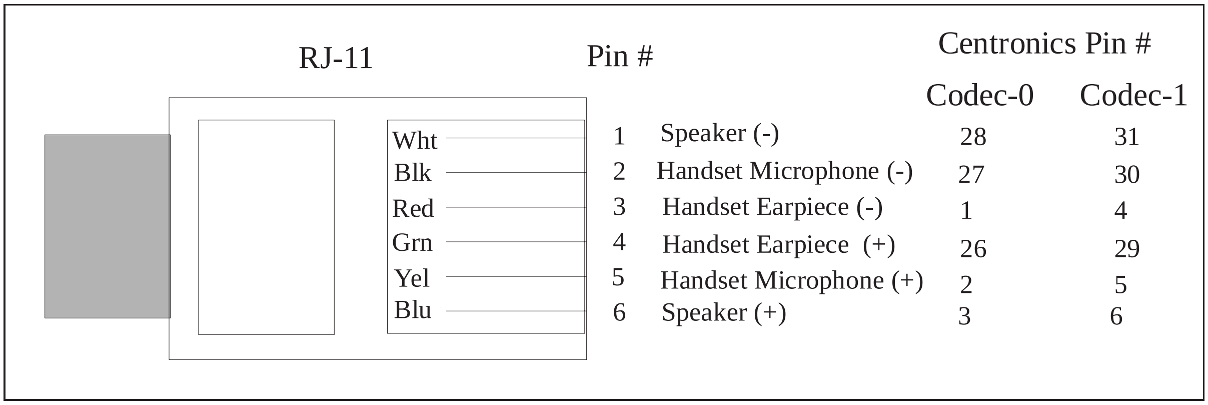
TABLE 14. Thor-2 MVIP Connector Pin Assignments

Pin	Signal	Pin	Signal
5	Reserved	6	Reserved
7	Digital Stream Out 0 (DSo0)	8	Digital Stream In 0 (DSi0)
9	Digital Stream Out 1 (DSo1)	10	Digital Stream In 1 (DSi1)
11	Digital Stream Out 2 (DSo2)	12	Digital Stream In 2 (DSi2)
13	Digital Stream Out 3 (DSo3)	14	Digital Stream In 3 (DSi3)
15	Digital Stream Out 4 (DSo4)	16	Digital Stream In 4 (DSi4)
17	Digital Stream Out 5 (DSo5)	18	Digital Stream In 5 (DSi5)
19	Digital Stream Out 6 (DSo6)	20	Digital Stream In 6 (DSi6)
21	Digital Stream Out 7 (DSo7)	22	Digital Stream In 7 (DSi7)
23	Reserved	24	Reserved
25	Reserved	26	Reserved
27	Reserved	28	Reserved
29	Reserved	30	Ground
31	/C4	32	Ground
33	/F0	34	Ground
35	C2	36	Ground
37	SEC8K	38	Ground
39	Reserved	40	Reserved

9.4 Connecting Accessories to Thor-2

9.4.1 Handset

The Thor-2 board contains two codecs which implement phone functionality. Two standard phone handsets can be connected to the codecs through connectors 1 and 2 in the Harmonica. Please note that to ensure proper connection, the handset cable must be equipped with an RJ-11 (6W6P or 4W6P) connector instead of RJ-12 (4W4P) connector commonly used in phone handsets. The RJ11 connector for the handsets is shown in Figure 23. The handset connectors can also be user to connect external loudspeakers to the codecs as shown in Figure 23.

