MicroLAB Systems

DIGITAL SIGNAL PROCESSING

TORNADO-32L

Floating Point DSP System with TMS320C32 DSP for ISA-bus PC

User's Guide

MicroLAB Systems Ltd
59a Beskudnikovsky blvd, 127486, Mocsow, RUSSIA
phone/fax: +7-(095)-485-6332 Email: mlabsys@online.ru

CAUTION

Фирма *МикроЛАБ Системс Лтд* (*МЛС*) оставляет за собой право вносить любые изменения и прекращать выпуск и поддержку любых изделий и программного обеспечения, упоминаемых в настоящем документе, без какого-либо предварительного уведомления, если иное специально не оговаривается. *МЛС* рекомендует своим покупателям пользоваться подлинными и самыми последними версиями фирменных информационных документов и осуществлять предварительное консультирование с фирмой перед размещением заказа, чтобы быть уверенным, что настоящая информация достоверна и применима на текущий момент.

МЛС гарантирует качество и соответствие технических параметров поставляемой продукции приведенной технической спецификации. Всякое тестирование и проверка продукции производятся фирмой в степени и объемах, необходимых для поддержки настоящей гарантии. Какое-либо дополнительное тестирование и проверка продукции на соответствие другим требованиям проводятся лишь и только в случаях, выполняемых по специальным заказам, или тогда, когда это специально оговаривается.

МЛС не несет никакой ответственности за правильность функционирования и работоспособность оборудования и программного обеспечения, разработанного и изготовленного с применением продукции (или отдельных ее компонентов) фирмы, если это не подтверждено специальным фирменным сертификатом *МЛС*.

Продукция *МЛС* не предназначена для применения в аппаратуре, системах или изделиях для поддержки жизнидеятельности. Применение продукции *МЛС* в таком оборудовании категорически запрещено без специального письменного подтверждения от *МЛС* или оригинального фирменного сертификата *МЛС*.

IMPORTANT NOTICE

Настоящая продукция предназначена для использования в составе лабораторного тестового и научно-исследовательского оборудования. *МЛС* не несет ответственности за работоспособность настоящей продукции в составе другого типа оборудования и/или в отличных от специфицированных условиях эксплуатации. При повреждениях настоящей продукции, вызванных ее применением в составе другого типа оборудования и/или условий эксплуатации, гарантийные обязательства аннулируются без какого-либо возмещения ущерба и ремонт производится за счет покупателя.

Настоящая продукция генерирует, использует и может излучать радиочастотную энергию, которая может создавать радиочастотные помехи для другой аппаратуры, несмотря на все конструктивные и другие меры, предпринятые для минимизации создаваемых помех. Однако, в случае возникновения помех для работы другой аппаратуры покупатель должен сам и за свой счет принять меры для их устранения или уменьшения.

ITEMS OF LICENSE AGREEMENT

Никакие части настоящего документа, аппаратные и программные части настоящей продукции не могут быть реассемблированы, ретрассированы и/или изменены с целью восстановления и/или изменения электрической схемы, конструкции, алгоритма работы или принципа функционирования любыми методами, воспроизведены, скопированы, запомнены в архивах с возможностью воспроизведения, а также переданы по средствам связи в любом виде и любыми методами, будь то электронные, механические, копировальные, фотографические, записывающие или другие, без предварительно выданного фирменного письменного разрешения от *МЛС*. Нарушение настоящего положения вне зависимости от приобретения настоящей продукции и/или документа трактуется как нарушение авторских прав и преследуется по закону.

Приобретение настоящей продукции автоматически означает согласие покупателя с положениями лицензионного соглашения, равно как и другими положениями закона об авторских правах. Нарушение настоящих положений, равно как и других положений закона об авторских правах, трактуется как нарушение авторских прав, преследуется по закону и автоматически ведет к аннулированию всех обязательств *МЛС* по поддержке настоящей продукции.

DOC: MLS-SPUG-025E revision 1A-1
October 1996

About this Document

This user's guide contains description for *TORNADO-32L* floating point digital signal processing (DSP) system utilizing TMS320C32 DSP for host ISA-bus IBM PC personal computers.

This document does not include detail description neither for TI TMS320C3x DSP nor for corresponding software and hardware applications. To get the corresponding information please refer to the following documentation:

1. ***TMS320C3x User's Guide.*** Texas Instruments Inc, SPRU031D, USA, 1994.
2. ***TMS320C32 Addendum to the TMS320C3x User's Guide.*** Texas Instruments Inc, SPRU132B, USA, 1995.
3. ***TMS320C3x C Source Debugger User`s Guide.*** Texas Instruments Inc, SPRU053A, USA, 1991.
4. ***TMS320 Floating-Point DSP Optimizing C Compiler User`s Guide.*** Texas Instruments Inc, SPRU034B, USA, 1995.
5. *B.W.Kernighan and D.M.Ritchie.* ***The C Programming Language.*** Second Ed. Prentice Hall, Englewood Cliffs, NJ, USA, 1988.
6. ***TMS320 Floating-Point DSP Assembly Language Tools User`s Guide.*** Texas Instruments Inc, SPRU035B, USA, 1995.
7. ***Nucleus RTX Reference.*** Accelerated Technology Inc, USA, 1992.
8. ***Nucleus Plus Reference Manual.*** Accelerated Technology Inc, USA, 1993.
9. ***SPOX. The DSP Operating System. A Technical Overview.*** Spectron Microsystems Inc, USA, 1991.
10. ***VIRTUOSO. User Manual.*** Intelligent Systems International, Belgium, 1993.

Warranty

All hardware and software products purchased from MicroLAB Systems Ltd are guaranteed against damages during *one year* after shipment. MicroLAB Systems Ltd guarantees free of charge repair or replacement for the manufacturer caused damaged products during warranty period. Software updates will be also sent free of charge to the customer during warranty period.

If you need assistance, documentation or information...

Should you need technical assistance for purchased MicroLAB Systems Ltd products or if you want to order additional documentation or get latest information about MicroLAB Systems Ltd products, please call, fax or mail to MicroLAB Systems Ltd customer support service:

address:	59a Beskudnikovsky blvd, 127486, Moscow, RUSSIA. *MicroLAB Systems Ltd*
phone:	+7-(095)-485-6332, +7-(095)-488-8744
fax:	+7-(095)-485-6332
E-mail:	mlabsys@online.ru

Trademarks

Contents

Figures

Tables

Chapter 1. Introduction

This chapter contains general description of *TORNADO-32L* system and ordering information.

1.1 General Information

TORNADO-32L system is a high performance floating point DSP system board for ISA-bus IBM PC host computers. *TORNADO-32L* was designed to be plugged into 16-bit ISA-bus slot of host PC and features low cost and flexible modular system architecture in order to meet requirements for a variety of different applications while keeping a cost of project to a minimum.

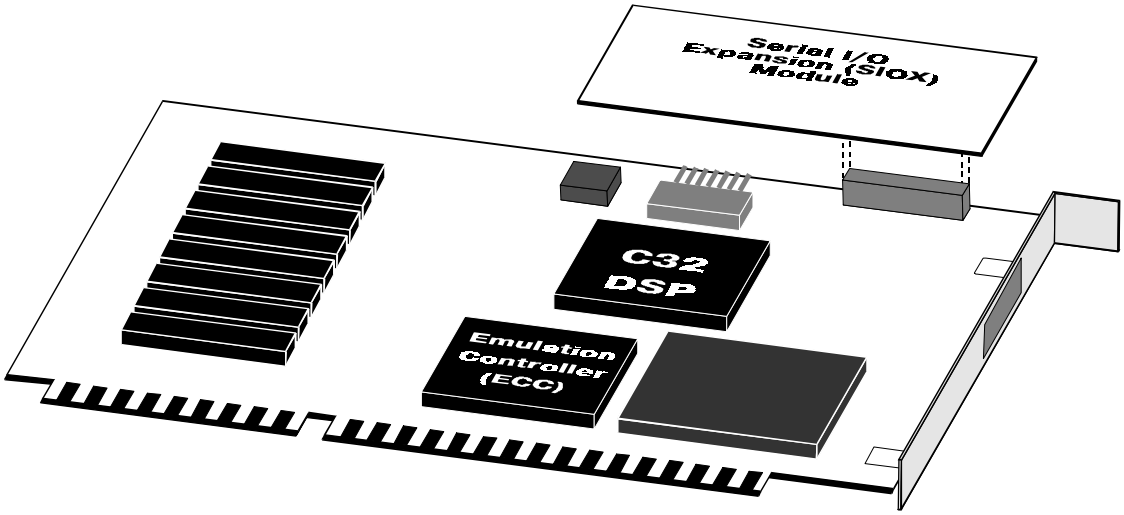

Fig. 1-1. *TORNADO-32L* DSP System board with serial I/O expansion daughter module and emulation controller *ECC* installed.

The following are some applications for *TORNADO-32L* :

- *real-time DSP and signal acquisition*
- *speech signal processing and acoustics*
- *communication*
- *multimedia*
- *instrumentation and industrial*
- *floating point DSP accelerator for CAD/CAE* software systems for PC
- *evaluation*
- many more ...

TORNADO-32L utilizes TI TMS320C32 60 MFLOPS 32-bit floating point DSP chip and has 8K..256Kx32 0ws on-board static RAM (SRAM) for program and data. On-board SRAM is splitted into two memory banks: bank #0 and bank #1.

TORNADO-32L has internal shared bus (SB) architecture that shares access to bank #0 of SRAM between the on-board TMS320C32 DSP chip and host ISA-bus memory interface. Host ISA-bus memory interface can provide access to SRAM both in random and block data transfer modes all in parallel with DSP chip operation almost without consuming the DSP time. This feature as well as high floating point DSP performance and large SRAM capacity make *TORNADO-32L* an ideal selection as a hardware floating point accelerator for different CAD/CAE software systems with intensive computations and multiple dataflows between ISA-bus host PC and DSP engine.

TORNADO-32L features an optional facility for installation of serial I/O expansion (SIOX) daughter card module from a variety of AD/DA and digital I/O SIOX modules for real-time instrumentation, industrial and speech, telecommunication and audio signal processing applications.

TORNADO-32L uses the scan-path emulation control of the on-board TMS320C32 DSP chip in order to develop and debug resident TMS320C32 DSP software. The scan-path emulation control of the on-board TMS320C32 DSP chip can be performed either via external TI XDS510 or MicroLAB' *MIRAGE-510D* scan-path emulators, or by means of optional *emulation controller chip (ECC)* that can be plugged into the supplied on-board socket. The *ECC* is a low cost replacement for the TI XDS510 and MicroLAB's *MIRAGE-510D* scan-path emulators and runs under identical industry standard TI TMS320C3x C Source Debugger and GoDSP C3x Code Composer IDE.

TORNADO-32L software development tools include TI floating point DSP C compiler and Assembly language tools.

TORNADO-32L is supported by a variety of industry standard 3rd party DSP software tools, that inculde real-time operating systems (RTOS) (*Nucleus RTX/PLUS* from Accelerated Technology Inc, *VIRTUOSO* from Eonic Systems Inc and *SPOX* from Spectron Microsystems Inc), DSP algorithm development and simulation tools from Hyperception Inc, digital filter design tools, DSP/vector/math function libraries, vocoder/fax/modem function libraries, and many more.

TORNADO-32L provides unique burn-in device serial codes, that can be read by host software and used for hardware copyright protection of the supplied software tools for OEM software vendors.

1.2 Host PC Specifications

TORNADO-32L requires that the host ISA-bus IBM PC configuration should be at least 80386SX CPU and provides at least one 16-bit ISA-bus slot.

In order to learn configuration requirements for host PC running TMS320C3x DSP software development and debugging tools, refer to the corresponding documentation from TI and Go DSP Corp as well as to the MicroLAB' "*UECM/ECC User's Guide*".

1.3 Ordering Specification

The following are p/n for the *TORNADO-32L* DSP System hardware.

Спецификация	*описание*
T32L	*TORNADO-32L* 60MFLOPS DSP System board, 0 SRAM.
DS8K-15	8Kx32 0ws SRAM chip set for *TORNADO-32L*.
DS32K-15	32Kx32 0ws SRAM chip set for *TORNADO-32L*.
DS128K-15	128Kx32 0ws SRAM chip set for *TORNADO-32L*.
T3x/ECC	Emulation controller *ECC* for *TORNADO-32L*.

1.4 Technical Specification

The following are the technical specifications for the *TORNADO-32L* system specified for the temperature +25°C of the environment.

parameter description	*parameter value*
power supply voltage	+5V for *TORNADO-32L* board, optional ±5V/±12V for SIOX daughter modules
power consumption (with 128Kx32 SRAM installed in bank #0, *ECC* emulation controller installed)	+5V@1.3A
DSP performance	60 MFLOPS, 30 MIPS
dimensions	160x118 mm
operating temperature	+10..+45°C
I/O expansion I/F	TORNADO-SIOX (with one SIO port)
host ISA-bus interface:	
number of I/O ports	8
size of ISA-bus memory page in the UMB memory address space for accessing the SB	32Kx8
timeout control time for waiting the SB granting	4.2 us

host IRQ lines	IRQ 3, 4, 5, 6, 7, 10, 11, 12, 15
on-board SRAM:	
number of memory banks	2
memory capacity of each bank	8K..128K x32
capacity of SRAM chips used	8K/32K/64K/128K x8
number of SRAM chips in each	4
access time for SRAM chips	≤15 ns
SRAM chip package	DIP 28pin 300MIL for 8K/32K x8 chips DIP 32pin 300MIL for 64K/128K x8 chips

Chapter 2. System Architecture and Construction

This chapter contains description for the *TORNADO-32L* system architecture and construction, host ISA-bus interface and SIOX I/O expansion interface.

2.1 *TORNADO-32L* System Architecture

TORNADO-32L mainboard was designed to be installed into ISA-bus of host IBM PC/AT personal computer (fig.1-1). The *TORNADO-32L* mainboard architecture is presented at fig.2-1.

