PM8621

NSE-8G

8G Narrowband Switch Element

Data Sheet

Preliminary

Issue 1: May, 2001

Document ID: PMC- PMC-2010850, Issue 1

Legal Information

Copyright

PMC-2010850, (P1)

Disclaimer

Trademarks

Contacting PMC-Sierra

PMC-Sierra
8555 Baxter Place Burnaby, BC
Canada V5A 4V7

Tel: (604) 415-6000
Fax: (604) 415-6200

Document Information: document@pmc-sierra.com
Corporate Information: info@pmc-sierra.com
Technical Support: apps@pmc-sierra.com
Web Site: http://www.pmc-sierra.com

Table of Contents

List of Registers

List of Figures

List of Tables

1 Features

The Narrowband Switch Element 8G (NSE-8G):

- Implements a Scaleable Bandwidth Interconnect (SBI™) DS0 granularity Space switch.
- Implements a SONET/SDH VT1.5/VT2/TU11/TU12 granularity Space switch for the serial 777.6 MHz LVDS TelecomBus.
- With an allied PM8610 SBS or PM8611 SBS-lite device, implements a DS0 granularity Memory-Space-Memory switch.
- Supports 12 STS-12 equivalent serial ports via 777.6 MHz, 8B/10B encoded LVDS links (each port can be either Serial TeleCombus or Serial SBI336S)
- When configured for SBI mode, switches DS0 or N*DS0 for all T1 and E1 tributaries and aggregate columns for switching T1, E1, TVT1.5, TVT2, DS3 and E3 tributaries.
- When configured for the serial 777.6 MHz TelecomBus interface, switches any SONET/SDH virtual tributary (VT) or tributary unit (TU) up to STS-1.
- Supports switching of arbitrary non-standard octet aggregates.
- Supports unicast, multicast, and broadcast for all switching modes.
- Provides 8 Gbit/s (96,768 DS0s, 4.032 T1s/VT1.5s, 3,024 E1s/VT2s, 144 DS3s/E3s) switching.
- Works with SBS devices that support up to four 19.44 MHz SBI or one 77.76 MHz SBI336 bus that communicates with PMC-Sierra's SBI device family. Alternatively, the SBS and SBS-lite devices support up to four 19.44 MHz STS-3 TelecomBuses or one 77.76 MHz STS-12 TelecomBus for connection with PMC-Sierra's SPECTRA family of devices.
- Can be combined in applications with PMC-Sierra's CHESS™ chip set devices (PM5374 TSE and PM5307 TBS).
- Supports a microprocessor interface used to configure/control the NSE and make DS0-granularity switch settings.
- Supports clean error checked 8 Mbit/s full-duplex, in-band communications channels from its attached microprocessor to the attached microprocessors of each of the 12 attached SBS336S devices. This channel is used to initialize and control the SBSs, or other such devices, and to implement call-establishment set-up changes.
- Supports JTAG for all non-LVDS signals.
- Requires dual power supplies at 1.8V and 3.3V.
- Packaged as a 480 ball UBGA.
- In conjunction with the SBS or SBS-lite, supports "1+1" and "1:N" fabric redundancy.

2 Applications

The PM8621 Narrowband Switch Element 8G (NSE-8G) supports a variety of flexible Layer 1/Layer 2 architectures in combination with the following PMC-Sierra devices:

- PM8610 SBS and PM8611 SBS-lite (SBI Serializer and Memory switching stage).
- SBI bus devices (TEMUX™/TEMAP, FREEDM devices, S/UNI®-IMA devices, AAL1gator™ devices and other future devices).
- CHESS™ chip set devices (PM5374 TSE, PM5307 TBS, PM5315 SPECTRA™-2488, and PM7390 S/UNI®-MACH48).

These architectures include:

- T1/E1 SONET ADMs.
- TDM ASAP applications.
- PHY cards with DS0 (and above) level switching.
- PSTN replacement switching cores, as part of any-service-any-port applications, and
- Voice Gateways.

3 References

1. ANSI - T1.105-1995, “Synchronous Optical Network (SONET) – Basic Description including Multiplex Structure, Rates, and Formats”, 1995.

2. Telcordia - SONET Transport Systems: Common Generic Criteria, GR-253-CORE, Issue 2, Revision 2, January 1999.

3. ITU, Recommendation G.707 - "Digital Transmission Systems – Terminal equipments - General", March 1996.

4. IEEE 802.3, “Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications”, Section 36.2, 1998.

5. A.X. Widmer and P.A. Franaszek, “A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code,” IBM Journal of Research and Development, Vol. 27, No 5, September 1983, pp 440-451.

6. U.S. Patent No. 4,486,739, P.A. Franaszek and A.X. Widmer, “Byte Oriented DC Balanced (0,4) 8B/10B Partitioned Block Transmission Code,” December 4, 1984.

7. IEEE Std 1596.3-1996, “IEEE Standard for Low-Voltage Differential Signals (LVDS) for Scalable Coherent Interface (SCI)”, Approved March 21, 1996

8. L.R. Ford, D.R. Fulkerson, “Flows in Networks", Maximum Cardinality Matchings in Bipartite Graphs

4 Application Examples

Figure 1 illustrates an OC-48 SONET Ring Add/Drop Multiplexer. The PM5363 TUPP-622 devices align all paths to transport frames in preparation for VT1.5/VT2 granularity switching. The PM8610 SBI336 Bus Serializer (SBS™) an PM8621 Narrowband Switching Element 8G (NSE-8G™) devices support VT1.5/VT2 and above switching. The Add and Drop buses are provided by the SBSs that are not in the SONET Ring path. In this case, they connect to T1 and E1 mapper ports.