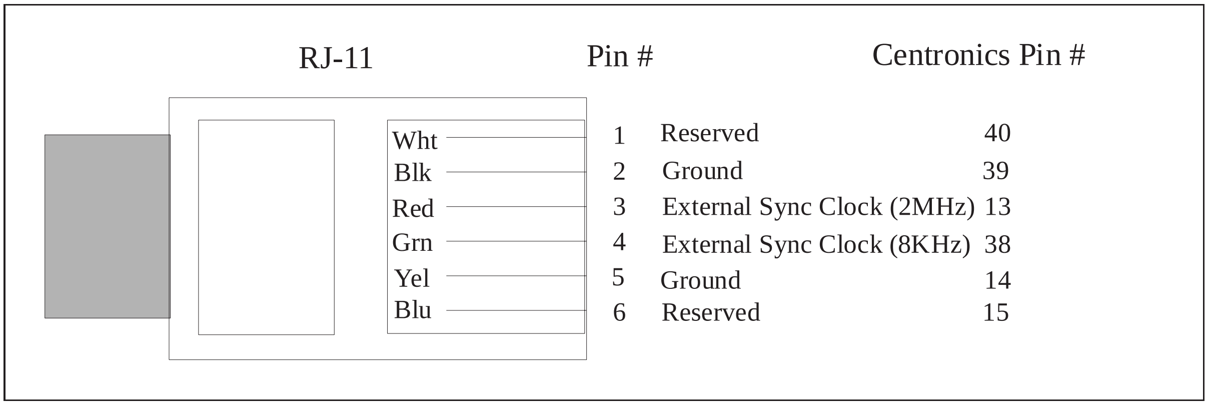


RJ-11	Pin #		Centronics Pin #	
			Codec-0	Codec-1
Wht	1	Speaker (-)	28	31
Blk	2	Handset Microphone (-)	27	30
Red	3	Handset Earpiece (-)	1	4
Grn	4	Handset Earpiece (+)	26	29
Yel	5	Handset Microphone (+)	2	5
Blu	6	Speaker (+)	3	6

Figure 23. RJ-11 Connector for Handsets and Speakers

9.4.2 External Synchronization Clock

An external synchronization clock can be provided for Thor-2. For more information on clocks, please refer to the *Thor-2 Technical Description*. The RJ-11 connector for the synchronization clocks is shown in Figure 24.



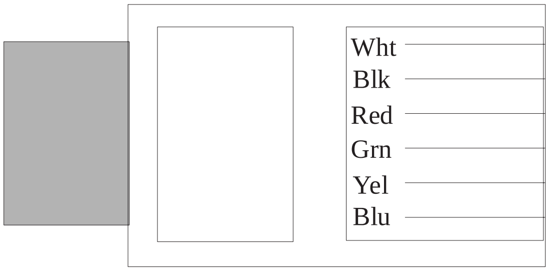
RJ-11	Pin #		Centronics Pin #
Wht	1	Reserved	40
Blk	2	Ground	39
Red	3	External Sync Clock (2MHz)	13
Grn	4	External Sync Clock (8KHz)	38
Yel	5	Ground	14
Blu	6	Reserved	15

Figure 24. RJ-11 Connector for the External Synchronization Clock

9.4.3 Serial Device or Modem

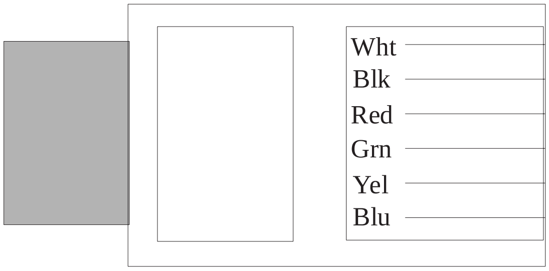
Thor-2 contains an on-board processor which supports two serial ports (COM ports). The serial ports can be accessed through connectors 6, 7, and 8 in the Harmonica. For more information on the on-board processor, please refer to the *Thor-2 Technical Description*.

The first COM port (COM0) includes modem control signals, while the second COM port (COM1) only supports data send and receive signals. The COM0 port is available on Connectors 6 and 7 (See Figure 25 and Figure 26), and the COM1 port is available on connector 8 (See Figure 27).