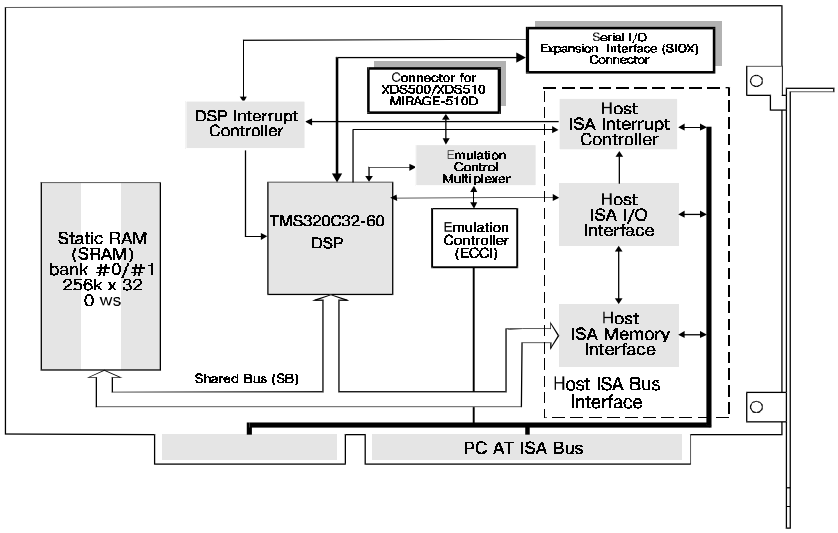

Fig.2-1. Architecture of *TORNADO-32L* mainboard.

The main components of the *TORNADO-32L* mainboard are:

- *TMS320C32 60 MFLOPS 32-bit floating point DSP* chip
- *SRAM* including banks #0 and #1
- *host ISA-bus memory interface*
- *serial I/O expansion I/F (SIOX)*
- *emulation controller (ECC)*

The TMS320C32 DSP, bank #0 of SRAM and host ISA-bus memory I/F are linked together by means of the on-board *Shared Bus (SB)*. The SB shares SRAM bank #0 resource between two 'bus masters', that can execute SB access cysles: the on-board TMS320C32 DSP chip and host ISA-bus memory interface. The on-board arbitration assumes that TMS320C32 DSP bus master has the highest SB priority whereas the ISA-bus memory interface is designed to access the SB in-parallel

with the DSP internal operation without any software overhead and almost without consuming the DSP time.

Construction of the *TORNADO-32L* mainboard is presented at fig.2-2.

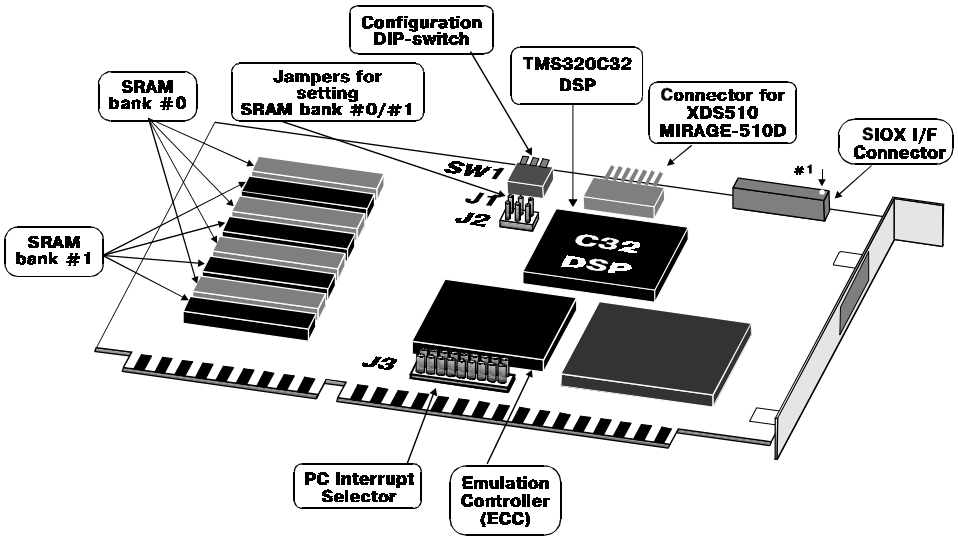

Fig.2-2. Construction of the *TORNADO-32L* mainboard.

On-board TMS320C32 DSP

The on-board TMS320C32 DSP is the 32-bit floating point DSP chip with on-chip Harvard architecture providing 60 MFLOPS and 30 MIPS of peak performance..

Static RAM (SRAM)

TORNADO-32L provides 8K..256Kx32 of total on-board SRAM capacity that comprises of two SRAM banks (bank #0 and bank #1). Each SRAM banks appears as four byte wide SRAM chips in 300MIL DIP package. *TORNADO-32L* allows installation of 8Kx8, 32Kx8, 64Kx8 and 128Kx8 SRAM chips in order to meet memory requirements of particular applications. particular size for each memory bank can be setup separately using on-board jumpers J1 and J2.

Shared Bus (SB)

The on-board SB of *TORNADO-32L* provides access to the on-board SRAM bank #0 for both on-board TMS320C32 DSP chip and host ISA-bus memory interface. The SB address space consist of 128K 32-bit words. The SB supports 8/16/32-bit data cycles with maximum throughoutput of 120 MB/sec. It is important to note, that all host acceeses to the on-board SRAM bank #0 are performed concurrently with the DSP running and without any DSP software overhead.

Host ISA-bus Interface

TORNADO-32L host ISA-bus interface was designed for DSP/system control and high-speed data transfer between host ISA-bus and on-board SRAM bank #0. *TORNADO-32L* host ISA-bus interface includes:

- *ISA-bus memory interface* that performs access to SRAM bank #0
- *ISA-bus I/O interface* that provides *TORNADO-32L* system control and configuring of the SB access modes for ISA-bus memory interface.

Host ISA-bus memory interface accesses SB data via 32Kx8 *shared memory page* (*SMP*) that is mapped into ISA-bus UMB memory address space. Once ISA-bus executes memory cycle within address range of *SMP*, then the on-board *TORNADO-32L* ISA-bus memory I/F generates request to SB. Particular allocation of *SMP* onto SB address space is done by means of SB page mapping register from ISA-bus I/O interface. Host can access the SB data using any of 8/16/32-bit data cycles. Host ISA-bus memory interface has lowest SB access priority.

ISA-bus base memory addresses for host ISA-bus memory is setup by host software and can be switched off in case *TORNADO-32L* board is not used.

ISA-bus base I/O address for ISA-bus I/O interface is configured by the on-board SW1 DIP-switch.

Serial I/O Expansion Interface (SIOX)

TORNADO-32L SIOX interface site is used for installation of AD/DA/DIO daughter card modules and comprises of signals for TMS320C32 DSP on-chip serial port, timers and interrupt control.

SIOX compatible daughter card modules include a variety of multichannel instrumentation AD/DA/DIO modules, telecommunication codecs, speech and stereo-audio codecs, DAT interface, and many more.

Debugging of resident DSP Software

Resident DSP software for *TORNADO-32L* can be debugged using TI XDS510 and MicroLAB' *MIRAGE-510D* scan-path emulators. However, to minimize cost of debugging tools, the emulation controller chip (*ECC*) that can be plugged into the dedicated on-board socket. *ECC* is a low cost replacement for XDS510 and *MIRAGE-510D* emulators and runs under identical industry standard TI TMS320C3x C Source Debugger and GoDSP Code Composer IDE.

2.2 Shared Bus

The *TORNADO-32L* on-board shared bus (SB) has 128Kx32 address space and supports 8/16/32-bit data cycles with the throughout performance of up to 120 MB/sec. The SB is shared between the the on-board TMS320C32 DSP chip and host ISA-bus memory interface.

SB Address Space

The SB address space is actually the address space for the *TORNADO-32L* on-board SRAM bank #0. Table 2-1 specifies the valid SB address subspaces.

Table 2-1. SB address subspaces and data ready wait times.

SB address subspace name	address range (in the 32-bit word units)	maximum wait time for the SB_READY signal (SB data ready) after SB is granted for	
		on-board TMS320C32 DSP	*host ISA-bus memory interface*
SRAM bank #0 (128Kx32)	000000H...01FFFFH	0ws	0ws

On the host ISA-bus I/F side the SB address space appears as a series of dual-access 32KB *shared memory pages* (*SMP*) that are mapped onto the predefined ISA-bus UMB (upper memory blocks) memory window by means of *SB PAGE MAPPER* register in host ISA-bus I/O interface. The SB can be accessed by host ISA-bus memory interface by means of random accesses to software variables or data arrays that are allocated within *SMP*, or by means of block data transfers between PC main memory and *SMP* using either host i80x286 CPU MOVSB/MOVSW/etc instructions or host DMA controller.

SB Data Ready Signal

SB has internal *SB_READY* signal that is generated by passive addressed device (SRAM bank #0) in order to acknowledge that the SB data are valid after SB is granted to the corresponding SB master. When SB is accessed by the on-board TMS320C32 DSP master, the *SB_READY* signal is logically connected to the *RDY* pin of TMS320C32 DSP, whereas for accesses from host ISA-bus memory interface the *SB_READY* signal is automatically processed by the SB access controller of ISA-bus memory interface.

TORNADO-32L provides 0ws SRAM bank #0 access times for both TMS320C32 DSP and host ISA-bus memory I/F masters.

SB Data Cycle Formats

The SB supports 8/16/32-bit data cycles in order to address any byte or 16-bit halfword of the SB 32-bit data words. The on-board TMS320C32 DSP master provides only 32-bit SB data cycles whereas host ISA-bus memory interface can be software configured for any of 8/16/32-bit SB data cycles (see section 2.4).

SB Arbitration

When SB is requested by any of the SB masters (TMS320C32 DSP or host ISA-bus memory I/F), then some time is required to resolve the arbitration. This normally takes about 1-2 TMS320C32 DSP clock cycles.

In case TMS320C32 DSP is requesting SB while the latter is occupied by host ISA-bus memory I/F, then the DSP should wait until host ISA-bus memory I/F will release SB. After SB is granted to the DSP, it is holded by DSP in order to provide 0ws for all further SB accesses.

In case host ISA-bus memory I/F is requesting SB while the latter is occupied by the TMS320C32 DSP, then host ISA-bus memory I/F has to wait until DSP will complete current SB access cycle and release SB.

When SB is requested by both DSP and host ISA-bus memory I/F, then DSP has the highest SB access priority.

SB Locking

The SB arbiter supports program *SB locking* in order to lock the SB access for processing of program semaphores or shared PIOX resources.

The SB locking by the on-board TMS320C32 DSP bus master is performed automatically when it executes the *LDII/LDFI* instructions. The corresponding SB unlocking is provided by execution of the *STII/STFI/SIGI* instructions (see section 2.3).

CAUTION

Time interval between execution of *LDII/LDFI* and *STII/STFI* instructions by the on-board TMS320C32 DSP should not exceed 4.2 μsec.

The SB locking by host ISA-bus memory interface master is performed by means of program setting the *SB_GLOCK* or the *SB_LOCK* bits of *CONTROL REGISTER* from the host ISA-bus I/O interface.

CAUTION

Continuous SB locking by host ISA-bus memory interface by means of setting the *SB_GLOCK* and *SB_LOCK* bits can result in continuous halting of the on-board TMS320C32 DSP bus master and may lead to time distortions of real-time data processing.

2.3 TMS320C32 DSP Environment

The *TORNADO-32L* on-board TMS320C32 DSP is the powerful 60 MFLOPS 32-bit floating point DSP chip from Texas Instruments Inc. Pls refer to the original TI technical documentation for details:

TMS320C3x User's Guide. Texas Instruments Inc, SPRU031D, 1994.

TMS320C32 Addendum to the TMS320C3x User's Guide. Texas Instruments Inc, SPRU132B, USA, 1995.

TMS320C32 DSP Address Space

The address space for the *TORNADO-32L* on-board TMS320C32 DSP is presented in table 2-2.

Table 2-2. Address space for TMS320C32 DSP of *TORNADO-32L*.

Address space of TMS320C32 DSP	**address range (in 32-bit words)**	**access wait states**
SRAM bank #0 (8•..128Kx32), which is a shared resource of the on-board SB and can be accessed by both TMS320C32 DSP and ISA-bus memory I/F; is strobe by the STRB0 memory strobe	000000H...01FFFFH	0ws
local SRAM bank #1 SRAM (8•..128Kx32); is strobe by the STRB1 memory strobe	900000H...91FFFFH	0ws
on-chip memory and registers of TMS320C32 DSP	808000H...8097FFH 87FE00H...87FFFFH	0ws

The TMS320C32 DSP can access both SRAM banks (bank #0 and bank #1) using 8/16/32 bit data access cycles.

Configuring the STRB0, STRB1 and IOSTRB Control Registers of TMS320C32 DSP

Note, that in order to benefit of full performance of TMS320C32 DSP, be sure to setup on-chip *STRB0/STRB1/IOSTRB CONTROL* registers as the following:

- the *STRB0 Control* (@808064H) and *STRB1 Control* (@808068H) registers should be configured as the following:
  000C0600H when accessing memory as 8-bit words
  000B0600H when accessing memory as 16-bit words
  000F0600H when accessing memory as 16-bit words
- the *IOSTRB Control* (@808060H) register is not used and should be set to 00000000H

SB Locking by the on-board TMS320C32 DSP Master

SB locking technique is used for processing of software shared semaphores that can be allocated in SRAM bank #0.

SB locking/unlocking by the on-board TMS320C32 DSP master is performed automatically when DSP chips executes the *LDII/LDFI/STII/STFI/SIGI* instructions (*Interlocked Operations*). For more detail information please refer to the

TMS320C3x User's Guide. Texas Instruments Inc, SPRU031D, USA, 1994.

The *LDII/LDFI/STII/STFI/SIGI* instructions assumes automatic utilization of the TMS320C32 on-chip *XF0/XF1* hardware flags (pins) for handshaking between the SB requester and SB arbiter.

The *XF0/XF1* flags/pins of *TORNADO-32L* on-board TMS320C32 DSP are used by the on-board SB arbiter for SB locking. The *XF0* flag is used to lock/unlock the SB whereas the *XF1* always reads as '0' and is used to acknowledge the lock/unlock event.

CAUTION

The *XF0/XF1* flags/pins of *TORNADO-32L* on-board TMS320C32 DSP cannot be used as programmable I/O flags/pins by *TORNADO-32L* resident software.

SB locking is performed automatically by SB arbiter when *TORNADO-32L* on-board TMS320C32 DSP executes *LDII/LDFI* instructions. These instructions result in setting flag *XF0* to the *XF0*=0 state that corresponds to active SB locking by TMS320C32 DSP. The *XF0*=0 state is held by TMS320C32 DSP until TMS320C32 DSP will execute *STII/STFI/SIGI* instructions that reset flag *XF0* to the *XF0*=1 state. The SB lock-to-unlock time interval is not limited by *TORNADO-32L* hardware, however long duration of theSB lock event by DSP may cause timeout access faults for SB accesses by host ISA-bus memory I/F.