Figure 1 An OC-48 T1/E1 ADM (Individually Drop/Add any T1/E1 in STS-48)

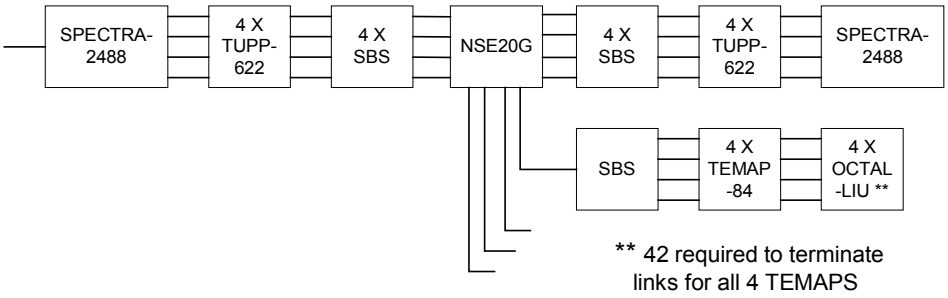


Figure 2 illustrates another OC-48 SONET Ring ADM. In this application, the network of three PM5310 TelecomBus Serializers (TBSs) from PMC-Sierra's CHESS™ chip set add, drop, and groom traffic at STS-1 granularities. The four TUPP-622 devices align any dropped STS-1s (paths to transport frames). The virtual tributary (VT) or tributary unit (TU) switching solution is provided by the SBS-NSE-8G-SBS network below the TUPP-622s. Four SBSs support up to an STS-48 amount of add/drop traffic.

Figure 2 An OC-48 T1/E1 ADM (Drop/Add up to STS-48 at STS-1 Granularity)

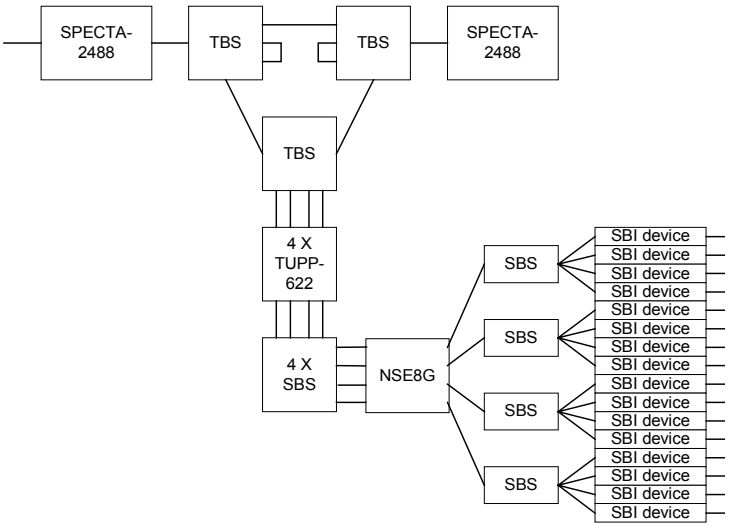
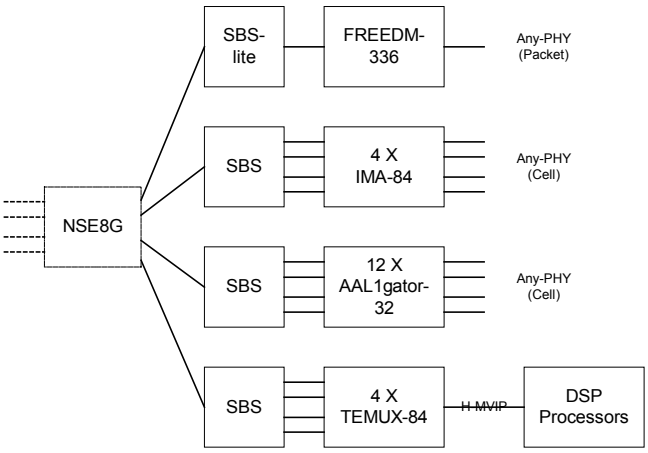


Figure 3 illustrates the organization of the access line size card(s) from a SONET Any Service Any Port (ASAP) product. All traffic from the NSE-8G to the SBI link layer devices is path-aligned. See Figure 4 for a description of the PHY line cards compatible with the system in Figure 3.