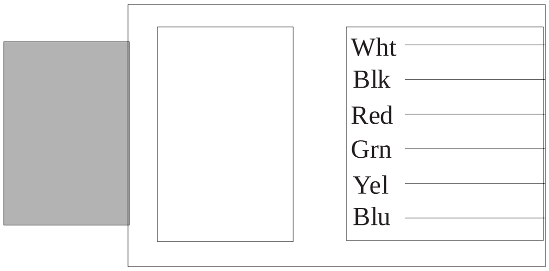
RJ-11	Pin #		Centronics Pin #
Wht	1	Ground	46
Blk	2	Request to Send (RTS)	45
Red	3	Transmit Data (TXD)	19
Grn	4	Receive Data (RXD)	44
Yel	5	Clear to Send (CTS)	20
Blu	6	Reserved	21

Figure 25. RJ-11 Connector for Com0-A



RJ-11	Pin #		Centronics Pin #
Wht	1	Ground	43
Blk	2	Ring Indicator (RI)	42
Red	3	Data Terminal Ready (DTR)	16
Grn	4	Data Set Ready (DSR)	41
Yel	5	Data Carrier Detected (DCD)	17
Blu	6	Reserved	18

Figure 26. RJ-11 Connector for Com0-B (Modem Control Signals)



RJ-11	Pin #		Centronics Pin #
Wht	1	Ground	49
Blk	2	Reserved	48
Red	3	Transmit Data (TXD)	22
Grn	4	Receive Data (RXD)	47
Yel	5	Reserved	23
Blu	6	Reserved	24

Figure 27. RJ-11 Connector for Com1

10. Setting up Thor-2 Processor Software Development Environment

As explained in Chapter 8, the Thor-2 on-board processor (Local Processing Unit, LPU) can run ROM-DOS^{TM} 6.22 embedded operating system. The processor serial ports available on the Harmonica connector can be used to communicate with the on-board processor. The host PC or another PC can be used as a console for the on-board processor, and/or the PC can be used to run a debugger user interface while performing remote debugging in the on-board processor.

10.1 Serial Cable

To connect a PC to the processor serial port, use a cable which has an RJ-11 connector at one end and a DB9 connector at the other end. The cable must contain at least three wires: the ground wire and the receive and transmit wires. Figure 28 illustrates how the RJ-11 and DB9 connectors are connected in the LPU serial cable.

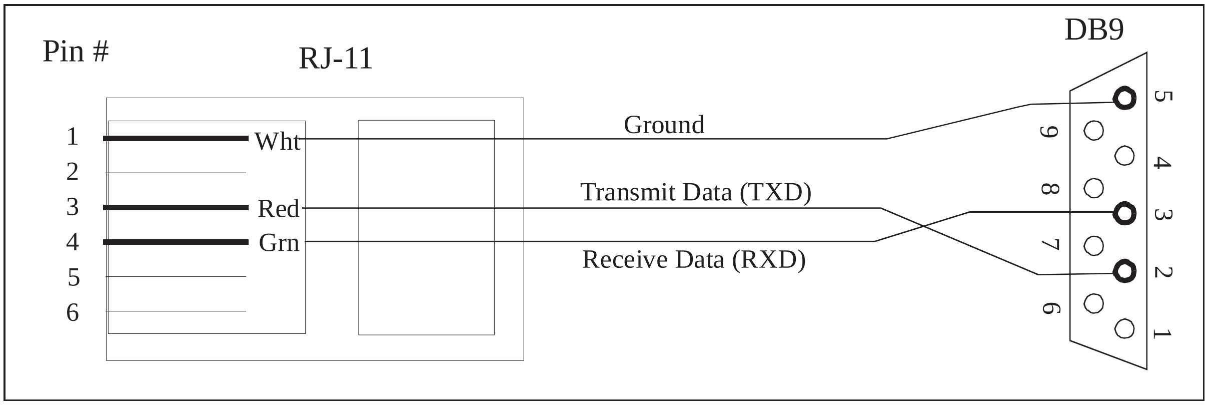


Figure 28. Thor-2 LPU Serial Cable

10.2 Setting up a Console for the Thor-2 on-board processor

When the LPU is running ROM-DOS, the processor console input and output can be redirected over the Harmonica COM1 port[1]. The console interface allows the user to utilize the ROM-DOS command interpreter and issue DOS commands.