The following is a software example that demonstrates processing of software shared semaphore using SB locking technique:

```
                …
Wait_Sem_Free:  LDII  @Sem,R1        ; read semaphore using SB locking
                BZ    L1             ; check for semaphore is free (Sem=0)
                SIGI                 ; semaphore is not free, unlock SB
                B     Wait_Sem_Free  ; repeat semaphore wait cycle
          L1:   LDI   1,R1           ; semaphore is free
                STII  R1,@Sem        ; set semaphore (Sem=1) and unlock SB
                …                    ; perform some processing with the semaphore
                …                    ; being set (Sem=1)
                LDII  @Sem,R1        ; reset semaphore (Sem=0) using SB locking
                LDI   0,R1
                STII  R1,@Sem        ; save semaphore and unlock SB
                …
```

Note that SB accesses from host ISA-bus memory interface are performed under on-board hardware timeout control for the SB granting wait time. This timeout interval is setup to 4.2 μsec. In case timeout will occur due to SB locking by DSP, the *SB_ERROR* flag in *FLAG STATUS REGISTER* of host ISA-bus I/O interface will be set to the *SB_ERROR*=1 state. This will result in cancellation of all further SB requests from host ISA-bus memory interface until the *SB_ERROR* flag will be reset to the *SB_ERROR*=0 state by host PC software.

CAUTION

Time interval between execution of *LDII/LDFI* and *STII/STFI* instructions by the on-board TMS320C32 DSP should not exceed 4.2 μsec.

Generating Request to Host PC

TORNADO-32L supports attention request (interrupt request) from the on-board TMS320C32 DSP to host IBM PC CPU in order to synchronize between program execution in host and on-board DSP environments. This is called *MH_RQ* (master to host request).

The *MH_RQ* is generated when the on-board TMS320C32 DSP executes *IACK* (*interrupt acknowledge*) instruction. Execution of *IACK* instruction results in setting flag *MH_RQ* in *FLAG STATUS REGISTER* of host ISA-bus I/O interface into the *MH_RQ*=1 state. The address pointer that should be specified with the *IACK* instruction has no meaning.

The following is a software example that generates request to the host PC:

```
…
LDI 0,AR5
IACK *AR5            ; generation of request to the host IBM PC/AT
…
```

The *MH_RQ*=1 event may generate active interrupt to host PC in case the *MH_RQ_IE* bit in *CONTROL REGISTER* of host ISA-bus I/O interface is set to the *MH_RQ_IE*=1 state. *MH_RQ* flag may also be read by host PC software from the *FLAG STATUS REGISTER* of host ISA-bus I/O interface.

Processing Request from Host PC

TORNADO-32L supports attention request (interrupt request) from host PC to TMS320C32 DSP in order to synchronize between the program execution in host and on-board DSP environments. This is called *HM_RQ* (host to master request).

The *HM_RQ* is generated when host PC setup flag *SET_HOST_TO_MASTER_REQUEST* in host ISA-bus I/O interface, i.e. by means of setting *FLAG SELECTOR REGISTER* to the *SET_HOST_TO_MASTER_REQUEST* value and succeeding writing of any code to *FLAG CONTROL REGISTER* of host ISA-bus I/O interface. This results in generation of *INT3* external hardware interrupt for the on-board TMS320C32 DSP. User software for the TMS320C32 DSP should provide processing of *INT3* hardware interrupt request in accordance with software requirements.

External Hardware Interrupts for TMS320C32 DSP

The *TORNADO-32L* on-board TMS320C32 DSP supports four external hardware interrupt requests *INT0...INT3* with the *INT0* request having the highest priority. These requests correspond to the following events:

- *INT0...INT2* interrupt requests can be generated by SIOX daughter modules
- *INT3* is on-board wired for software request *HM_RQ* from host PC to the TMS320C32 DSP

TORNADO-32L hardware provides direct wiring of external interrupt request source signals to the corresponding *INT0...INT3* pins of TMS320C32 DSP chip. User software can select between usage of edge- or level-triggered interrupts (using bit *INT CONFIG* from TMS320C32 *ST* register) in accordance with software requrements.

2.4 Host ISA-bus Memory Interface

Host ISA-bus memory interface is used to transfer data between host IBM PC environment and *TORNADO-32L* on-board SRAM bank #0 without any software overhead for both host PC and on-board TMS320C32 DSP.

On the host ISA-bus I/F side the SB address space appears as a series of dual-access 32KB *shared memory pages* (*SMP*) that are mapped onto the predefined ISA-bus UMB (upper memory blocks) memory window by means of *SB PAGE MAPPER* register in host ISA-bus I/O interface. The SB can be accessed by host ISA-bus memory interface by means of random accesses to software variables or data arrays that are allocated within *SMP*, or by means of block data transfers between PC main memory and *SMP* using either host i80x286 CPU MOVSB/MOVSW/etc instructions or host DMA controller.

Host ISA-bus memory interface issues SB request and provides SB access using 8/16/32-bit data cycles each time host PC performs ISA-bus memory read/write cycle within the ISA-bus *SMP* address range.

SMP ISA-bus Memory Base Address

SMP ISA-bus memory base address can be setup within the ISA-bus UMB (upper memory blocks) memory address range by means of 3-bit *ISA_MI_BADDR_FRG* flag register from *TORNADO-32L* host ISA-bus I/O I/F in accordance with the predefined configuration settings in table 2-3.

Table 2-3. ISA-bus memory base address for *SMP*.

ISA-bus memory base address for *SMP*	ISA-bus memory address range for *SMP*	bit settings for *ISA_MI_BADDR_FRG* flag register		
		bit#2	*bit#1*	*bit#0*
SMP is switched OFF	-	0	0	0
B8000H	*B8000H ... BFFFFH*	0	0	1
C0000H	*C0000H ... C7FFFH*	0	1	0
C8000H	*C8000H ... CFFFFH*	0	1	1
D0000H	*D0000H ... D7FFFH*	1	0	0
D8000H	*D8000H ... DFFFFH*	1	0	1
E0000H	*E0000H ... E7FFFH*	1	1	0
E8000H	*E8000H ... EFFFFH*	1	1	1

Notes: 1. The highlighted configuration corresponds to power on default value.

TORNADO-32L, as well as other *TORNADO-3x* DSP systems, offers software control for *SMP* activity, i.e. switching *SMP* to either 'ON' or 'OFF' state in ISA-bus memory address space.

CAUTION

SMP is activated and appears in ISA-bus memory address space after loading any non-zero value to *ISA_MI_BADDR_FRG* flag register in accordance with table 2-3.

SMP is disactivated and disappears from ISA-bus memory address space after loading the zero value o *ISA_MI_BADDR_FRG* flag register.

Software control over *SMP* activity in *TORNADO-32L* delivers rational usage of deficit UMB area in host PC and allows multiple *TORNADO* DSP systems to operate within one PC environment.

Software Applications for ISA-bus Memory Interface

Since *TORNADO-32L* host ISA-bus memory interface provides direct mapping of 32KB *SMP* onto the ISA-bus UMB window, the following data transfer techniques are available:

- *random access to variables or data arrays* allocated within the *SMP* by host PC software
- *block data transfers using MOVS/MOVSB/MOVSW instructions* of host PC i80x86 CPU
- *block data transfers under control of host PC DMA controller* using memory-to-memory or memory-to-port transfer cycles.

Operation Description for ISA-bus Memory Interface

SB access from host ISA-bus memory interface is performed under hardware control of the on-board programmable *Multiformat SB Access Controller* from ISA-bus interface of *TORNADO-32L*. Timing diagram for SB read cycle invoked by ISA-bus memory interface is presented on fig.2-3.

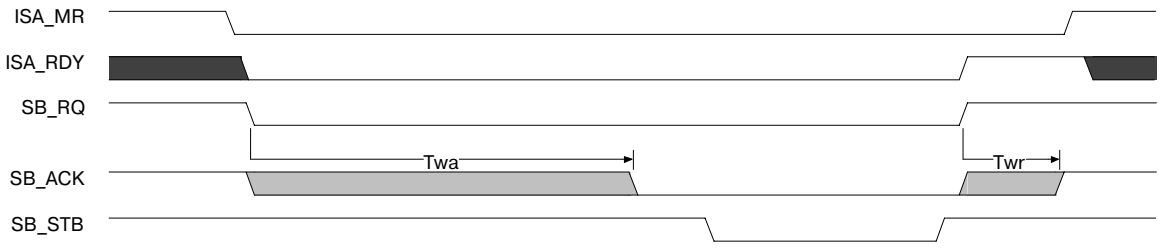

Fig.2-3. Timing diagram of SB read cycle invoked by ISA-bus memory interface.

Any host ISA-bus memory cycle within *SMP* memory address space (*ISA_MR*=0 or *ISA_MW*=0) will result in activation of host ISA-bus memory interface that will generate request to SB arbiter (*SB_RQ*=0) and will set ISA-bus data ready signal to the 'NOT READY' state (*ISA_RDY*=0). Host ISA-bus memory interface will keep staying in this state until SB will be granted (*SB_ACK*=0) by SB arbiter. SB will be granted at least after the T_{wa}=66ns wait time. After that, SB access cycle will be generated by host ISA-bus memory interface with *SB_STB* signal set to the *SB_STB*=0 state. Afterthat, *SB_RQ* signal is removed (*SB_RQ*=1) and *ISA_RDY* signal is set to the *ISA_RDY* =1 state

in order to finish current ISA-bus memory access cycle. The *SB_ACK* signal will return to its inactive state (*SB_ACK*=1) within T_{Wr}=33ns after *SB_RQ* will be removed (*SB_RQ*=1).

SB Access Timeout Control

TORNADO-32L provides hardware timeout control for the wait time of the SB granting signal when SB is accessed by host ISA-bus memory interface. This is required in order to avoid idling of host PC software and crashing of host PC environment.

Hardware timeouts for SB granting signal is set to 4.2 usec. Once SB granting timeout occurs, the *SB_ERROR* bit in *CONTROL REGISTER* from host ISA-bus I/O interface is set to the *SB_ERROR*=1 state. This will cancell all succeeding SB requests from host ISA-bus memory interface until *SB_ERROR* bit will be reset by host software.

Data Formats for Host SB Data Access Cycles

TORNADO-32L supports 8/16/32-bit SB data access cycles for host ISA-bus memory interface. Current format for host SB data cycle can be programmed by host software by means of setting *SB_CCL* bit field {*SB_CCL-0*,*SB_CCL-1*} of *CONTROL REGISTER* from host ISA-bus I/O interface in accordance with table 2-4.

Table 2-4. Data formats for host SB data cycles.

format of host SB data cycle	*SB_CCL* bit field value of *CONTROL REGISTER* from ISA-bus I/O interface		description
	SB_CCL-0	*SB_CCL-1*	
8-bit data cycle	0	0	Host SB data cycle is generated each time host PC CPU executes ISA-bus memory read/write cycle within *SMP* ISA-bus memory address range. Actual byte selection within the addressed SB 32-bit word is performed by ISA-bus address bits {*A0, A1*}. *SMP* appears as the 32KB linear byte space.
16-bit data cycle	1	0	Host SB data cycle is generated when host PC CPU performs either ISA-bus memory read cycle for even (*A0*=0) byte or ISA-bus memory write cycle for odd (*A0*=1) byte within the *SMP* ISA-bus memory address range. This cycle is also generated in case host PC CPU performs ISA-bus memory read/write cycle for 16-bit words allocated at the even (*A0*=0) address boundary. When this data cycle format is set and host PC CPU performs memory byte accesses to other bytes of *SMP*, then no SB data cycle is generated and data is read/written from/to the on-board bidirectional register transceivers. Actual 16-bit word selection within the addressed SB 32-bit word is performed by ISA-bus address bit *A1*. *SMP* appears as a linear 16Kx16 space of 16-bit words.

32-bit data cycle	0 1	1 1	SB data cycle is generated only when host CPU performs ISA-bus memory read of theleast significant byte (*A0*=*A1*=0) or memory write of the most significant byte (*A0*=*A1*=1) of the 32-bit meory words within *SMP* ISA-bus memory address range. When host CPU accesses other bytes of the *SMP* then no SB data cycle is generated and data is read/written from/to the on-board bidirectional register transceivers. SB data are transferred as 32-bit data words. *SMP* appears as a linear 8Kx32 space of 32-bit words.

Notes: 1. The highlighted configuration corresponds to power on default value.

Data format for host SB data access cycle can be changed by host software during *TORNADO-32L* operation and depends upon user requirements for host and resident DSP software.

8-bit data cycles are the recommended selection when host software either assumes *SMP* to appear as a linear set of bytes (as well as of 16-bit words or 32-bit words), or when host software may require access to any random selected byte of *SMP* data. In this case the SB data cycle is generated each time when host PC CPU or DMA controller perform ISA-bus *SMP* memory access cycle.

16-bit data cycles are the recommended selection when host software assumes *SMP* to appear as a linear set of either 16-bit words or 32-bit words. In this case the SB data cycle is generated when host PC CPU either reads even *SMP* memory bytes or writes to odd *SMP* memory bytes. The 16-bit host data cycle format normally saves about 50% time required for equivalent 8-bit SB access cycles.

32-bit data cycles are the recommended selection when host software assumes *SMP* to appear as a linear set of 32-bit words. In this case the SB data cycle is generated when host PC CPU either reads least significant byte of 32-bit *SMP* memory words of writes to most significant byte of 32-bit *SMP* memory words. The 32-bit data cycle format normally saves about 75% time required for equivalent 8-bit SB access cycles.

SB Locking by ISA-bus Memory Interface

SB locking is used in order to lock SB access when processing software shared semaphores allocated in SRAM bank #0.

SB locking by ISA-bus memory interface may be set by means of setting either the *SB_GLOCK* bit or *SB_LOCK* bit of *CONTROL REGISTER* from ISA-bus I/O interface.

Timing diagrams of SB locking using *SB_GLOCK* and *SB_LOCK* bits are presented at fig. 2-4 and fig. 2-5.