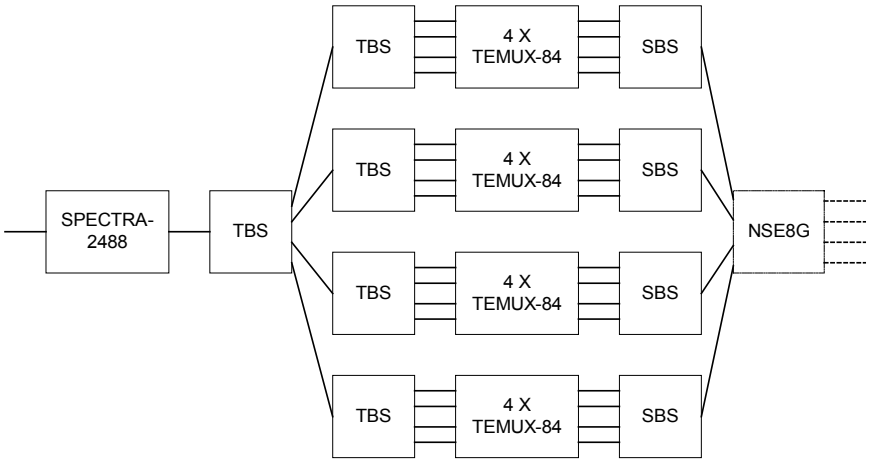
Figure 3 Any-Service-Any-Port TDM Access Solution



T1/E1/DS0/N*DS0 Layer 2 Processing

Figure 4 shows the organization of a SONET PHY card compatible with Figure 3. As shown, both Figure 3 and Figure 4 have NSE-8Gs, but only one instance of this device is required to connect all the SBSs. A likely packaging of this combined system would place the NSE-8G (and a standby NSE-8G) on separate fabric cards. In Figure 4, PM8315 TEMUXs align paths to transport frames. Note: Figure 3 assumes this alignment.

Figure 4 Any-Service-Any-Port DS0-Granularity PHY Card

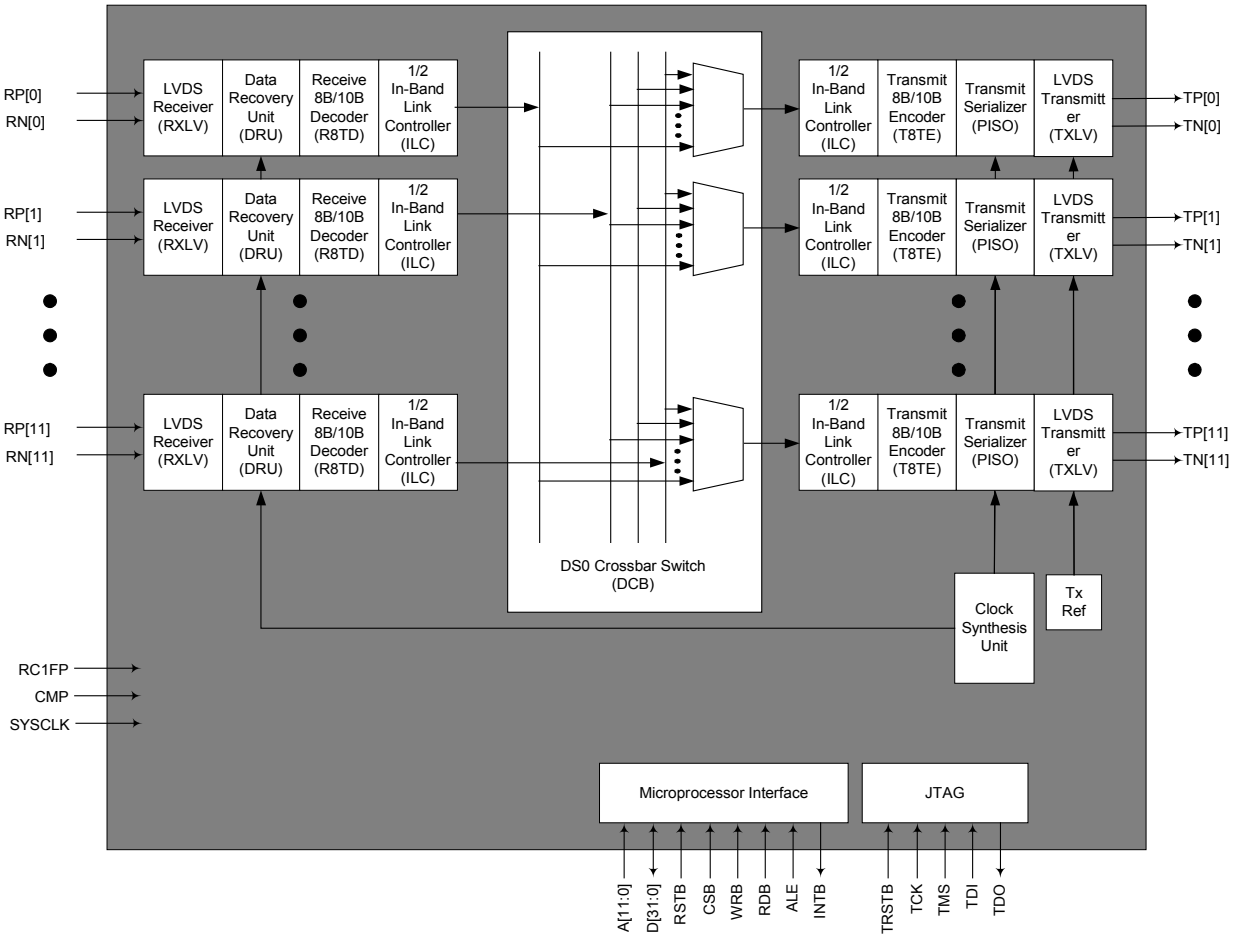


SONET/T1/E1 Termination - VT/TU/DS0 Switching

5 Block Diagram

The NSE-8G is organized as a DS0 granularity space switch. Alternatively, the NSE-8G is organized as a self aligning (with respect to STS-12 boundaries in TelecomBus mode) VT1.5/VT2 granularity space switch.