To setup the LPU console, perform the following steps:

1. Connect a LPU serial Cable to the COM1 port on the Harmonica[2].
2. Connect the other end of the LPU serial Cable to a serial port in a PC.

[1] The Harmonica Port “Com0” is connected to the LPU Serial Port COM1, and the Harmonica Port “COM1” is connected to the LPU Serial Port COM2.

[2] Ditto.

3. In the PC run the provided utility program *Comm.exe* with parameters /B9600 and /COM# (Use the parameter /COM1 for Serial port #1 and /COM2 for serial port #2):

```
COMM /B9600 /COM1
```

4. Boot the on-board processor with ROM-DOS. See Chapter 8.2: "Using T2Boot.exe to boot the on-board processor with ROM-DOS".

5. You should now see the boot status messages and a *Thor-2>* command prompt. You are now executing the ROM-DOS command interpreter and can issue DOS commands towards the LPU.

10.3 Debugging applications in the Thor-2 on-board processor

Application Software developed for the Thor-2 LPU can be debugged in the on-board processor using a remote debugger. The Thor-2 hardware does not set any limitations on which Intel x86 remote debugger to use. However, this example explains how to use the Turbo Debugger (*TD.EXE*) from Borland International, Inc. To run a remote debugger in the LPU, first connect a console to the LPU as explained in Chapter 10.2: "Setting up a Console for the Thor-2 on-board processor". Once you have the console setup, connect a debugging console by performing the following steps:

1. Connect a LPU Serial cable to the COM0A port on the Harmonica[1].

2. Connect the other end of the LPU serial Cable to a serial port in a PC.

3. Boot the on-board processor with ROM-DOS. Make sure that you are using a disk-image that contains the remote debugger *TDREMOTE.EXE*. Also, make sure that the *CONFIG.SYS* file contains VDISK.SYS which will create writable a RAM-Disk (appears as B: drive to the LPU).

4. Use the console (See Chapter 10.2) to make the writable RAM-Disk (B:) as the default drive and start the remote debugger *TDREMOTE.EXE* in the LPU. Or alternatively, place commands *B:* and *TDREMOTE.EXE* in the LPU *AUTOEXEC.BAT* and restart the processor.

```
B:
TDREMOTE.EXE
```

5. In the PC, run Turbo Debugger *TD.EXE* with -r and -rpX options (for COM1 use -rp1, for COM2 use the -rp2 option):

```
TD -r -rp1
```

[1] The Harmonica Port “Com0” is connected to the LPU Serial Port COM1, and the Harmonica Port “COM1” is connected to the LPU Serial Port COM2.

11. Thor-2 Configuration File Reference

11.1 Board Configuration Parameters

11.1.1 IoBaseAddress

Synopsis

Set the I/O Base address to be used by Thor-2. Note that Thor-2 requires 16 I/O addresses. For example, by setting the I/O-base address to 0x280, instructs Thor-2 to use I/O-addresses 0x280 - 0x28F.

Note: This parameter is needed by the T2Config.exe to access the Thor-2 board. It is not saved in the flash.

Possible Values

0x000 ... 0x3FF

Example / Default Value

IoBaseAddress = 0x280

11.1.2 HostMemoryWindowOffset

Synopsis

Set the starting address of the memory reserved for Thor-2 sliding memory window. With DOS applications the memory offset is typically located in the upper memory area between 640K and 1M. Commonly used values are 32 K reserved between addresses 0xD0000 - 0xD7FFF or 64 K reserved between addresses 0xD0000 - 0xDFFFF. Also note that the memory area reserved for Thor-2 should be made unavailable to other application by, e.g., instructing the memory manager in use to exclude the area from the free memory pool.

Note: This parameter is needed by the T2Config.exe to access the Thor-2 board. It is not saved in the flash.

Possible Values

0x00000 ... 0xFFFFF

Example / Default Value

```
HostMemoryWindowOffset  =  0xD0000
```

11.1.3 HostMemoryWindowSize

Synopsis

Set the size of the Thor-2 sliding memory window.