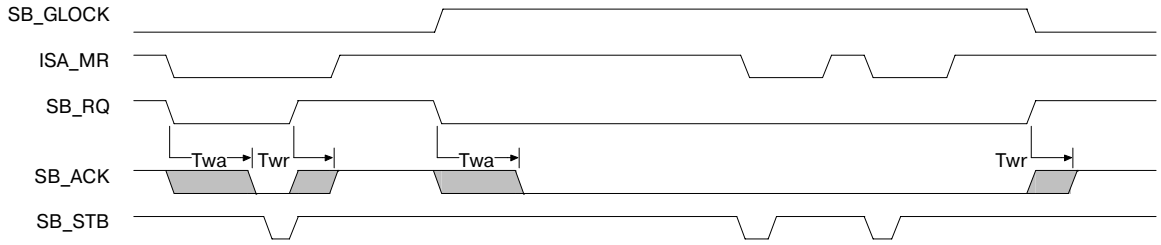

Fig.2-4. Timing diagram for SB access cycle invoked by ISA-bus memory interface with SB locking using *SB_GLOCK* bit.

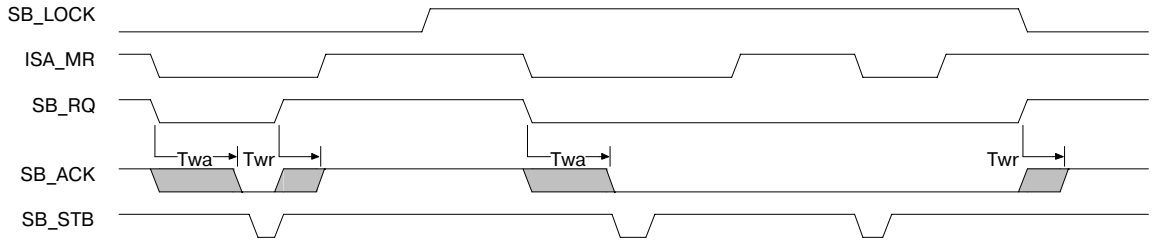

Fig.2-5. Timing diagram for SB access cycle invoked by ISA-bus memory interface with SB locking using *SB_LOCK* bit.

CAUTION

SB locking by ISA-bus memory interface using *SB_GLOCK* and *SB_LOCK* bits of *CONTROL REGISTER* from ISA-bus I/O interface should be used for short-time SB locking.

Continuous SB locking by ISA-bus memory interface using *SB_GLOCK* and *SB_LOCK* bits may result in continuous holding of TMS320C32 DSP (in case it requests access to SB data) and may lead to significant time distortions of real-time data processing.

2.5 Host ISA-bus I/O Interface

TORNADO-32L host ISA-bus I/O interface should be used for *TORNADO-32L* system control, configuring of ISA-bus memory interface and interrupt handshaking between host PC CPU and TMS320C32 DSP.

I/O Base Address of Host ISA-bus I/O Interface

Host ISA-bus I/O interface occupies eight 8-bit registers inside ISA-bus I/O address space. Base address of host ISA-bus I/O interface is defined by means of 3-button on-board DIP switch SW1 (see fig.2-2) in accordance with the predefined settings listed in table 2-5.

Table 2-5. ISA-bus I/O base address for host ISA-bus I/O interface.

ISA-bus I/O base address for ISA-bus I/O interface	button SW1-3	button SW1-2	button SW1-1
300H	OFF	OFF	OFF
310H	OFF	OFF	ON
320H	OFF	ON	OFF
330H	OFF	ON	ON
340H	ON	OFF	OFF
350H	ON	OFF	ON
360H	ON	ON	OFF
370H	ON	ON	ON

Note: 1. The highlighted configuration corresponds to the factory setting.

ISA-bus I/O Interface Registers

List of *TORNADO-32L* ISA-bus I/O interface registers is presented in table 2-6.

Table 2-6. Register set of Host ISA-bus I/O interface.

register #	register address	access mode	description
#0	*BA*+0	r/w	*SB PAGE MAPPER* (*LSB*)
#1	*BA*+1	r/w	*SB PAGE MAPPER* (*MSB*)
#2	*BA*+2	r/w	*CONTROL REGISTER*
#3	*BA*+3	r/w *or* r w	*FLAG DATA REGISTER* *FLAG STATUS REGISTER* *FLAG CONTROL REGISTER*
#403	*BA*+403H	r/w	*FLAG SELECTOR REGISTER*
#4	*BA*+4		reserved (do not use)
#5	*BA*+5		reserved (do not use)
#6	*BA*+6		reserved (do not use)
#7	*BA*+7		reserved (do not use)

Notes:
1. '*BA*' denotes ISA-bus I/O base address of host ISA-bus I/O interface in accordance with table 2-5.
2. Register access modes: *r* - read only, *w* - write only, *r/w* - read/write.

SB PAGE MAPPER Register

Registers #0 и #1 of *TORNADO-32L* ISA-bus I/O interface are least significant byte (LSB) and most significant byte (MSB) of 16-bit *SB PAGE MAPPER* register, which is used to setup *SMP* SB base address in 128Kx32 SB address space in the 8Kx32 (32Kx8 or 16Kx16) increments.

SB PAGE MAPPER LSB register comprises of bits *A13..A16* of SB address. All other bits of *SB PAGE MAPPER LSB* and *SB PAGE MAPPER MSB* registers are ignored on writing and read as zeros. The *A0..A12* bits of SB address for 32-bit SB words within *SMP* are derived from ISA-bus memory address bits *ISA_A2..ISA_A14*, whereas address bits *ISA_A0..ISA_A1* are used to select particular byte or 16-bit word within adressed 32-bit *SMP* SB word.

SB PAGE MAPPER (LSB)

bit-7	bit-6	bit-5	bit-4	bit-3	bit-2	bit-1	bit-0
0	*0*	*0*	*0*	*A16*	*A15*	*A14*	*A13*

SB PAGE MAPPER (MSB)

bit-7	bit-6	bit-5	bit-4	bit-3	bit-2	bit-1	bit-0
0	*0*	*0*	*0*	*0*	*0*	*0*	*0*

CONTROL Register

Register #2 of *TORNADO-32L* ISA-bus I/O interface is called *CONTROL REGISTER*. It used for reset control of on-board TMS320C32 DSP, for configuration of ISA-bus memory interface and for DSP-to-host interrupt communication.

CONTROL REGISTER

bit-7	bit-6	bit-5	bit-4	bit-3	bit-2	bit-1	bit-0
SB_ERROR IE	*MH_RQ_IE*	*SB_CCL-1*	*SB_CCL-0*	*SB_LOCK*	*SB_GLOCK*	*0 (reserved)*	*MRES*

Table 2-7 contains detail description for bits and bit fields of *CONTROL REGISTER*.

Table 2-7. Bits and bit fields of *CONTROL REGISTER*.

bit or bit field of CONTROL REGISTER	***power on defalt state***	***description***
MRES	0	*On-board TMS320C32 DSP reset line* control: • '0' corresponds to RESET ON of the on-board TMS320C32 DSP • '1' corresponds to RESET OFF of the on-board TMS320C32 DSP (i.e. the on-board DSP is in the program execution mode)
SB_GLOCK	0	*Shared Bus Global Lock*. With the *SB_GLOCK*=1 the multiformat SB access controller of ISA-bus memory interface generates active SB locking.
SB_LOCK	0	*Shared Bus Lock*. With the *SB_LOCK*=1 the multiformat SB access controller of ISA-bus memory interface generates active SB locking starting from first next SB request cycle after the *SB_LOCK* bit is set to the *SB_LOCK*=1.
SB_CCL-0, SB_CCL-1	{0,0}	*Shared Bus Cycle format selector for Host-to-SB accesses* : • {0,0} corresponds to 8-bit host SB data cycle • {0,1} corresponds to 16-bit host SB data cycle • {1,0} and {1,1} correspond to 32-bit host SB data cycle
MH_RQ_IE	0	*Master TMS320C32 DSP to Host Request Interrupt Enable*. If *MH_RQ_IE*=1 and *MH_RQ*=1, active interrupt request to host PC is generated. Host PC interrupt request is logical OR between (*MH_RQ_IE & MH_RQ*) and (*SB_ERROR_IE & SB_ERROR*) logic terms.

SB_ERROR_IE	0	*SB Error Interrupt Enable*. If *SB_ERROR_IE*=1 and *SB_ERROR*=1, active interrupt request to host PC is generated. Host PC interrupt request is logical OR between (*MH_RQ_IE & MH_RQ*) and (*SB_ERROR_IE & SB_ERROR*) logic terms.

FLAG Registers

Registers #3 and #403 of *TORNADO-32L* ISA-bus I/O interface are used for auxiliary control (*flags*) of *TORNADO-32L*.

Flags are comprized into a set of multiplexed 8-bit *flag registers*. The particular *flag register* is addressed by *FLAG SELECTOR REGISTER* (register #403). Flags within a currently addressed (selected) *flag register* can be read/write using I/O read/write operation into *FLAG DATA REGISTER* (register #3).

FLAG SELECTOR REGISTER is available for I/O r/w operations has the following data format:

FLAG SELECTOR REGISTER (read/write)

FSEL-7	*FSEL-6*	*FSEL-5*	*FSEL-4*	*FSEL-3*	*FSEL-2*	*FSEL-1*	*FSEL-0*
bit-7	bit-6	bit-5	bit-4	bit-3	bit-2	bit-1	bit-0

A list of available flag registers, which can be addressed by *FLAG SELECTOR REGISTER* in *TORNADO-32L*, is presented in table 2-8.

Table 2-8. Flag registers for *TORNADO-32L*.

code written to FLAG SELECTOR REGISTER	***name of addressed flag register (register #3)***	***description***
00H	r: *SYS_STATUS_FRG* w: *T31_1A_FLAG_CONTROL_RG*	Not recommended for usage in *TORNADO-32L*. This flag register Is provided for compatibility with *TORNADO-31* DSP system rev.1A/1B only. In read mode indicates current status of major run-time systems flags of *TORNADO-32L* (refer to subsection "*SYS_STATUS_FRG Flag Register*" later in this section). In write mode delivers compatibility with *FLAG CONTROL REGISTER* of *TORNADO-31* rev.1A/1B DSP system. The flag codes listed below in this table (10H/20H/30H) can be loaded directly to *FLAG CONTROL REGISTER* in order to generate output flag signals (refer to subsection "*Emulation of TORNADO-31 rev.1A/1B Flag Control Protocol'"* later in this section). This procedure has limited functionality when *TORNADO-32L* is installed into i80286/80386SX based host PC.

10H	r: *SYS_STATUS_FRG* w: *SET_HM_RQ_FRG*	In read mode indicates current status of major run-time systems flags of *TORNADO-32L* (refer to subsection "*SYS_STATUS_FRG Flag Register*" later in this section). In write mode sets active *Host_to_Master_Request (HM_RQ)* via *INT3* external interrupt request input for TMS320C32 DSP. Data written to *SET_HM_RQ_FRG* register has no meaning and is ignored.
20H	r: *SYS_STATUS_FRG* w: *CLEAR_MH_RQ_FRG*	In read mode indicates current status of major run-time systems flags of *TORNADO-32L* (refer to subsection "*SYS_STATUS_FRG Flag Register*" later in this section). In write mode clears active *Master_to_Host_Request (MH_RQ)* flag in *SYS_STATUS_FRG* flag register. *Master_to_Host_Request* flag can be set by resident TMS320C32 DSP software in order to set attention or interrupt request to host PC. Data written to *CLEAR_MH_RQ_FRG* register has no meaning and is ignored. Host PC should perform this operation in the end of corresonding procedure or interrupt handler (in case *MH_RQ_IE*=1) after *MH_RQ* interrupt source has been identified.
30H	r: *SYS_STATUS_FRG* w: *CLEAR_SB_ERROR_FRG*	In read mode indicates current status of major run-time systems flags of *TORNADO-32L* (refer to subsection "*SYS_STATUS_FRG Flag Register*" later in this section). In write mode clears active *SB_ERROR* flag in *SYS_STATUS_FRG* flag register. Active *SB_ERROR* flag is set by timout during host-to-SB access. Data written to *CLEAR_MH_RQ_FRG* register has no meaning and is ignored. Host PC should perform this operation in the end of corresonding interrupt handler (in case *SB_ERROR_IE*=1) after *SB_ERROR* interrupt source has been identified. Also, host PC should perform this operation in the end of data transmission between host ISA-bus and SB in case active *SB_ERROR* flag is detected.
E0H	r/w: *ISA_MI_BADDR_FRG*	*ISA-bus Memory I/F Base Address Register*. Sets ISA-bus memory base address for host ISA-bus memory I/F of *TORNADO-32L* in accordance with table 2-3. Sets/resets activity of host ISA-bus memory I/F of *TORNADO-32L* within UMB area of ISA-bus memory address space. Only three least significant bits of this register are valid; all other bits are ignored and reads as zeros.

E2H	r/w: *ISA_ECC_BADDR_FRG*	*ECC ISA-bus I/O Base Address Register*. Sets ISA-bus I/O base address for *TORNADO-32L* optional on-board emulation controller (*ECC*) in accordance with table 2-11. Sets/resets activity of *ECC* within ISA-bus I/O address space. Only three least significant bits of this register are valid; all other bits are ignored and reads as zeros.
F0H *F1H*	r: *DEV_ID0_FRG* r: *DEV_ID1_FRG*	*Device Identifier and S/N Registers #0/#1*. These registers are read only and contain LSB and MSB of unique device and s/n identifier for *TORNADO-32L*

Notes:
1. Unused codes for flag registers are reserved for future expansion.
2. Access modes: *r* - read only; *w* - write only; *r/w* - read and write.
3. Highlighted configuration corresponds to PC power on default setting.

After a desired flag register has beed selected by loading the corresponding code into *FLAG SELECTOR REGISTER* in accordance with table 2-8, actual input flags status can be obtained by reading *FLAG DATA REGISTER* (*FLAG STATUS REGISTER*), whereas actual output flag settings can be performed by writing to *FLAG DATA REGISTER* (*FLAG CONTROL REGISTER*). Note, that *FLAG STATUS REGISTER* and *FLAG CONTROL REGISTER* are actually read and write denotification of *FLAG DATA REGISTER* that is available for both read and write operations. However, this makes useful sense due since some of *TORNADO-32L* flag registers have different meaning for read and write operations.

SYS_STATUS_FRG Flag Register

SYS_STATUS_FRG flag register (see table 2-8) comprizes of major *TORNADO-32L* run-time system flags settings. It is available as read only register and has the following data format:

FLAG STATUS REGISTER (read only)

SB_ERROR	*MH_RQ*	*0*	*0*	*0*	*0*	*0*	*SB_ACK*
bit-7	bit-6	bit-5	bit-4	bit-3	bit-2	bit-1	bit-0

Detail description for bits of *FLAG STATUS REGISTER* is presented in table 2-9.