Figure 5 NSE- 8G Block Diagram Showing Functional Blocks



The R8TD, in combination with the RXLV and DRU receive, decode and align incoming SBI336/STS-12-equivalent LVDS links. Outputs are provided to the primary switching flow and to the in-band signaling channel. These provide all analog and digital functions to terminate a full-duplex 777.6 MHz serial SBI336S or 777.6 MHz serial TelecomBus on LVDS.

A 12 X 12 DS0 Crossbar Switch(DCB) stage switches data and control signals between the 12 ports. The switching instructions are stored in two pages of ram configured as offline and online allowing the user to modify the offline page.

The T8TE, in combination with the PISO and TXLV perform 8B/10B coding and emits the LVDS bit streams. These provide all analog and digital functions to launch a full-duplex 777.6 MHz serial SBI336S bus or 777.6 MHz serial TelecomBus on LVDS.

The microprocessor bus interface and in-band signaling units (ILC) provide a clean (error checked) channel between the NSE-8G and SBSs. This can be used to send messages between the NSE-8G microprocessor-and the SBS microprocessors in a user defined format.

6 Description

The PM8621 NSE-8G is a monolithic CMOS integrated circuit packaged in a 480 ball UBGA that performs DS0 and above granularity space switching on 12 SBI336 streams carried as serial SBI336S in 8B/10B coding over LVDS at 777.6 Mbit/s. The NSE-8G also performs VT1.5/VT2 and above granularity switching on 12 STS-12/STM-4 SONET/SDH streams, carried as Serial TelecomBus signals in 8B/10B coding over LVDS at 777.6 Mbit/s.

The NSE-8G is typically used with up to 12 PM8610 SBS or PM8611 SBS-lite devices to provide Memory-Space-Memory switching systems. As each SBS supports either four SBI buses at 19.44 MHz or one SBI336 bus at 77.76 MHz, the overall system supports any mixture of SBI and SBI336 byte serial buses, ranging from 48 19.44 MHz SBI buses to 12 SBI336 77.76 MHz buses that do not exceed an aggregate bandwidth of STS-144, or about 7.5 Gbit/s. In TelecomBus mode, the SBS devices support the same range of flexibility for 48 19.44 MHz and 12 77.76 MHz TelecomBuses at VT1.5/VT2 granularity

Central to the NSE-8G is a 12 x 12 cross bar switch. Every clock cycle, the cross bar switches a byte of data with control signals from each input port to an output port. The byte of data may be a DS0 channel from a T1/E1 or may be one byte of a column comprising a T1, E1, DS3, E3, VT1.5, VT2 or STS-1.

In order for switching to take place, all input and output streams must be synchronized. This is done via the RC1FP input signal. When switching T1s, E1s, VTs and other higher order units, only SBI336 multiframe alignment is required. The same applies for TelecomBus mode where only frame alignment is required.

An in-band control link over the serial LVDS interface allows the NSE-8G to communicate with the microprocessors attached to the SBS, SBS-lite or other serial SBI336S devices. The effective bandwidth of each inband link to each device is 8 Mbit/s. The inband link provides error detection on 32 byte user messages and some near realtime control signals between devices. Using the near realtime control signals the NSE-8G is able to synchronize page switching, indicate switchover between working or protected links and exchange three user defined signals (software) and eight Auxilliary signals (software). The User and Auxilliary signals can be used to indicates things like interrupts or can be used for handshaking between the end point microprocessors. The message format is left to the user of the devices. The only constraint is that each message is a maximum of 32 bytes long.

7 Pin Diagram

The NSE-8G is packaged in a 35 mm x 35 mm 480 ball UBGA.

Figure 6 NSE-8G UBGA-480 Ball Diagram (Bottom-View)

Upper Left

	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
A	VSS	VSS	VSS	VSS	VDDO	VSS	NC	VSS	NC	VSS	Reserved	VSS	Reserved	VSS	Reserved	VSS	VDDI
B	VSS	AVDH	VDDO	VDDO	VDDO	VDDI	NC	NC	NC	VDDI	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	RSTB
C	VSS	AVDH	AVDH	VDDO	VDDI	Reserved	NC	NC	VDDI	NC	Reserved	VDDI	Reserved	Reserved	Reserved	Reserved	VDDI
D	VSS	AVDH	AVDH	AVDH	VDDO	Reserved	VDDI	VDDO	NC	NC	Reserved	VDDO	Reserved	Reserved	Reserved	VDDO	VDDI
E	RESK1	RES1	NC	NC													
F	VSS	NC	NC	AVDL1													
G	NC	NC	NC	NC													
H	VSS	NC	NC	AVDH													
J	NC	NC	NC	NC													
K	VSS	NC	NC	VDDI													
L	NC	NC	NC	NC													
M	VSS	NC	NC	AVDH													
N	VDDI	AVDL2	NC	NC													
P	VSS	NC	NC	VDDI													
R	NC	NC	NC	NC													
T	NC	NC	AVDL4	AVDL3													
U	NC	NC	AVDL5	CSU_AVDH													