Note: This parameter is needed by the T2Config.exe to access the Thor-2 board. It is not saved in the flash.

Possible Values

1024 ... 16777216   /* 1K ... 16M */

Example / Default Value

```
HostMemoryWindowSize  =  0x2000    /* 8 K */
```

11.1.4 HostIoWindowOffset

Synopsis

Set the starting address of the I/O reserved for the Thor-2 sliding I/O window for indirect I/O access. It is recommended that this area is located adjacent to the I/O-base address area used for direct I/O access.

Note: This parameter is needed by the T2Config.exe to access the Thor-2 board. It is not saved in the flash.

Possible Values

0x000 ... 0x3FF

Example / Default Value

```
HostIoWindowOffset  =  0x290
```

11.1.5 HostIoWindowSize

Synopsis

Set the size of the Thor-2 sliding I/O window.

Note: This parameter is needed by the T2Config.exe to access the Thor-2 board. It is not saved in the flash.

Possible Values

4 ... 1024

Example / Default Value

HostIoWindowSize = 16

11.1.6 ClockSource

Synopsis

Select the source for the Thor-2 Internal clocks. The clocking for the Thor-2 internal highways can be derived from multiple sources. The possible sources are On-board oscillator (INTERNAL), MVIP bus 8 kHz framing clock (MVIP_MASTER_CLK), MVIP bus 8 kHz secondary clock (MVIP_SEC_CLK), 8 kHz frame clock extracted from the incoming T1/E1 span 0 (LI0), 8 kHz frame clock extracted from the incoming T1/E1 span 1 (LI1), External 8 kHz framing clock (EXTERNAL_8K), or external 2 Mhz bit clock (EXTERNAL_2M).

Possible Values

INTERNAL
MVIP_MASTER_CLK
MVIP_SEC_CLK
LI0
LI1
EXTERNAL_8K
EXTERNAL_2M

Example / Default Value

ClockSource = LI0

11.1.7 HdlcStartAddress

Synopsis

Set the start address of the on-board memory area reserved for the HDLC receive and transmit buffers. If the on-board processor is run with the Odin proprietary monitor, the start address can be located anywhere above 0x8000 (32k). By default, the HDLC start address is located at 0x80000 (512k).

If on-board processor is run with the ROM-DOS operating system, the HDLC start address has to be moved upwards to avoid conflict with the ROM-DOS memory. The ROM-DOS operates and can access in real mode the first one Mega bytes of memory. The processor can also access memory above the first Mega byte, but in that case the processor must be switched to protected mode and back which is a relatively slow operation. If the ROM-DOS application does not have to access the HDLC transmit and receive buffers (or if performance is not a concern), then the HDLC start address should be set to 1 Mega Byte at (0x100000).

If the ROM-DOS application needs to access the HDLC buffers, then the start address should ne positioned within the ROM-DOS upper memory area between 0xAE000 and 0x100000.

The ROM-DOS memory map is as follows:
0x0 - 0x9FFFF : ROM-DOS and free RAM for Applications
0xA0000 - 0xA1FFF : BIOS
0xA2000 - 0XADFFF : ROM-DOS
0xAE000 - 0xXXXXX : ROM-DISK (with your application)

If the ROM-DOS upper memory area is used for HDLC buffers, it is suggested that the start address is set at 0xAE000 overlaying the ROM-DISK. This will give the maximum memory available for the HDLC controller and the best performance for the LPU access to the HDLC buffers. However, once started, the on-board processor application cannot exit, but the processor has to be re-booted (since the ROM-DISK has been overwritten with data buffers).

Possible Values

0x8000 (32K) ... 0x100000 (1M)

Example / Default Value

HdlcStartAddress = 0x80000 // 512k

11.2 Line Interface (LI) Configuration Parameters

11.2.1 DefaultMode

Synopsis

Select the mode the Line Interface is configured to by default.