Table 2-9. Bits of *FLAG STATUS REGISTER* for *TORNADO-32L*.

bit name	***power on default value***	***description***
SB_ACK	0	*SB Request Acknowledge*. *SB_ACK*=1 denotes that SB arbiter has granted SB access to host ISA-bus memory interface. *SB_ACK* flag can be read during active SB locking from host ISA-bus memory interface. It is used for diagnostic purpose only.
MH_RQ	0	*Master_to_Host Request (MH_RQ)*. *MH_RQ*=1 denotes that TMS320C32 DSP has generated active request to host PC CPU. *MH_RQ* flag will keep staying in active *MH_RQ*=1 state until it will be reset by host PC software by means of writing to *CLEAR_MH_RQ_FRG* flag register. If *MH_RQ_IE*=1 and *MH_RQ*=1, then active host PC CPU interrupt request is generated.
SB_ERROR	0	*SBus Error (SB_ERROR)*. *SB_ERROR*=1 denotes that the 4.2 μsec timeout has been detected during previous SB access from host ISA-bus memory interface. *SB_ERROR* flag will keep staying in active *SB_ERROR*=1 state until it will be reset by host PC software by means of writing to *CLEAR_SB_ERROR_FRG* flag register. If *SB_ERROR* flag is set active (*SB_ERROR*=1), all succeeding host-to-SB requests from host ISA-bus memory interface of *TORNADO-32L* will be ignored and data returned will be undefined. If *SB_ERROR_IE*=1 and *SB_ERROR*=1, an active host PC CPU interrupt request is generated.

Emulation of TORNADO-31 rev.1A/1B Flag Control Protocol

Flag control protocol for *TORNADO-31* rev.1A/1B DSP systems included registers *FLAG CONTROL REGISTER* and *FLAG STATUS REGISTER* only. Register *FLAG SELECTOR REGISTER* did not exist. In order to generate an output flag signal, one should write a corresponding code directly to *FLAG CONTROL REGISTER*.

In order to gain compatibility with flag control protocol for *TORNADO-31* rev.1A/1B DSP systems, *TORNADO-32L* include *T31_1A_FLAG_CONTROL_RG* flag pseudoregister that is available for read only. *T31_1A_FLAG_CONTROL_RG* flag pseudoregister is addressed as default on host power on and when 00H code is written into *FLAG SELECTOR REGISTER*. After *T31_1A_FLAG_CONTROL_RG* pseudoregister has been addressed, then succeeding writing of codes from table 2-10 into *FLAG DATA REGISTER* will generated output flag signals compatible to those for *TORNADO-31* rev.1A/1B DSP systems.

Table 2-10. Output flags that are generated on writes to *T31_1A_FLAG_CONTROL_RG* flag pseudoregister.

code loaded into T31_1A_FLAG_CONTROL_RG	***description***
10H	*Set_Host_to_Master_Request (HM_RQ)*. Set active *Host_to_Master_Request* via *INT3* external interrupt request input for TMS320C32 DSP. (see description for *SET_HM_RQ_FRG* flag register in table 2-8 for details)
20H	*Clear_Master_to_Host_Request (MH_RQ)*. Clear active *Master_to_Host_Request (MH_RQ)* flag in *SYS_STATUS_FRG* flag register. (see description for *CLEAR_MH_RQ_FRG* flag register in table 2-8 for details)
30H	*Clear_Shared_Bus_Error (SB_ERROR)*. Cllears active *SB_ERROR* flag in *SYS_STATUS_FRG* flag register. (see description for *CLEAR_SB_ERROR_FRG* flag register in table 2-8 for details)

In case reading of run-time status for input flags is required, then *SYS_STATUS_FRG* flag register is available for read only during *T31_1A_FLAG_CONTROL_RG* is being addressed and delivers compatibility with reading of input flag status for *TORNADO-31* rev.1A/1B DSP systems.

Flag Registers for Identifying TORNADO-32L DSP System

TORNADO-32L includes *DEV_ID0_FRG/DEV_ID1_FRG* read-only flag registers (see table 2-8), which contains unique code for identification of *TORNADO-32L* DSP systems and its serial number.

Usage of *DEV_ID0_FRG/DEV_ID1_FRG* flag registers in host PC software is recommended for those applications that require to be protected from unauthorized duplication of software.

Generation of Request to TMS320C32 DSP

TORNADO-32L can generate request from host PC to TMS320C32 DSP in order to synchronize between programs execution in host and on-board DSP environments. This is called *HM_RQ* (host to master request) and results in generation of active *INT3* external interrupt request for the on-board TMS320C32 DSP.

In order to generate output *HM_RQ* flag, host PC software has to write to *SET_HM_RQ_FRG* flag register.

Processing of Request from TMS320C32 DSP

TORNADO-32L can generate request from TMS320C32 DSP to host CPU in order to synchronize between program execution in host and on-board DSP environments. This is called *MH_RQ* (master to host request) and results in setting flag *MH_RQ* in *SYS_STATUS_FRG* flag register. *MH_RQ* can

generate active host PC interrupt request in case *MH_RQ_IE* bit in the *CONTROL REGISTER* is set to the *MH_RQ_IE*=1 state.

MH_RQ will remain in active state *MH_RQ*=1 until it will be recognized and reset by host software. In order to reset the *MH_RQ* flag, host PC software has to write to *Clear_Master_to_Host_Request* flag register.

Host Interrupts

Host ISA-bus I/O interface can generate active interrupts to host PC CPU in the following cases:

- when *SB_ERROR* flag is active (*SB_ERROR*=1) in *SYS_STATUS_FRG* flag register and *SB_ERROR_IE* bit in *CONTROL REGISTER* is set to *SB_ERROR_IE*=1 state
- when *MH_RQ* is active (*MH_RQ*=1) in *SYS_STATUS_FRG* flag register and *MH_RQ_IE* bit in *CONTROL REGISTER* is set to *MH_RQ_IE*=1 state.

Host ISA-bus interrupt request is generated as logical 'OR' of the above events. Decoding of interrupt source should be performed by host PC software by means of analyzing the contents of *SYS_STATUS_FRG* flag register.

Host PC interrupt request may be generated using one of nine available ISA-bus interrupt request lines. Particular PC interrupt line is selected by the on-board interrupt configuration jumper J3 (see fig.2-2). The following ISA-bus interrupt request lines are available: IRQ-3, IRQ-4, IRQ-5, IRQ-6, IRQ-7, IRQ-10, IRQ-11, IRQ-12 and IRQ-15.

Setting ISA-bus I/O base address for emulation controller (ECC)

TORNADO-32L provides software setting of ISA-bus I/O base address for optional on-board emulation controller chip (*ECC*) by means of writing to *ISA_ECC_BADDR_FRG* flag register. Only three least significant bits of *ISA_ECC_BADDR_FRG* flag register are valid, and all other bist are ignored on writes and reads as zeroes. Available settings for *ISA_ECC_BADDR_FRG* flag register are presented in table 2-11.

Table 2-11. ISA-bus I/O base address for *TORNADO-32L* optional on-board emulation controller (*ECC*).

ISA-bus I/O base address for *ECC*	**ISA-bus I/O address range for *ECC***	**bit setting for *ISA_ECC_BADDR_FRG* flag register**		
		bit#2	***bit#1***	***bit#0***
ECC is disconnected from host ISA-bus; attachement of external TI XDS510 or MicroLAB' MIRAGE-510D emulator is allowed	-	0	x	x
[2)]240H	*240H ... 25FH*	1	0	0
280H	*280H ... 29FH*	1	0	1
320H	*320H ... 33FH*	1	1	0
340H	*340H ... 35FH*	1	1	1

Note: 1. Highlighted configuration corresponds to host power on default setting.
2. This configuration is used as default by *T32CC.EXE* software utility.

CAUTION

Attachment of external TI XDS510 or MicroLAB' *MIRAGE-510D* emulator is allowed only in case on-board emulation controller (*ECC*) is disconnected from ISA-bus (see table 2-11).

Attachment of external TI XDS510 or MicroLAB' *MIRAGE-510D* emulator while on-board emulation controller (*ECC*) is active is strongly prohibited and may result in damaging emulator and/or *ECC*.

2.6 Serial I/O Expansion Interface (SIOX)

TORNADO-32L architecture provides expansion of the on-board TMS320C32 I/O resources via serial I/O expansion interface (SIOX) site, which is designed to carry compatible optional daughter card modules (see fig.1-1 and fig.2-2). Interface SIOX of *TORNADO-32L* is compatible with SIOX interfaces of all *TORNADO-3x/54x* DSP systems and stand-alone *TORNADO-E/EL/SX* DSP controllers.

Available SIOX daughter cards for *TORNADO-3x/54x* DSP systems include a variety of AD/DA/DIO daughter cards for telecommunication, speech and audio signal processing, industrial and instrumentation applications, and many more.

Description

SIOX interface comprises of signals for TMS320C32 DSP on-chip serial port, timers and interrupts control, as well as DSP reset signal and power supply line. Maximum performance of SIOX serial port is 15 Mbit/sec.

External analog and digital I/O signals for the installed SIOX modules should be attached by means of the on-board SIOX module I/O connector via rear panel of host PC.

SIOX I/F Connector

SIOX I/F connector is and industry standard dual-row 20-pin female header with 0.1"x0.1" pin pattern. SIOX connector pinout is presented at fig.2-9 and signal specifications are listed in table 2-12.

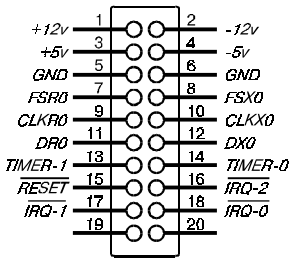

Fig.2-9. SIOX I/F connector pinout (top view).

Table 2-12. SIOX signal specification.

SIOX signal name	signal type	description
DX0, FSX0, CLKX0	I/O/Z	Data, frame synchronization and serial clock signals for TMS320C32 DSP on-chip serial port transmitter. These signals are wired directly to the corresponding TMS320C32 DSP pins.
DR0, FSR0, CLKR0	I/O/Z	Data, frame synchronization and serial clock signals for TMS320C32 DSP on-chip serial port receiver. These signals are wired directly to the corresponding TMS320C32 DSP pins.
TIMER-0, TIMER-1	I/O/Z	Control signals for TMS320C32 DSP on-chip timers #0 and #1. These signals are wired directly to the corresponding TMS320C32 DSP pins.

$\overline{RESET}$	O	Reset signal ($\overline{RESET}$=0) for TMS320C32 DSP.
$\overline{IRQ-0}$, $\overline{IRQ-1}$, $\overline{IRQ-2}$	I	External hardware interrupt request lines ($\overline{IRQ-0}$ has the highest priority) for TMS320C32 DSP. Both edge- and level-triggered interrupts are supported. Selection between edge- and level-triggered interrupts is performed by DSP software by setting *INT CONFIG* bit of TMS320C32 DSP on-chip *ST* status register. The corresponding *INT0...INT2* external DSP interrupts are generates either at low logical level or at falling edge (1→0) of source signals $\overline{IRQ-0}$..$\overline{IRQ-2}$.
GND		Ground.
+5v		+5v power (from ISA-bus).
+12v		+12v power (from ISA-bus).
-5v		-5v power (from ISA-bus).
-12v		-12v power (from ISA-bus).

Note: Signal type is denoted as the following: *I* - input, *O* - output, *Z* - high impedance.

Input logical levels for $\overline{RESET}$ and $\overline{IRQ-0}$/$\overline{IRQ-1}$/$\overline{IRQ-2}$ signals correspond to that for TTL signals. Output logical levels and load currents for *TCLK0/TCLK1* and *DX0/FSX0/CLKX0/DR0/FSR0/CLKR0* signals correspond to that for TTL levels with I_{OL}=2ma and I_{OH}=-0.3ma load currents.

2.7 Software Development Tools

TMS320C32 is now an industry standard DSP and is supported by a variety of software development tools from multiple 3rd party vendors.

Compilers and Debuggers

Software development for *TORNADO-32L* is supported by TI Floating-Point DSP Optimizing C Compiler and Assembly Language Tools.

Debugging of DSP resident software for *TORNADO-32L* is supported by TI C Source Debugger and Code Composer IDE from GoDSP Corp. Both debuggers require on-board optional emulation controller chip (*ECC*) installed and may be shipped together with *TORNADO-32L* DSP system.

Real-time Multitasking Operating Systems (RTOS)

TORNADO-32L is supported by multiple RTOSs that provide multitasking capabilities:

- *Nucleus RTX/PLUS* from Accelerated Technology Inc is a modular single-processor RTOS with powerful task management, inter-task communication. NucleusRTX/PLUS delivers multiple options including real-time debugger, file I/O, net and streams options.

Nucleus RTX/PLUS and all options come with well documented C/Assembler source codes and are royalty free.

- *VIRTUOSO* from Eonic Systems Inc is an industry standard high-performance RTOS and provides full feature multitasking support. It comes standard with capabilities for host file, keyboard and screen text/graphics I/O from DSP environment via *TORNADO-32L* host ISA-bus I/F, and is available with a wide selection of function libraries for DSP, math, matrix, 2D, etc. computations.
- *SPOX* from Spectron Microsystems Inc is an industry standard RTOS for DSP that provides multitasking support. It is available with a selection of function libraries for DSP, math, matrix, etc. computations.

Application Software Tools for TORNADO-32L

Application specific tools for *TORNADO-32L* DSP system include a variety of function libraries for DSP, math, vector, image, etc computation, as well as function libraries for vocoder/fax/modem applications and audio multimedia.

DSP Algorithm Development and Digital Filter Design Tools

TORNADO-32L is supported by DSP algorithm development tools from Hyperception Inc (Hypersignal Block Diagram, RIDE and Code Generator) and digital filter design tools from Momentum Data Systems Inc (QEDesign) and Atlanta Signals Processing Inc (DFDP4/Plus). Both development of DSP algorithm and design of digital filters result in generation of source code for TMS320C32 DSP and can be easily incorporated into user supplied resident software for *TORNADO-32L*.

Chapter 3. Installation and Configuration

This chapter includes instructions for configuring and installation of *TORNADO-32L* DSP system into host PC.

3.1 Setting I/O Base Address for Host ISA-bus I/O Interface

You have to setup ISA-bus I/O base address for host ISA-bus I/O interface of *TORNADO-32L* prior installation of *TORNADO-32L* board into ISA-bus slot of host PC. This procedure should be done while host PC power is switched off.

I/O base address for host ISA-bus I/O interface of *TORNADO-32L* is configured by means of on-board 3-button DIP-switch SW1 (see fig.1-1 and fig.2-2) in accordance with configuration setting in table 2-5.