Upper Right

17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
NC	SYSCLK	NC	VSS	NC	VSS	NC	VSS	Reserved	VSS	Reserved	VSS	NC	VSS	VSS	VSS	VSS	A
NC	NC	NC	TCK	TMS	NC	VDDI	Reserved	Reserved	Reserved	Reserved	Reserved	NC	VDDO	VDDO	VDDO	VSS	B
Reserved	NC	VDDI	NC	VDDI	TDI	TDO	NC	Reserved	Reserved	Reserved	Reserved	VDDI	VDDO	VDDO	AVDH	VSS	C
NC	RC1FP	VDDI	TRSTB	VDDI	VDDO	VDDI	CMP	Reserved	VDDO	Reserved	Reserved	NC	VDDO	AVDH	AVDH	VSS	D
													AVDH	ATB0[1]	AVDH	AVDH	E
													ATB1[1]	TN[1]	TP[1]	VSS	F
													TN[3]	TP[3]	TN[2]	TP[2]	G
													AVDH	VDDI	NC	VSS	H
													RP[1]	RN[1]	TN[4]	TP[4]	J
													VDDI	RP[2]	RN[2]	VSS	K
													VDDI	AVDL14	RP[3]	RN[3]	L
													AVDH	RP[4]	RN[4]	VSS	M
													TN[6]	TP[6]	TN[5]	TP[5]	N
													VDDI	TN[7]	TP[7]	VSS	P
													RP[5]	RN[5]	TN[8]	TP[8]	R
													AVDH	VDDI	AVDL13	VSS	T
													RP[7]	RN[7]	RP[6]	RN[6]	U

Lower Left

V	NC	NC	NC	NC													
W	VSS	AVDL6	VDDI	AVDH													
Y	NC	NC	NC	NC													
AA	VSS	NC	NC	VDDI													
AB	NC	NC	NC	NC													
AC	VSS	NC	NC	AVDH													
AD	NC	NC	AVDL7	VDDI													
AE	VSS	NC	NC	VDDI													
AF	NC	NC	NC	NC													
AG	VSS	NC	VDDI	AVDH													
AH	NC	NC	NC	NC													
AJ	VSS	NC	NC	ATB1[2]													
AK	AVDH	AVDH	ATB0[2]	AVDH													
AL	VSS	AVDH	AVDH	VDDO	ALE	NC	VDDI	VDDO	A[6]	A[2]	VDDI	VDDO	D[27]	VDDI	NC	NC	VDDI
AM	VSS	AVDH	VDDO	VDDO	CSB	RDB	VDDI	A[9]	A[5]	A[3]	D[31]	D[29]	VDDI	D[25]	VDDI	D[21]	D[20]
AN	VSS	VDDO	VDDO	VDDO	INTB	WRB	NC	A[10]	A[7]	A[4]	A[0]	D[30]	D[28]	D[26]	NC	D[22]	D[19]
AP	VSS	VSS	VSS	VSS	NC	VSS	A[11]	VSS	A[8]	VSS	A[1]	VSS	NC	VSS	D[24]	D[23]	D[18]
	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18

Lower Right

													CSU_AVDH	AVDL12	RP[8]	RN[8]	V
													AVDL10	AVDL11	TN[9]	TP[9]	W
													TN[10]	TP[10]	TN[11]	TP[11]	Y
													VDDI	TN[12]	TP[12]	VSS	AA
													RP[9]	RN[9]	AVDL9	VDDI	AB
													AVDH	RP[10]	RN[10]	VSS	AC
													RP[11]	RN[11]	RP[12]	RN[12]	AD
													VDDI	NC	NC	VSS	AE
													NC	NC	NC	NC	AF
													AVDH	NC	NC	VSS	AG
													NC	NC	NC	NC	AH
													AVDL8	NC	NC	VSS	AJ
													NC	NC	RES2	RESK2	AK
D[17]	VDDO	D[13]	D[11]	D[8]	VDDO	D[5]	D[3]	D[0]	VDDO	NC	NC	VDDO	AVDH	AVDH	AVDH	VSS	AL
VDDI	D[15]	VDDI	D[10]	D[9]	D[7]	NC	D[2]	D[1]	NC	NC	NC	NC	VDDO	AVDH	AVDH	VSS	AM
D[16]	D[14]	D[12]	NC	VDDI	D[6]	D[4]	VDDI	NC	NC	NC	NC	VDDO	VDDO	VDDO	AVDH	VSS	AN
NC	VSS	VDDI	VSS	VDDI	VSS	NC	VSS	NC	VSS	NC	VSS	VDDO	VSS	VSS	VSS	VSS	AP
17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	

8 Pin Description

8.1 Pin Description Table

Pad Name	Type	Pin No.	Function
LVDS Ports (128 Balls)			
RP[1]	Analog LVDS Input	J4	**Receive Serial Data.** The differential receive serial data links (RP[11:0]/RN[11:0]) carry the receive SBI336S or SONET/SDH STS-12 frame data from upstream sources in bit serial format. Each differential pair RP[*X*]/RN[*X*] carries a constituent SBI336 or STS-12 stream. Data on RP[*X*]/RN[*X*] is encoded in an 8B/10B format extended from IEEE Std. 802.3. The 8B/10B character bit 'a' is transmitted first and the bit 'j' is transmitted last. All RP[*X*]/RN[*X*] differential pairs must be frequency locked and phase aligned (within a certain tolerance) to each other. RP[11:0]/RN[11:0] are nominally 777.6 Mbit/s data streams. Any unused or N/C, but available inputs should be tied low using a 10 k resistor.
RN[1]		J3	
RP[2]		K3	
RN[2]		K2	
RP[3]		L2	
RN[3]		L1	
RP[4]		M3	
RN[4]		M2	
RP[5]		R4	
RN[5]		R3	
RP[6]		U2	
RN[6]		U1	
RP[7]		U4	
RN[7]		U3	
RP[8]		V2	
RN[8]		V1	
RP[9]		AB4	
RN[9]		AB3	
RP[10]		AC3	
RN[10]		AC2	
RP[11]		AD4	
RN[11]		AD3	
RP[12]		AD2	
RN[12]		AD1	