Possible Values

T1
E1

Example / Default Value

DefaultMode = E1

11.3 E1 Mode Configuration Parameters

11.3.1 TransmitLineCode

Synopsis

Set the Line Code to be used to the transmit direction.

Possible Values

```
HDB3  /* High Density Bipolar 3  */
AMI   /* Alternate Mark Inversion */
```

Example / Default Value

TransmitLineCode = HDB3

11.3.2 ReceiveLineCode

Synopsis

Set the Line Code expected from the receive direction.

Possible Values

HDB3 /* High Density Bipolar 3 */
AMI /* Alternate Mark Inversion */

Example / Default Value

ReceiveLineCode = HDB3

11.3.3 TransmitFrameFormat

Synopsis

Set the Frame Format to be used to the transmit direction.

Possible Values

DOUBLE_FRAME
CRC4_MULTI_FRAME
CRC4_MULTI_FRAME_G706 /* Alignment according to ITU-T G.706 */

Example / Default Value

TransmitFrameFormat = DOUBLE_FRAME

11.3.4 ReceiveFrameFormat

Synopsis

Set the Frame Format expected from the receive direction.

Possible Values

DOUBLE_FRAME
CRC4_MULTI_FRAME
CRC4_MULTI_FRAME_G706 /* Alignment according to ITU-T G.706 */

Example / Default Value

ReceiveFrameFormat = DOUBLE_FRAME

11.3.5 AisDetection

Synopsis

Set the algorithm that is used to detect the Alarm Indication Signal (AIS). According to ETX300233 an AIS alarm is detected if the incoming data stream contains less than 3 zeros within 512 bits and a Loss of Frame Alignment is indicated. The alarm will be cleared if 3 or more zeros will be detected within 512 bits or a Frame Alignment Signal (FAS) word is found.

According to ITU-T G.775 an AIS alarm is detected if the incoming data stream contains less than 3 zeros within a doubleframe (512 bits) over a two doubleframe period (1024 bits). The alarm will be cleared if 3 or more zeros are detected within a doubleframe for two consecutive doubleframes.

Possible Values

ETS300233
G775

Example / Default Value

aisDetection = G775

11.3.6 SiBits

Synopsis

Set the S_i bits (spare bits for international use) transmitted in the bit position 1 of time-slot 0 in doubleframe format and in bit the 1 of time-slot 0 in frames 13 and 15 in CRC-Multiframe format.

Possible Values

```
0   /* Si Frame 13 = 0, Si Frame 15 = 0 */
1   /* Si Frame 13 = 0, Si Frame 15 = 1 */
2   /* Si Frame 13 = 1, Si Frame 15 = 0 */
3   /* Si Frame 13 = 1, Si Frame 15 = 1 */
```

Example / Default Value

SiBits = 0x3

11.3.7 SaBits

Synopsis

Set the S_a bits (spare bits for national use) transmitted in bit positions 4-8 of time-slot 0 of every other (odd) frame.

Possible Values

```
0   /* Sa4 = 0, Sa5 = 0, Sa6 = 0, Sa7 = 0, Sa8 =0 */
1   /* Sa4 = 0, Sa5 = 0, Sa6 = 0, Sa7 = 0, Sa8 = 1 */
  :
30  /* Sa4 = 1, Sa5 = 1, Sa6 = 1, Sa7 = 1, Sa8 = 0 */
31  /* Sa4 = 1, Sa5 = 1, Sa6 = 1, Sa7 = 1, Sa8 = 1 */
```

Example / Default Value

```
SaBits  =  0x1F
```

11.3.8 ExtendedHdb3ErrorDetection

Synopsis

Select the error detection mode. When turned off, only double violations are detected. When turned on, 0000 strings are detected as code violations as well.

Possible Values

YES
NO

Example / Default Value

```
ExtendedHdb3ErrorDetection  =  NO
```

11.3.9 AutomaticRegainMultiframe

Synopsis

When enabled, the receiver will search a new basic- and multiframing if more than 914 CRC errors have been detected in a time interval of one second.