CAUTION

When setting I/O base address for host ISA-bus I/O interface be sure to check I/O base address for other hardware installed in your host PC in order to avoid I/O address conflicts on ISA-bus.

TORNADO-32L is shipped from factory with I/O base address for host ISA-bus I/O interface in accordance with default settings from table 2-5.

3.2 Setting Interrupt Request Line for Host PC

Setting of host PC interrupt request line is optional procedure for *TORNADO-32L* and should be performed in accordance with requirements of application software that you use together with *TORNADO-32L*. This procedure should be done while host PC power is switched off.

Setting of host PC interrupt request line is provided via on-board host PC interrupt request jumper J3 (see fig2-2). See section 2.5 for details about *TORNADO-32L* host PC interrupt support. *TORNADO-32L* is shipped from factory without PC interrupt request jumper installed.

CAUTION

When setting host CPU interrupt request line be sure to check interrupt requests for other hardware installed in your host PC in order to avoid interrupt request conflicts on ISA-bus.

3.3 Installing the *TORNADO-32L* Mainboard into Host PC

After I/O base address for host ISA-bus I/O interface of *TORNADO-32L* has been configured and host PC interrupt request line has been setup, you can now install *TORNADO-32L* board into 16-bit ISA-bus slot of host PC and screw on-board *TORNADO-32L* mounting bracket to rear panel of host PC. Afterthat, you can safely switch on power of host PC and load operating system of your PC.

3.4 Configuring Memory Base Address for Host ISA-bus Memory Interface

After *TORNADO-32L* has been installed into PC, PC power has been switched on, and DOS has been loaded, you can proceed with configuring ISA-bus memory base address for host ISA-bus memory interface of *TORNADO-32L*.

It is recommended to use *T32CC.EXE* software utility for configuring ISA-bus memory base address for host ISA-bus memory interface of *TORNADO-32L* at DOS prompt.

Setting Memory Base Address for Host ISA-bus Memory Interface

You have to invoke *T32CC.EXE* software utility with *-imXXXXX* command line option in order to setup ISA-bus memory base address for host ISA-bus memory interface of *TORNADO-32L* at DOS prompt in accordance with table 2-3.

The following example sets *D8000H* memory base address for host ISA-bus memory interface of *TORNADO-32L*:

T32CC -imD8000

In case I/O base adddress of host ISA-bus I/O interface of *TORNADO-32L* differs from default value specified in table 2-5, then you have also specify *-ipXXX* command line option (where *XXX* denotes I/O base address of host ISA-bus I/O interface for *TORNADO-32L*) in DOS command line when invoking *T32CC.EXE* software utility. The following example demonstrates how to invoke *T32CC.EXE* software utility for *TORNADO-32L* DSP system with I/O base address for host ISA-bus I/O interface beeing configured to 300H value:

T32CC -imD8000 -ip300

Switching Off Host ISA-bus Memory Interface

The following example demonstrates how to switch off host ISA-bus memory interface of *TORNADO-32L* DSP system:

T32CC -im0

CAUTION

When setting memory base address for host ISA-bus memory interfaces be sure to check memory base address for other hardware installed in your host PC in order to avoid memory address conflicts on ISA-bus.

3.5 Installing SRAM Banks

TORNADO-32L permits installation of 8K..256Kx32 0ws on-board SRAM into the corresponding sockets of two on-board SRAM banks (bank #0 and bank #1). Bank #0 should always be installed on *TORNADO-32L* board since it is used for communication between host PC and on-board DSP environment, whereas bank #1 is optional and may be installed only when needed.

Each of SRAM banks #0 and #1 is designed to accomodate four user installed SRAM chips with SRAM capacity 8K/32K/64K/128Kx8 and access time 15ns. SRAM chips should be either in DIP-28 (8K/32Kx8) or DIP-32 (64K/128Kx8) packages with the package with of 300 MIL. SRAM banks may have different SRAM capacity and different SRAM chips installed.

Setting SRAM Chips Type

When installing SRAM chips into sockets for banks #0 and #1 you have to configure on-board *TORNADO-32L* hardware to recognize those SRAM chips correctly. This is performed by means of on-board jumpers J2 and J1 for SRAM bank #0 and bank #1 correspondingly in accordance with tables 3-1 and 3-2 below.

Table 3-1. Setting SRAM chips type for SRAM bank #0.

SRAM DIP chips capacity for SRAM bank #0	jumper J2
8Kx8 15ns (DIP-28)	2-3
32Kx8 15ns (DIP-28) *64Kx8 15ns(DIP-32)* *128Kx8 15ns (DIP-32)*	1-2

Note: 1. Highlighted configuration corresponds to default factory settings.

Table 3-1. Setting SRAM chips type for SRAM bank #1.

SRAM DIP chips capacity for SRAM bank #1	jumper J1
8Kx8 15ns (DIP-28)	2-3
32Kx8 15ns (DIP-28) *64Kx8 15ns (DIP-32)* *128Kx8 15ns (DIP-32)*	1-2

Note: 1. Highlighted configuration corresponds to default factory settings.

Installing SRAM Chips into SRAM Banks

When installing SRAM DIP chips into on-board SRAM sockets for banks #0 and bank #1 you have to match correct orientation of SRAM DIP chips (see fig.3-1) in order to exclude damages of SRAM chips and/or on-board hardware.

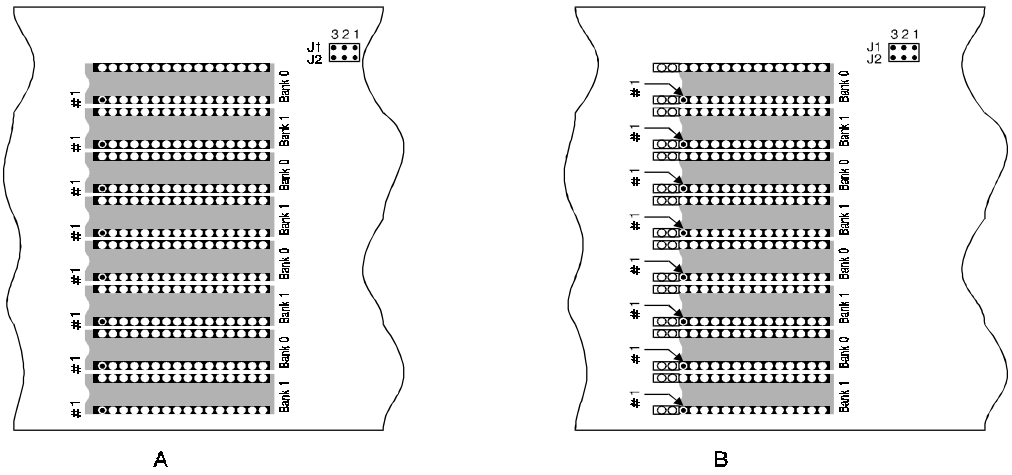

Fig.3-1. Installation of SRAM DIP chips into DIP sockets of *TORNADO-32L* on-board SRAM banks #0 and #1 (A: - installation of SRAM chips in DIP-32 package; B: - installation of SRAM chips in DIP-32 package).

CAUTION

You have to match correct orientation of SRAM DIP chips when installing SRAM DIP chips into *TORNADO-32L* on-board SRAM banks #0 and #1. If you do not match correct orientation of SRAM DIP chips, you can damage SRAM DIP chips and/or *TORNADO-32L* hardware.

3.6 Installation and Configuring Emulation Controller Chip (ECC)

Installation of emulation controller chip (*ECC*) onto *TORNADO-32L* board should be performed with host PC power being switched off.

Installation of ECC

In order to install emulation controller chip (*ECC*) onto *TORNADO-32L* board, unplug *TORNADO-32L* board from host PC ISA-bus slot and install *ECC* into the corresponding socket on *TORNADO-32L* board (see fig. 3-2).

In order to install emulation controller chip (*ECC*) onto *TORNADO-32L* board follow recommendations below (see fig. 3-2):

- switch off power of host PC
- unplug *TORNADO-32L* board from host PC ISA-bus slot
- take *ECC* by your fingers in such a way that its labelling surface looks to you
- adjust *ECC* to be parallel to surface of the corresonding socket on *TORNADO-32L* board
- orient *ECC* in such way, that the key cutted corner of its PLCC-44 package would match the corresponding corner of on-board PLCC-44 socket
- safely insert *ECC* into on-board socket
- safely plug and fix *ECC* in the on-board socket
- install *TORNADO-32L* back into 16-bit ISA-bus slot of host PC
- switch on power of host PC

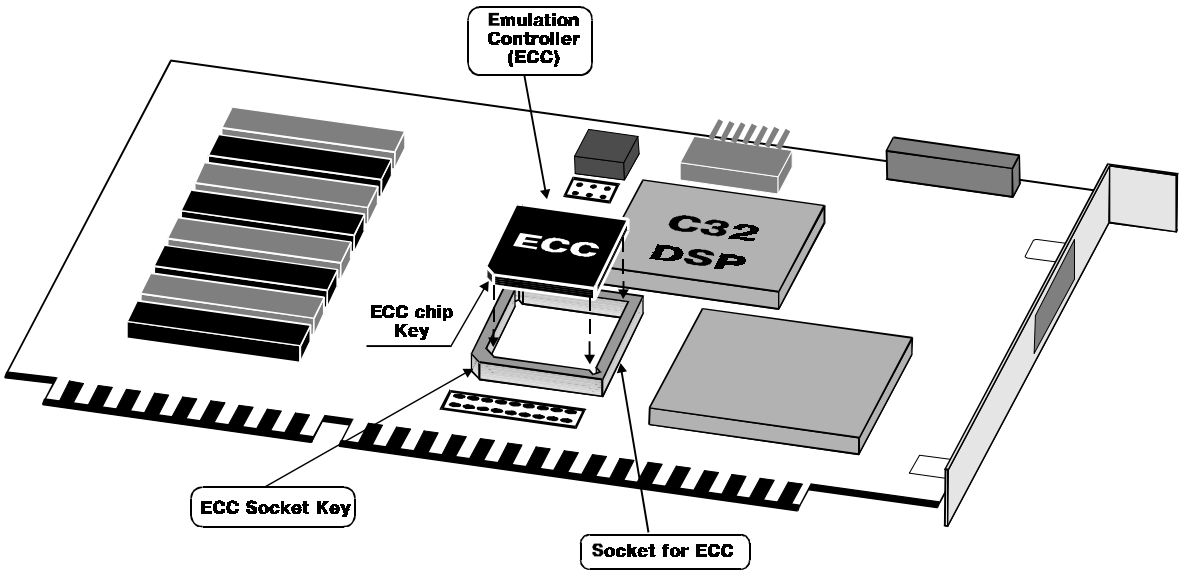

Fig.3-2. Installation of emulation controller chip (*ECC*) onto *TORNADO-32L* board.

Setting I/O Base Address for ECC

You have to invoke *T32CC.EXE* software utility with *-epXXX* command line option in order to setup ISA-bus I/O base address for *ECC* at DOS prompt in accordance with table 2-11.

The following example sets *240H* I/O base address for *ECC*:

T32CC -ep240

In case I/O base adddress of host ISA-bus I/O interface of *TORNADO-32L* differs from default value specified in table 2-5, then you have also specify *-ipXXX* command line option (where *XXX* denotes I/O base address of host ISA-bus I/O interface for *TORNADO-32L*) in DOS command line when invoking *T32CC.EXE* software utility. The following example demonstrates how to invoke *T32CC.EXE* software utility for *TORNADO-32L* DSP system with I/O base address for host ISA-bus I/O interface beeing configured to 300H value:

T32CC -ep240 -ip300

Switching ECC Out from ISA-bus I/O Address Space

The following example demonstrates how to switch *ECC* out from ISA-bus I/O address space:

T32CC -ep0

CAUTION

When setting I/O base address for *ECC* be sure to check I/O base address for other hardware installed in your host PC in order to avoid I/O address conflicts on ISA-bus.

Chapter 4. Utility Programs

This chapter contains description of utility software for *TORNADO-32L* DSP system.

4.1 T32CC.EXE Control Program

T32CC.EXE ("*TORNADO-32L* Control Center") software utility program is the DOS command line control tool for *TORNADO-32L* that delivers simple and powerful user control of the *TORNADO-32L* hardware. *T32CC.EXE* utility offers user the following functionality:

- set and indicate contents of all registers of *TORNADO-32L* host ISA-bus I/O interface
- configure *TORNADO-32L* host ISA-bus memory interface
- read/write to on-board SRAM bank #0 via *TORNADO-32L* host ISA-bus memory interface
- configure emulation controller *ECC*.

T32CC.EXE utility should be invoked from DOS with up to ten command line options specified:

T32CC [*-option1*] [*-option2*] [*-option3*] *...*

Each command line option corresponds to specific *TORNADO-32L* hardware control operation. The following lists all available command line options for *T32CC.EXE* utility. Note, that some command line options are specified with equivalent command line options in brackets that correspond to old command line options for *T31CC.EXE* utility for *TORNADO-31* DSP systems rev.1A/1B.

System Control via CONTROL REGISTER

-c (-cd)	Display current contents of *CONTROL REGISTER*.
-cc (-ccd)	Display current setting for host SB data cycle format, which is specified by {*SB_CCL-0,SB_CCL-1*} bit field of *CONTROL REGISTER*.
-ccb	Set 8-bit (byte) format for host SB data cycle. Corresponds to setting of {*SB_CCL-0,SB_CCL-1*} bit field of the *CONTROL REGISTER* to the {0,0} state.
-cch	Set 16-bit (halfword) format for host SB data cycle. Corresponds to setting of {*SB_CCL-0,SB_CCL-1*} bit field of *CONTROL REGISTER* to the {1,0} state.
-ccw	Set 32-bit (word) format for host SB data cycle. Corresponds to setting of {*SB_CCL-0,SB_CCL-1*} bit field of *CONTROL REGISTER* to the {0,1} state.
-cg (-cgd)	Display current state of *SB_GLOCK* bit (global SB locking by host ISA-bus memory interface) of *CONTROL REGISTER*.
-cg0	Clear *SB_GLOCK* bit of *CONTROL REGISTER* and unlock SB.