Pad Name	Type	Pin No.	Function
TP[1]	Analog LVDS Output	F2	**Transmit Serial Data.** The differential transmit working serial data links (TP[11:0]/TN[11:0]) carry the transmit SBI336S or SONET/SDH STS-12 frame data to a downstream sinks in bit serial format. Each differential pair carries a constituent STS-12 stream. Data on TP[*X*]/TN[*X*] is encoded in an 8B/10B format extended from IEEE Std. 802.3. The 8B/10B character bit 'a' is transmitted first and the bit 'j' is transmitted last. All TP[*X*]/TN[*X*] differential pairs are frequency locked and phase aligned (within a certain tolerance) to each other. TP[11:0]/TN[11:0] are nominally 777.6 Mbit/s data streams.
TN[1]		F3	
TP[2]		G1	
TN[2]		G2	
TP[3]		G3	
TN[3]		G4	
TP[4]		J1	
TN[4]		J2	
TP[5]		N1	
TN[5]		N2	
TP[6]		N3	
TN[6]		N4	
TP[7]		P2	
TN[7]		P3	
TP[8]		R1	
TN[8]		R2	
TP[9]		W1	
TN[9]		W2	
TP[10]		Y3	
TN[10]		Y4	
TP[11]		Y1	
TN[11]		Y2	
TP[12]		AA2	
TN[12]		AA3	
NSE-8G Control and Clocking (5 Balls)			
SYSCLK	Input	A16	**System Clock.** The system clock signal (SYSCLK) is the master clock for the NSE-8G device. SYSCLK must be a 77.76 MHz clock, with a nominal 50% duty cycle. CMP and RC1FP are sampled on the rising edge of SYSCLK.

Pad Name	Type	Pin No.	Function
RC1FP	Input	D16	**Receive Serial Interface Frame Pulse.** The receive serial interface frame pulse signal (RC1FP) provides system timing for the receive serial interface. RC1FP is supplied in common to all devices in a system containing one or more NSE-20G devices. In TelecomBus mode, RC1FP is set high once every four frames, in SBI mode without any DS0 switching, or when switching DS0s (**WITHOUT** CAS) RC1FP is also set high once every four frames, or multiple thereof. When in SBI mode switching DS0s **WITH** CAS RC1FP indicates signaling multiframe alignment by pulsing once every 48 frames or multiples thereof. A software configurable delay from RC1FP is used to indicate that the C1 multiframe boundary 8B/10B characters have been delivered on all the receive serial data links (RP[32:1]/RN[32:1]) and are ready for processing by the time-space-time switching elements. RC1FP is sampled on the rising edge of SYSCLK.
Reserved	Output	C17	Reserved pin, must be left floating
CMP	Input	D10	**Connection Memory Page**. The connection memory page select signal (CMP) controls the selection of the connection memory page in the NSE. When CMP is set high, connection memory page 1 is selected. When CMP is set low, connection memory page 0 is selected. Changes to the connection memory page selection are synchronized to the boundary of the next C1FP frame or multiframe depending on the mode: <u>4-Frame SBI/SBI336 mode:</u> CMP is sampled at the C1 byte position of the incoming bus on the first frame of the four-frame multiframe. Changes to the connection memory page selection are synchronized to the frame boundary (A1 byte position) of the next four-frame multiframe. <u>48-Frame SBI/SBI336 mode:</u> CMP is sampled at the C1 byte position of the incoming bus on the first frame of the 48-frame multiframe. Changes to the connection memory page selection are synchronized to the frame boundary (A1 byte position) of the next 48-frame multiframe. <u>TelecomBus mode:</u> CMP is sampled at the C1 byte position of every frame on the incoming bus. Changes to the connection memory pate selection are synchronized to the frame boundary (A1 byte position) of the next frame. CMP is sampled on the rising edge of SYSCLK at the RC1FP frame position.
RSTB	Input	B18	**Reset Enable Bar.** The active low reset signal (RSTB) provides an asynchronous reset for the NSE. RSTB is a Schmitt triggered input with an integral pull-up resistor

Pad Name	Type	Pin No.	Function
Microprocessor Interface (49 Balls)			
CSB	Input	AM30	**Chip Select Bar.** The active low chip select signal (CSB) controls microprocessor access to registers in the NSE-8G device. CSB is set low during NSE-8G Microprocessor Interface Port register accesses. CSB is set high to disable microprocessor accesses. If CSB is not required (i.e. register accesses controlled using RDB and WRB signals only), CSB should be connected to an inverted version of the RSTB input.
RDB	Input	AM29	**Read Enable Bar.** The active low read enable bar signal (RDB) controls microprocessor read accesses to registers in the NSE-8G device. RDB is set low and CSB is also set low during NSE-8G Microprocessor Interface Port register read accesses. The NSE-8G drives the D[31:0] bus with the contents of the addressed register while RDB and CSB are low.
WRB	Input	AN29	**Write Enable Bar.** The active low write enable bar signal (WRB) controls microprocessor write accesses to registers in the NSE-8G device. WRB is set low and CSB is also set low during NSE-8G Microprocessor Interface Port register write accesses. The contents of D[31:0] are clocked into the addressed register on the rising edge of WRB while CSB is low.