Possible Values

YES
NO

Example / Default Value

AutomaticRegainMultiframe = NO

11.3.10 AutoResynchronization

Synopsis

When enabled, the Framer will automatically re-synchronize after loss of synchronization. If disabled the Framer will not synchronize automatically but a re-synchronization procedure must be specifically initiated by the software.

Possible Values

YES
NO

Example / Default Value

AutoResynchronization = YES

11.3.11 AutomaticRemoteAlarm

Synopsis

Select whether the Remote Alarm (Yellow Alarm) will be sent automatically in the outgoing data stream when the receiver is out of synchronization. If automatic remote alarm is enabled, the alarm is automatically removed when the receiver returns to the synchronized state.

Possible Values

YES
NO

Example / Default Value

AutomaticRemoteAlarm = NO

11.3.12 LOSSensitivity

Synopsis

Set the sensitivity for the Loss Of Signal (LOS) detection. A LOS alarm will be generated if the incoming data stream has no transitions during programmable number (N) of consecutive pulse periods. The number given as the LOSSensitivity value will be multiplied by 16 to arrive to the pulse count number N. The time for detecting LOS will be 16 x LOSSensitivity x PulseLength (488ns). The maximum time is 16 x 256 x 488ns = 2ms

Possible Values

1 .. 256

Example / Default Value

LOSSensitivity = 128 /* 16 x 128 x 488ns = 1ms */

11.3.13 LOSRecovery

Synopsis

Set the sensitivity of the Loss Of Signal (LOS) recovery. A LOS alarm will be cleared if in the incoming data a number of Pulses (N) occur during the time period set with the LOSSensitivity parameter. The time interval starts with the first detected pulse transition and a pulse count is incremented with every received pulse. If the pulse number is greater than the number N set with this parameter, the LOS alarm will be cleared.

Possible Values

1 .. 256

Example / Default Value

LOSRecovery = 128

11.3.14 LineLength

Synopsis

Specify the length of the line cable that the Thor-2 board is connected in meters. This information is used by the Thor-2 transceivers to adjust the transmit pulse masks according to the line length.

Possible Values

0 .. 65535

Example / Default Value

LineLength = 100

11.3.15 TransmitPower

Synopsis

When set to HIGH, the pulses on the line interface will be sent with high power. When set to LOW, the pulses will be sent with low power.

Possible Values

HIGH
LOW

Example / Default Value

TransmitPower = LOW

11.3.16 ReceiveEqualizer

Synopsis

The Receive Equalizer can be used match the 18dB ITU-T I.431 requirement. When the Equalizer is turned off, the Receiver is 6dB, when turned on, the Receiver is 18dB.

Possible Values

YES
NO

Example / Default Value

ReceiveEqualizer = YES

11.4 T1 Mode Configuration Parameters

11.4.1 TransmitLineCode

Synopsis

Set the Line Code to be used to the transmit direction.

Possible Values

```
B8ZS /* Bipolar 8 Zero Substitution */
AMI   /* Alternate Mark Inversion */
```

Example / Default Value

TransmitLineCode = B8ZS

11.4.2 ReceiveLineCode

Synopsis

Set the Line Code expected from the receive direction.

Possible Values

```
B8ZS    /*Bipolar 8 Zero Substitution*/
AMI     /*Alternate Mark Inversion*/
```

Example / Default Value

ReceiveLineCode = B8ZS

11.4.3 FrameFormat

Synopsis

Set the Frame Format to be used for both transmit and receive.

Possible Values

```
ESF      /* Extended Superframe */
F4       /* 4-Frame Multiframe    */
F12      /* 12-Frame Multiframe   */
F72      /* 72-Frame Multiframe   */
```

Example / Default Value

```
FrameFormat  =  ESF
```

11.4.4 EnableCRC6

Synopsis

Set whether CRC6 checking and generation will be enabled when using the ESF frame format. When disabled, the receive will not perform the CRC6 check and the CRC6 bits are set to ‘1’ to the transmit direction.