-cg1	Set *SB_GLOCK* bit of *CONTROL REGISTER* for immediate active SB locking.
-cl (-cld)	Display current state of *SB_LOCK* bit (SB locking by host ISA-bus memory interface) of *CONTROL REGISTER*.
-cl0	Clear *SB_LOCK* bit of *CONTROL REGISTER* and unlock SB.
-cl1	Set *SB_LOCK* bit of *CONTROL REGISTER* and issue active SB locking during nearest SB access from host ISA-bus memory interface.
-cie (-cied)	Display current state of *SB_ERROR_IE* bit (host interrupt enable on SB error) of *CONTROL REGISTER*.
-cie0	Clear *SB_ERROR_IE* bit of *CONTROL REGISTER* and disable host interrupts on SB error.
-cie1	Set *SB_ERROR_IE* bit of *CONTROL REGISTER* and enable host interrupts on SB error.
-cim (-cimd)	Display current state of *MH_RQ_IE* bit (host interrupt enable on requests from TMS320C32 DSP) of *CONTROL REGISTER*.
-cim0	Clear *MH_RQ_IE* bit of *CONTROL REGISTER* and disable host interrupts on requests from TMS320C32 DSP.
-cim1	Set *MH_RQ_IE* bit of *CONTROL REGISTER* and enable host interrupts on requests from TMS320C32 DSP.
-cr (-crd)	Display current state of reset signal for TMS320C32 DSP, which is specified by *MRES* bit of *CONTROL REGISTER*.
-cr0	Remove reset signal for TMS320C32 DSP, i.e. put DSP into "RUN" state. This option sets *MRES* bit of *CONTROL REGISTER*.
-cr1	Apply reset signal for TMS320C32 DSP, i.e. put DSP into "RESET" state. This option clears *MRES* bit of *CONTROL REGISTER*.

Flag Registers Control

-fsr	Display contents of *FLAG SELECTOR REGISTER*.
-fsrXX	Select flag register #*XX (hex)*, i.e. load *XX* 8-bit hex data into *FLAG SELECTOR REGISTER*.
-fr	Display contents of currently selected flag register (display *FLAG STATUS REGISTER*). The number of currently selected flag register is defined by the contents of *FLAG SELECTOR REGISTER*.
-frXX	Loads *XX* 8-bit бит hex data into the cunntly selected flag register (load *FLAG CONTROL REGISTER*). The number of currently

selected flag register is defined by the contents of *FLAG SELECTOR REGISTER*..

-frs (-fd) Display contents of *SYS_STATUS_FRG* flag register.

-fe (-fed) Display current state of *SB_ERROR* flag from *SYS_STATUS_FRG* flag register.

-fe0 Clear *SB_ERROR* flag, i.e. write to *CLEAR_SB_ERROR_FRG* flag register.

-fb (-fbd) Display current state of *SB_ACK* flag from *SYS_STATUS_FRG* flag register.

-fh (-fhd) Display current state of *MH_RQ* flag from *SYS_STATUS_FRG* flag register.

-fh0 Clear *MH_RQ* flag, i.e. write to *CLEAR_MH_RQ_FRG* flag register.

-fm1 Generate interrupt request to TMS320C32 DSP, i.e. write to *SET_HM_RQ_FRG* flag register.

-frdi Display *TORNADO-32L* device ID and s/n, i.e. display contents of *DEV_ID0_FRG* and *DEV_ID1_FRG* flag registers.

SB Access Control

-ba (-bad) Display contents of *SB PAGE MAPPER* register that defines *SMP* SB base address for host-to-SB access.

-baIXXXXXX Load *SB PAGE MAPPER* register with the *SMP* SB base address that corresponds to *XXXXXX* hex SB address of 32-bit SB data word.

-bdSA,EA Display SB data in 32-bit data format. The *SA* and *EA* parameters specify hex SB starting and ending addresses of 32-bit SB data words correspondingly. Final contents of *SB PAGE MAPPER* register will be set to the *SMP* SB base address that corresponds to *EA* address.

-bdSA@bS,EA@bE Display SB data in 8-bit (byte) data format. The *SA* and *EA* parameters specify hex SB starting and ending addresses of 32-bit SB data words correspondingly whereas *bS* and *bE* parameters specify byte offsets (0..3) inside the 32-bit starting and ending data words correspondingly. Final contents of *SB PAGE MAPPER* register will be set to the *SMP* SB base address that corresponds to *EA* address.

-bdSA@hS,EA@hE Display SB data in 16-bit (halfword) data format. The *SA* and *EA* parameters specify hex SB starting and ending addresses of 32-bit SB data words correspondingly whereas *hS* and *hE* parameters specify halfword offsets (0..1) inside the 32-bit starting and ending data words correspondingly. Final contents of *SB PAGE MAPPER*

register will be set to the *SMP* SB base address that corresponds to *EA* address.

-bwAAAAAA,XXXXXXXX Write 32-bit *XXXXXXXX* hex data word at *AAAAAA* hex SB address. The *AAAAAA* parameter defines SB address of 32-bit SB word. *SB PAGE MAPPER* register will be set to the *SMP* SB base address that corresponds to *AAAAAA* address. (see section 2.4)

-bwAAAAAA@bS,XX Write 8-bit *XX* hex data into *bS*-th byte (0..3) of 32-bit SB data word at *AAAAAA* hex SB address. *SB PAGE MAPPER* register will be set to the *SMP* SB base address that corresponds to *AAAAAA* address.

-bwAAAAAA@hS,XXXX Write 16-bit *XXXX* hex data (halfword) into *hS*-th halfword (0..1) of 32-bit SB data word at *AAAAAA* hex SB address. *SB PAGE MAPPER* register will be set to the *SMP* SB base address that corresponds to *AAAAAA* address.

Setting I/O and Memory Base Addresses for Host ISA-bus Interface

-im Display ISA-bus memory base for host ISA-bus memory I/F of *TORNADO-32L* in accordance with table 2-3 (display and interprete contents of *ISA_MI_BADDR_FRG* flag register).

-imXXXXX Set *XXXXX* hex ISA-bus memory base address for host ISA-bus memory interface of *TORNADO-32L* in accordance with table 2-3 (load *ISA_MI_BADDR_FRG* flag register). If *T32CC.EXE* utility is invoked with *-bd* or *-bw* command line options and option *-im* is not specified (or *ISA_MI_BADDR_FRG* flag register was not loaded previously), then the default *D8000H* ISA-bus memory base address will be used for host ISA-bus memory I/F during host-to-SB access, and option *-im0* will be automatically executed on exit from *T32CC.EXE* utility in order to deactivate host ISA-bus memory I/F afterthat.

-im0 Deactivates host ISA-bus memory I/F of *TORNADO-32L*, i.e. removes it from ISA-bus memory address on exit from *T32CC.EXE* utility.

-ipXXX Specifies *XXX* hex I/O base address for host ISA-bus I/O interface. If this option is omitted, then default factory setting in accordance with table 2-5 will is used.

Control for Emulation Controller (ECC)

-ep Display ISA-bus I/O base for emulation controller (*ECC*) in accordance with table 2-11 (display and interprete contents of *ISA_ECC_IO_BADDR_FRG* flag register).

-epXXX — Set *XXX* hex I/O base address for emulation controller (*ECC*) in accordance with table 2-11 and activate it, i.e. include ECC into ISA-bus I/O address space and connect it to scan-path interface of TMS320C32 DSP. Once *T32CC.EXE* utility has been invoked with this command line option, attachment of external TI XDS510 or MicroLAB' MIRAGE-510D emulator is not allowed to *TORNADO-32L* board unless *T32CC.EXE* utility will be invoked with *-ie0* command line option.

-ei — Set default I/O base address for emulation controller (*ECC*) in accordance with table 2-11 and activate *ECC*, i.e. include ECC into ISA-bus I/O address space and connect it to scan-path interface of TMS320C32 DSP. In case *D_OPTIONS* DOS system variable for TI C3x C Source Debugger is set and its list includes *-pXXX* option, then *XXX* hex I/O address will be used as default I/O base address for *ECC* instead of that in accordance with table 2-11. Once *T32CC.EXE* utility has been invoked with this command line option, attachment of external TI XDS510 or MicroLAB' MIRAGE-510D emulator is not allowed to *TORNADO-32L* board unless *T32CC.EXE* utility will be invoked with *-ie0* command line option.

-ep0 (-ex) — Deactivates *ECC* controller, i.e. remove it from ISA-bus I/O address space and diconnect it from scan-path interface of TMS320C32 DSP. Once *T32CC.EXE* utility has been invoked with this command line option, attachment of external TI XDS510 or MicroLAB' MIRAGE-510D emulator is allowed to *TORNADO-32L* board.

-er — Perform software reset of *ECC*. This is identical to a software reset performed by the M30RESET.EXE utility program from the TI TMS320C3x C Source Debugger software toolkit. Recommended on invocation and exit from TI TMS320C3x C Source Debugger.

Utility Options

-p — Set page-by-page display mode. The "ESC" keypress terminates display output whereas any other keypress results in the next page display.

-? — Display list of available options for *T32CC.EXE* utility program. Help list is also displayed when *T32CC.EXE* utility program is invoked without command line options.

T32CC.EXE utility processes command line options in accordance with the following priority list:

1. *CONTROL REGISTER* control options
2. *FLAG STATUS REGISTER* and *FLAG CONTROL REGISTER* control options
3. *ECC* control options

4. SB access control options.

T32CC.EXE utility returns DOS *exit code* in case it is invoked with *-im*, *-c*, *-cc*, *-cg*, *-cl*, *-cie*, *-cim*, *-fsr*, *-fr*, *-frs*, *-fe*, *-fb*, and *-fh* command line options,which correspond to display of contents of registers, bits and flags of *TORNADO-32L* host ISA-bus interface. The exit code returned corresponds to current value or contents of last displayed bit, bit field, flag or register. Exit code is useful when *T32CC.EXE* utility is integrated into DOS batch (.BAT) file that provides conditional processing. Exit code of *T32CC.EXE* utility program can be analyzed using succeeding '*IF ERRORLEVEL*' DOS batch file commands. The following example of DOS batch file performs conditional processing of *SB_ERROR* flag of *TORNADO-32L*:

```
...
T32CC -fed
IF ERRORLEVEL 1  T32CC -fe0
...
```

When multiple data display options for the *T32CC.EXE* utility are specified, then the returned exit code will correspond to the last processed data display command line option.

In case error is detected by *T32CC.EXE* utility, then the exit code '255' is returned. If *T32CC.EXE* utility is invoked without any data display command line options and no errors is detected, than the exit code '0' is returned.

4.2 *BSF*-files

TORNADO-32L utility software uses binary files with sectional data structure (*binary section format*, *BSF*-files) for high speed program/data upload from host PC disk drive to the on-board SRAM and TMS320C32 DSP environment.

BSF-files data transfer between host PC disk drive and on-board *TORNADO-32L* SRAM and TMS320C32 DSP environment is performed at full speed without utilization of *ECC* emulation emulation controller and TMS320C32 on-chip scan-path interface.

TORNADO-32L utility software tools assume conversion of source software program and data modules for TMS320C32 DSP to *BSF*-file format. Source software modules for TMS320C32 DSP can be obatinde in TI COFF format using TI floating-point DSP C and Assembly compilers or can appear in both source binary and hex formats. *BSF*-files can be iteratively linked from multiple different source modules.

Structure of BSF-files

BSF-file consists of multiple sequential data sections that are actually the binary data for continuous target memory sections. Each section should be loaded at specified target address. Length of each data section and number of sections are not limited.

BSF-files have smaller total file length due to removal of target memory inter-sectional information and due to the binary data format used.

BSF-files have advantages against Intel MCS-86 HEX, Motorola-S, Tektronix HEX, TI COFF, etc. file with sectional format since they do not require run-time interpreting and conversion of data, have smaller total length and can be uploaded directly from host disk drive at the highest speed available.

The *BSF*-file comprises of *global file descriptor* that is located in the very beginning of *BSF*-file, and of succeeding binary data sections each containing *section descriptor* and binary data (fig.4-1).

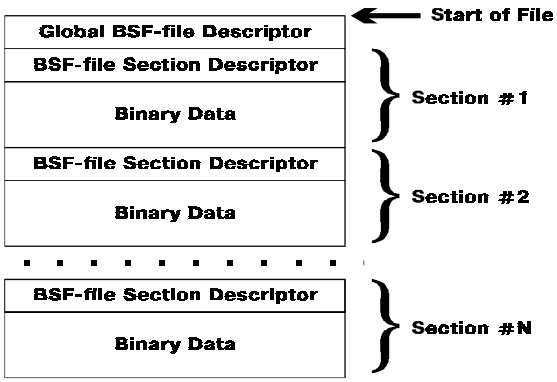

Fig.4-1. Structure of *BSF*-file comprising of *N* sections.

Global descriptor specifies number of sections in *BSF*-file and includes information required for identification and integrity control.

Each section descriptor specifies data words format, section length (in word units), start address and includes information required for identification and integrity control.

Tables 4-1 and 4-2 list structures of the global section descriptor for *BSF*-files.

Table 4-1. Structure of global descriptor for *BSF*-file.

byte number	***data type***	***field identifier***	***description***
0	Character	*CS*	Check sum of global file descriptor data starting from the *ID_Rev* field and up to the *File_Length* field. Check sum is defined as the inverse module 256 binary sum of all bytes included.
1-3	Character	*ID_Rev*	Global file descriptor identifier. This field should read and write as 10H, 0H, 1H bytes.
4-7	Long Integer	*Sections*	Number of *BSF*-sections.
8-15	Double Long Integer	*File_Length*	File length in bytes.

Table 4-2. Structure of section descriptor for *BSF*-file.

byte number	***data type***	***field identifier***	***description***
0	Character	*CS*	Check sum of section descriptor data starting from the *ID_Rev* field and up to the *Section_Length* field. Check sum is defined as the inverse module 256 binary sum of all bytes included.
1-3	Character	*ID_Rev*	Section descriptor identifier. This field should read and write as 80H, 0H, 1H bytes.
4-5	Integer	*Word_Data_Bits*	Data word length in bits. Values x8 (8, 16, 24, 32, etc) are valid.
6-13	Double Long Integer	*Load_Address*	Specifies 64-bit target start address of the section data. Address is specified in the word units.
13-21	Double Long Integer	*Section_Length*	Section length in word units.

4.3 Creating and Editing of *BSF*-files

BSF-files can be created from source HEX- and BIN-files using *H32BSF.EXE* and *BINBSF.EXE* software converters. You can provide conversion from the following source data files :

- from source *.H32 files that should be in Intel MSC-86 HEX (Extended Intel HEX) with 32-bit address/data words (using the *H32BSF.EXE* software converter)
- from source *.OUT TI COFF files that are generated by TI floating point DSP C and Assembler compilers (using the *H32BSF.EXE* software converter)
- from source *.BIN binary files (using the *BINBSF.EXE* software converter).