Pad Name	Type	Pin No.	Function
D[31] D[30] D[29] D[28] D[27] D[26] D[25] D[24] D[23] D[22] D[21] D[20] D[19] D[18] D[17] D[16] D[15] D[14] D[13] D[12] D[11] D[10] D[9] D[8] D[7] D[6] D[5] D[4] D[3] D[2] D[1] D[0]	I/O	AM24 AN23 AM23 AN22 AL22 AN21 AM21 AP20 AP19 AN19 AM19 AM18 AN18 AP18 AL17 AN17 AM16 AN16 AL15 AN15 AL14 AM14 AM13 AL13 AM12 AN12 AL11 AN11 AL10 AM10 AM9 AL9	**Microprocessor Data Bus.** The bi-directional data bus, D[31:0] is used during NSE-8G Microprocessor Interface Port register reads and write accesses. D[31] is the most significant bit of the data words and D[0] is the least significant bit.
A[11 A[10] A[9] A[8] A[7] A[6] A[5] A[4] A[3] A[2] A[1] A{0]	Input	AP28 AN27 AM27 AP26 AN26 AL26 AM26 AN25 AM25 AL25 AP24 AN24	**Microprocessor Address Bus.** The microprocessor address bus (A[11:0]) selects specific Microprocessor Interface Port registers during NSE-8G register accesses.

Pad Name	Type	Pin No.	Function
ALE	Input	AL30	**Address Latch Enable.** The address latch enable signal (ALE) is active high and latches the address bus (A[11:0]) when it is set low. The internal address latches are transparent when ALE is set high. ALE allows the NSE-8G to interface to a multiplexed address/data bus. ALE has an integral pull up resistor.
INTB	Open Drain Output	AN30	**Interrupt Request Bar**. The active low interrupt enable signal (INTB) output goes low when an NSE-8G interrupt source is active and that source is unmasked. INTB returns high when the interrupt is acknowledged via an appropriate register access. INTB is an open drain output.
JTAG Port (5 Balls)			
TCK	Input	B14	**Test Clock.** The JTAG test clock signal (TCK) provides timing for test operations that are carried out using the IEEE P1149.1 test access port.
TMS	Input	B13	**Test Mode Select.** The JTAG test mode select signal (TMS) controls the test operations that are carried out using the IEEE P1149.1 test access port. TMS is sampled on the rising edge of TCK. TMS has an integral pull-up resistor.
TDI	Input	C12	**Test Data Input.** The JTAG test data input signal (TDI) carries test data into the NSE-8G via the IEEE P1149.1 test access port. TDI is sampled on the rising edge of TCK. TDI has an integral pull-up resistor.
TDO	Tri-state	C11	**Test Data Output.** TheJTAG test data output signal (TDO) carries test data out of the NSE-8G via the IEEE P1149.1 test access port. TDO is updated on the falling edge of TCK. TDO is a tri-state output which is inactive except when scanning of data is in progress.
TRSTB	Input	D14	**Test Reset Bar.** The active low JTAG test reset signal (TRSTB) provides an asynchronous NSE-8G test access port reset via the IEEE P1149.1 test access port. TRSTB is a Schmitt triggered input with an integral pull-up resistor. Note that when TRSTB is not being used, it must be connected to the RSTB input.

Pad Name	Type	Pin No.	Function
Reserved			
Reserved	Input	C29 D29 B6 C6 D6 A7 B7 D7 C7 B8 C8 A9 B9 D9 C9 B10 C19 B19 C20 D20 B20 A20 D21 C21 B21 C22 D22 B22 A22 B23 C24 D24 B24 A24	**Reserved.** Must be left floating [internally pulled up].
External Resistors (4 Balls)			
RES[2] RES[1]	Analog Input	AK2 E33	**Reference Resistor Connection.** An off-chip 3.16kΩ ±1% resistor is connected between these the positive resistor reference pin RES and a Kelvin ground contact RESK. An on-chip negative feedback path will force the 1.20 V VREF onto RES, therefore forcing 252 µA of current to flow through the resistor.

Pad Name	Type	Pin No.	Function
RESK[2] RESK[1]	Analog Input	AK1 E34	**Reference Resistor Connection.** An off-chip 3.16 kΩ ±1% resistor is connected between these the positive resistor reference pin RES and a Kelvin ground contact RESK. An on-chip negative feedback path will force the 1.20 V VREF Voltage onto RES, therefore forcing 252 µA of current to flow through the resistor.
Analog Test Bus (4 Balls)			
ATB0[2] ATB0[1]	Analog	AK32 E3	Analog test bus for PMC validation and testing. This pin must be connected to GND.
ATB1[2] ATB1[1]	Analog	AJ31 F4	Analog test bus for PMC validation and testing. This pin must be connected to GND.
Digital Core Power (45 Balls)			
VDDI[44:0]	Power	AA4 AB1 AE4 AN10 AN13 AP13 AP15 AM15 AM17 AL18 AM20 AL21 AM22 AL24 AM28 AL28 AG32 AE31 AD31 AA31 W32 P31 N34 K31 T3 P4	The digital core power pins (VDDI[44:0]) should be connected to a well-decoupled +1.8 V DC supply.