Possible Values

YES
NO

Example / Default Value

```
EnableCRC6  =  YES
```

11.4.5 TransmitRemoteAlarmFormat

Synopsis

Select the Remote Alarm Indication (Yellow Alarm) format to be used in transmit direction with the F12 and ESF Frames. If Format A is selected, with F12 Frame Format the bit position 2 in every time-slot is forced to ‘0’. With ESF Frame Format a pattern on ‘1111 1111 0000 0000 ...’ is sent in the data link channel. If Format B is selected, with F12 Frame Format the Frame Synchronization (FS) bit of frame 12 is forced to ‘1’. With ESF Frame Format the bit #2 in every channel is forced to ‘0’.

With F4 and F72 frames formats the Yellow alarm is indicated in both formats by forcing the bit #2 to ‘1’ in every speech channel.

Possible Values

YELLOW_A
YELLOW_B

Example / Default Value

TransmitRemoteAlarmFormat = YELLOW_A

11.4.6 ReceiveRemoteAlarmFormat

Synopsis

Select the Remote Alarm Indication (Yellow Alarm) format to be expected in receive direction with the F12 and ESF Frames. If Format A is selected, with F12 Frame Format the bit #2 in every channel is set to ‘0’. With ESF Frame Format a pattern on ‘1111 1111 0000 0000 ...’ is sent in the data link channel. If Format B is selected, with F12 Frame Format the Frame Synchronization (FS) bit of frame 12 is set to ‘0’. With ESF Frame Format the bit position 2 in every time-slot is set to ‘0’.

With F4 and F72 frames formats the Yellow alarm is indicated in both formats by forcing the bit position 2 to ‘1’ in every speech channel.

Possible Values

YELLOW_A
YELLOW_B

Example / Default Value

ReceiveRemoteAlarmFormat = YELLOW_A

11.4.7 AutoResynchronization

Possible Values

See E1 Mode Configuration Parameters.

Synopsis

See E1 Mode Configuration Parameters.

Example

See E1 Mode Configuration Parameters.

11.4.8 LFASensitivity

Synopsis

Set the condition when the Loss of Frame Alignment (LFA) will be reported. LFA can be reported if 2 out of 4, 5, or 6 framing bits have been detected to be incorrect.

Possible Values

WRONG_2_OUT_OF_4
WRONG_2_OUT_OF_5
WRONG_2_OUT_OF_6

Example / Default Value

LFASensitivity = WRONG_2_OUT_OF_4

11.4.9 AutomaticRemoteAlarm

Possible Values

See E1 Mode Configuration Parameters.

Synopsis

See E1 Mode Configuration Parameters.

Example

See E1 Mode Configuration Parameters.

11.4.10 LOSSensitivity

Possible Values

See E1 Mode Configuration Parameters.

Synopsis

See E1 Mode Configuration Parameters.

Example

See E1 Mode Configuration Parameters.

11.4.11 LOSRecovery

Possible Values

See E1 Mode Configuration Parameters.

Synopsis

See E1 Mode Configuration Parameters.

Example

See E1 Mode Configuration Parameters.

11.4.12 LineLength

Synopsis

See E1 Mode Configuration Parameters.

Possible Values

See E1 Mode Configuration Parameters.

Example / Default Value

See E1 Mode Configuration Parameters.

11.4.13 TransmitPower

Synopsis

See E1 Mode Configuration Parameters.

Possible Values

See E1 Mode Configuration Parameters.

Example / Default Value

See E1 Mode Configuration Parameters.

11.4.14 ReceiveEqualizer

Synopsis

See E1 Mode Configuration Parameters.

Possible Values

See E1 Mode Configuration Parameters.

Example / Default Value

See E1 Mode Configuration Parameters.

Doc. No. 1412-1-HAA-1004-1
For more information on this product, please contact:

Odin TeleSystems Inc.
800 E. Campbell Road, Suite 300
Richardson, Texas 75081
U. S. A.

Tel: +1-972-664-0100
Fax: +1-972-664-0855
Email: Info@OdinTS.com
URL: http://www.OdinTS.com/