All software converters support append mode (command line option *-a*) in order to append new data sections to the end of existing *BSF*-file. This allows to assemble multisectional *BSF*-files from different source files with different source data formats.

BSF-file software editor is used in order to list the *BSF*-file directory and to remove undesired data sections.

Conversion of Intel MCS-86 HEX files (.H32) into BSF-files (.BSF)

H32BSF.EXE software converter provides conversion of source .H32 files in Intel MCS-86 HEX (Extended Intel HEX) format with 32-bit address/data words into the corresponding *BSF*-files

**.H32* $\rightarrow$ **.BSF*

Since Intel MCS-86 HEX format is the industry standard format and is supported by almost all compilers, the *H32BSF.EXE* utility may be used for conversion of source program modules generated by different compilers for TMS320C3x DSP chips.

H32BSF.EXE software converter should be invoked from DOS command line:

H32BSF in_file[*.H32*] *out_file*[*.BSF*] [*-a*]

where:

in_file	denotes source Intel MCS-86 HEX (Extended Intel HEX) file with 32-bit address/data words. If file extension is not specified, default .H32 file extension is assumed.
out_file	output *BSF*-file. If file extension is not specified, default .BSF file extension is assumed.
-a	command line option, which specifies that the converted source data sections should be appended to the end of existing *BSF*-file. If output *BSF*-file do not exist, then *-a* option is ignored and new *BSF*-file is created. If *-a* option is not specified, then existing *BSF*-file will be overwritten.

In case no errors are detected by *H32BSF.EXE* software converter, then exit code '0' will be returned, otherwise exit code '1' will be returned.

Conversion of TI COFF files (.OUT) into BSF-files (.BSF)

Conversion of TI COFF files (.OUT), that are generated by TI floating point DSP C and Assembler compilers, into the corresponding *BSF*-files can be performed by means of intermediate conversion of source COFF files into Intel MCS-86 HEX files .H32 and further conversion into *BSF*-file using TI *H32BSF.EXE* software converter:

**.OUT* → **.H32* → **.BSF*

In order to convert source TI COFF files (.OUT) into Intel MCS-86 HEX files (.H32) with 32-bit address/data words the HEX30.EXE utility, which is included with TI floating point DSP C and Assembler compiler tools (version 4.50 or later):

HEX30 -I -memwidth 32 -romwidth 32 IN_FILE.OUT -o OUT_FILE.H32

To get more information about HEX30.EXE utility refer to the following documentation:

Hex Conversion Utility. Addendum to the TMS320 Floating-Point DSP Assembly Language Tools User's Guide. Texas Instruments Inc, SPRU081, USA, 1992.

Final conversion of .H32 file into the corresponding *BSF*-file should be done with the *H32BSF.EXE* software converter (refer to the corresponding subsection of this section).

Conversion of Binary Files (.BIN) into the BSF-files (.BSF)

Conversion of binary source files .BIN with different source data formats (8-bit bytes, 16-bit and 32-bit words) into the corresponding *BSF*-files *.BSF*

**.BIN → *.BSF*

may be performed using *BINBSF.EXE* software converter that is included with utility software for *TORNADO-32L*.

BINBSF.EXE software converter provides conversion of source .BIN binary file and generates only one section into the output *BSF*-file. The *-a* command line option may be used in order to append new sections to existing *BSF*-files.

BINBSF.EXE software converter is useful for including large binary data tables into *BSF*-files, as well as for conversion of source program or data modules in binary format into *BSF*-files for *TORNADO-32L.*

BINBSF.EXE software converter should be invoked from DOS command line:

BINBSF in_file[*.BIN*] *out_file*[*.BSF*] *-fb/-fw16/-fw32 -lXXXXXXX* [*-a*]

where:

in_file	denotes source binary file. If file extension is not specified, default .BIN file extension is assumed.
out_file	output *BSF*-file. If file extension is not specified, default .BSF file extension is assumed.
-fb	specifies that source data are in 8-bit (byte) data word format. This option is alternative to *-fw16* and *-fw32* options.
-fw16	specifies that source data are in 16-bit data word format. In case source data contains non integer number of 16-bit data words, then the generated section for *BSF*-file will be supplied with corresponding number of zero bytes. This option is alternative to *-fb* and *-fw32* options.
-fw32	specifies that source data are in 32-bit data word format. In case source data contains non integer number of 32-bit words, than the generated section for *BSF*-file will be supplied with corresponding number of zero bytes. This option is alternative to *-fb* and *-fw16* options.
-lXXXXXXX	specifies hex starting (loading) address *XXXXXXX* of converted data in the address space of *TORNADO-32L* on-board TMS320C32 DSP chip. Starting address is specified in units of selected data words (defined by *-fb/-fw16/-fw32* options). In case data format is defined as 8-bit (byte) format (option *-fb*), then starting address is the address of first byte of section data in continuous bytes address space. In case data format is defined as 16-bit word format (option -

fw16), then starting address is the address of first 16-bit word of section data in continuous 16-bit words address space. In case data format is defined as 32-bit word format (option *-fw32*), then starting address is the address of first 32-bit word of section data in continuous 32-bit words address space.

-a specifies that converted source data section will be appended to the end of existing output *BSF*-file. If output *BSF*-file does not exist, then *-a* option is ignored and new *BSF*-file will be created. If the *-a* option is not specified, then existing *BSF*-file will be overwritten.

In case no errors are detected by *BINBSF.EXE* software converter, then exit code '0' will be returned, otherwise exit code '1' will be returned.

Listing Directory and Editing BSF-file

When using *BSF*-files you may want to list *BSF*-file directory or to remove undesired data section from *BSF*-file. This may be performed using *BSFEDIT.EXE* software editor that is included with utility software for *TORNADO-32L*.

BSFEDIT in_file[*.BSF*] *-dN*

where:

in_file denotes source and output *BSF*-file. If file extension is not specified, default .BSF file extension is assumed.

-dN specifies that *N*-th data section should be removed from *BSF*-file. *N* value should be not greater than number of sections in *BSF*-file.

In order to display directory of *BSF*-file, the *BSFEDIT.EXE* utility should be invoked with source *BSF*-file name specified only.

In case no errors are detected by *BSFEDIT.EXE* utility, then exit code '0' will be returned, otherwise exit code '1' will be generated.

4.4 Uploading *BSF*-files

Uploading of *BSF*-files into *TORNADO-32L* on-board SRAM on-board TMS320C32 DSP on-chip environment can be performed by means of *T32BSFLD.EXE* software utility, that is included with utility software for *TORNADO-32L*. *T32BSFLD.EXE* utility loads *BSF*-file into *TORNADO-32L* environment via host ISA-bus memory interface without utilization of emulation controller *ECC*.

BSF-file can be uploaded into the *TORNADO-32L* environment using different modes:

- *standard mode*, i.e. when data is uploaded to on-board SRAM via host ISA-bus memory interface without affecting TMS320C32 DSP chip reset line and SB locking
- *reset mode*, i.e. when data is uploaded to on-board SRAM and TMS320C32 DSP on-chip environment via host ISA-bus memory interface while holding TMS320C32 DSP in 'RESET' state

- *global SB locking mode*, i.e. when data is uploaded to on-board SRAM via host ISA-bus memory interface using global SB locking
- *SB locking mode*, i.e. when data is uploaded to on-board SRAM via host ISA-bus memory interface using the SB locking.

All modes except for *reset mode* provide uploading of *BSF*-file into SRAM only. However, these modes do not effect reset signal for TMS320C32 DSP, and data can be uploaded in parallel with TMS320C32 DSP running.

Reset mode provides uploading of *BSF*-file into both on-board SRAM and TMS320C32 DSP on-chip environment (including DSP on-chip memory and peripherals). This is performed by means of using run-time TMS320C32 loader that is loaded into on-board SRAM and then removed automatically by *T32BSFLD.EXE* utility each time loader recognizes that *BSF*-file data section should be loaded into the DSP on-chip resources.

Uploading of *BSF*-file into *TORNADO-32L* is performed by invoking *T32BSFLD.EXE* utility from DOS command line:

T32BSFLD FILE.BSF [*-option1*] [*-option2*] [*-option3*] *...*

The following is list of command line options for *T32BSFLD.EXE* utility, which are grouped into several functional groups:

Upload Mode Control

-/r — Set *RESET* mode for uploading of *BSF*-file. *BSF*-file is uploaded while holding TMS320C32 DSP in 'RESET' state by means of clearing *MRES* bit of *CONTROL REGISTER*. This mode is used for uploading of source program/data modules and supports uploading into both on-board SRAM and TMS320C32 DSP on-chip memory and peripherals. TMS320C32 DSP can be placed into the 'RUN' state on exit from *T32BSFLD.EXE* utility using *-cr0* command line option. The *-/g* option is used as default if none of *-/g*, *-/l* and *-/n* options is specified.

-/g — Set *GLOBAL SB LOCKING* mode for uploading of *BSF*-file. *BSF*-file is uploaded into on-board SRAM while holding SB locking by means of setting *SB_GLOCK* bit of *CONTROL REGISTER*. TMS320C32 DSP will not be able to access SRAM until uploading will be finished. TMS320C32 DSP on-chip resources cannot be loaded in this mode. This mode is normally used for uploading of shared data into on-board SRAM while TMS320C32 DSP is executing a program.

-/l — Set *SB LOCKING* mode for uploading of *BSF*-file. *BSF*-file is uploaded into on-board SRAM while holding SB locking by means of setting *SB_LOCK* bit of *CONTROL REGISTER*. TMS320C32 DSP will not be able to access SRAM until uploading will be finished. TMS320C32 DSP on-chip resources cannot be loaded in

this mode. This mode is normally used for uploading of shared data into on-board SRAM while TMS320C32 DSP is executing a program.

-ln — Set *STANDARD* mode for uploading of *BSF*-file. *BSF*-file is uploaded without affecting 'RESET' state of TMS320C32 DSP and without SB locking. TMS320C32 DSP will be able to access on-board SRAM during uploading of *BSF*-file. The on-board TMS320C32 DSP on-chip resources cannot be loaded during this mode. This mode is normally used for uploading of run-time program or data into on-board SRAM while on-board TMS320C32 DSP chip is executing a program.

-xi — Exclude uploading of TMS320C32 DSP on-chip memory and peripherals when using *RESET* mode for uploading. This option should be used together with *-lr* option only.

Setting Format of Host SB Data Cycle

-bcb — Set 8-bit (byte) format for host SB data cycle. Corresponds to setting of {*SB_CCL-0,SB_CCL-1*} bit field of the *CONTROL REGISTER* to the {0,0} state. This option is used as default when none of *-bch* and *-bcw* options is specified.

-bch — Set 16-bit (halfword) format for host SB data cycle. Corresponds to setting of {*SB_CCL-0,SB_CCL-1*} bit field of *CONTROL REGISTER* to the {1,0} state.

-bcw — Set 32-bit (word) format for host SB data cycle. Corresponds to setting of {*SB_CCL-0,SB_CCL-1*} bit field of *CONTROL REGISTER* to the {0,1} state.

Restarting TMS320C32 DSP on Exit

-cr0 — Remove reset signal for TMS320C32 DSP and put DSP into the program execution state state on exit from *T32BSFLD.EXE* utility. This option corresponds to setting *MRES* bit from *CONTROL REGISTER*.

Setting Base Addresses of ISA-bus Memory and I/O Interfaces

-imXXXXX — Set *XXXXX* hex ISA-bus memory base address for host ISA-bus memory interface of *TORNADO-32L* in accordance with table 2-3 (load *ISA_MI_BADDR_FRG* flag register). If *T32CC.EXE* utility is invoked with *-bd* or *-bw* command line options and option *-im* is not specified (or *ISA_MI_BADDR_FRG* flag register was not loaded previously), then the default *D8000H* ISA-bus memory base address will be used for host ISA-bus memory I/F during host-to-SB access,

and option *-im0* will be automatically executed on exit from *T32CC.EXE* utility in order to deactivate host ISA-bus memory I/F afterthat.

-im0 — Deactivates host ISA-bus memory I/F of *TORNADO-32L*, i.e. removes it from ISA-bus memory address on exit from *T32CC.EXE* utility.

-ipXXX — Specifies *XXX* hex I/O base address for host ISA-bus I/O interface. If this option is omitted, then default factory setting in accordance with table 2-5 will is used.

Utility Options

-? — Display list of available options for *T32BSFLD.EXE* utility. Help list is also displayed when *T32BSFLD.EXE* utility is invoked without command line options and parameters.

In case no errors are detected by *T32BSFLD.EXE* utility, then exit code '0' will be returned, otherwise exit code '1' will be generated.

CAUTION

If *T32BSFLD.EXE* utility is used with *-lr* command line option (or when *-lg*, *-ll* and *-ln* options are not specified) and if either emulation controller (*ECC*) is installed or any of TI XDS510 or MicroLAB' *MIRAGE-510D* emulator is attached, then the following error message may appear:

error: missing DSP handshaking

This error message states that the TMS320C32 DSP cannot be initialized correctly during uploading of TMS320C32 DSP on-chip memory or peripherals. This problem is caused by DSP on-chip execution controller that is locked by attached emulator or *ECC*.

In order to avoid this problem you have to reset *ECC* or attached emulator. The emulator can be reset using the supplied software reset utility whereas *ECC* can be reset by invoking *T32CC.EXE* utility program with *-er* command line option.

Dear CUSTOMER !

MicroLAB Systems Ltd is enhancing its technical documentation and will be very thankful for your opinion about documentation for the device you have purchased. Please take a while and fill in the questionnaire below and fax or mail it to MicroLAB Systems Ltd. *Thank you for your assistance!*

- *How do you use this documentation?*
  in order to search for specific information when required only
  has read only certain chapters and sections
  has read from front to back and always use it
- *What is your values of this documentation?*
  not enough detail documentation
  too detail documentation
  suffers of imbalance of details
  perfectly corresponds to my requirements
- *What topics should be more detail?*
- *Please list information that was difficult to find and why?*
- *Did you find any errors? If "yes", then specify page numbers and describe the errors.*
- *Additional comments and suggestions*

s/n of *TORNADO-32L* _____ rev. 1A _____

company _____

person name _____ title _____

address _____

phone _____ fax _____

date _____ signature _____

MicroLAB Systems Ltd
59a Beskudnikovsky blvd, 127486, Mocsow, RUSSIA
phone/fax: +7-(095)-485-6332 E-mail: mlabsys@online.ru