Pad Name	Type	Pin No.	Function
VDDI[44:0]	Power	L4K4 H3 C5 B11 D11 D13C13 C15 D15 C18 D18 A18 C23 B25 C26 D28 B29 C30	
Digital I/O Power (34 Balls)			
VDDO[33:0]	Power	AL5 AM4 AN3 AN4 AN5 AP5 AL8 AL12 AL16 AL23 AL27 AL31 AM31 AM32 AN31 AN32 AN33 A30	The digital I/O power pins (VDDO[33:0]) should be connected to a well-decoupled +3.3 V DC supply.

Pad Name	Type	Pin No.	Function
VDDO[33:0]	Power	B30 B31 B32 C31 D30 D27 D23 D19 D12 D8 B2 B3 B4 C3 C4 D4	
Digital Ground (72 Balls)			
VSS [71:0]	Ground	A1 A2 A3 A4 A6 A8 A10 A12 A14 A19 A21 A23 A25 A27 A29 A31 A32 A33 A34	The digital ground pins (VSS [71:0]) should be connected to GND.

Pad Name	Type	Pin No.	Function
VSS [71:0]	Ground	AP1	
		AP2	
		AP3	
		AP4	
		AP6	
		AP8	
		AP10	
		AP12	
		AP14	
		AP16	
		AP21	
		AP23	
		AP25	
		AP27	
		AP29	
		AP31	
		AP32	
		AP33	
		AP34	
		B1	
		C1	
		D1	
		F1	
		H1	
		K1	
		M1	
		P1	
		T1	
		AA1	
		AC1	
		AE1	
		AG1	
		AJ1	
		AL1	
		AM1	
		AN1	
		B34	
		C34	
		D34	
		F34	
		H34	

Pad Name	Type	Pin No.	Function
VSS [71:0]	Ground	K34 M34 P34 W34 AA34 AC34 AE34 AG34 AJ34 AL34 AM34 AN34	
Analog Power (8 Balls)			
AVDL[7:0]	Power	F31 N33 W33 AD32 AJ4 AB2 T2 L3	The analog power pins (AVDL[7:0]) should be connected to a well-decoupled +1.8 V DC supply. These pins supply the RXLVs.
Clock Synthesis 1.8 V Power (6 Balls)			
CSU_AVDL[5:0]	Power	T31 T32 U32 W4 W3 V3	The clock synthesis pins (CSU_AVDL[5:0]) should be connected to a well-decoupled +1.8 V DC supply. These pins supply the CSUs.
Clock Synthesis Power (2 Balls)			
CSU_AVDH[0:1]	Power	U31 V4	These two pins should be connected to a well-decoupled +3.3 V DC supply.

Pad Name	Type	Pin No.	Function
Analog I/O Power (34 Balls)			
AVDH[33:0]	Power	H4 M4 T4 AC4 AG4 AL2 AL3 AL4 AM2 AM3 AN2 C2 D2 D3 E1 E2 E4 B33 C32 C33 D31 D32 D33 AG31 AC31 W31 M31 H31 AK31 AK33 AK34 AL32 AL33 AM33	The analog I/O power pins (AVDH[33:0]) should be connected to a well-decoupled +3.3 V DC supply.

Pad Name	Type	Pin No.	Function
No Connect (50 Balls)			
NC[49:0]		AG33 AP30 AL29 AN28 AP22 AN20 AL20 AL19 AP17 AN14 AM11 AP11 AN9 AP9 AM8 AN8 AM7 AL7 AN7 AP7 AL6 AM6 AN6 AM5 H2 A5 B5 D5 C10 A11 B12 A13 C14 A15 B15 B16 C16	The No Connect pins (NC[49:0]) should be left floating.

Pad Name	Type	Pin No.	Function
NC[49:0]		D17 B17 A17 D25 C25 D26 B26 A26 C27 B27 C28 B28 A28	
TOTAL		480	

Notes

1. All NSE-8G inputs and bi-directional balls except the LVDS links present minimum capacitive loading and operate at TTL logic levels.
2. Inputs RSTB, ALE, TMS, TDI and TRSTB have internal pull-up resistors.
3. All outputs have a minimum 8 mA drive capability – this includes TDO, INTB and D[31:0]).
4. The VDDI and AVDL power pins are not internally connected to each other. Failure to connect these pins externally may cause malfunction or damage to the device.
5. The AVDH, CSU_AVDH and VDDO power pins are not internally connected to each other. Failure to connect these pins externally may cause malfunction or damage to the device.
6. The VDDI, VDDO, AVDH, CSU_AVDH and AVDL power pins all share the common ground VSS.
7. To prevent damage to the device and to ensure proper operation, power must be applied simultaneously to all 3.3 V power pins followed by power to all the 1.8 V power pins followed by input pins driven by signals.
8. To prevent damage to the device, power must first be removed from input pins followed by the removal of power from all the 1.8 V power supply pins followed by the simultaneous removal of power from all the 3.3 V power pins.
9. The 3.3 V supplies should never be less than the 1.8 V supplies at any time during power-up and power-down.

8.2 Analog Power Filtering Recommendations

To achieve best performance of the LVDS links, an analog filter network should be installed between the power balls and the supply.

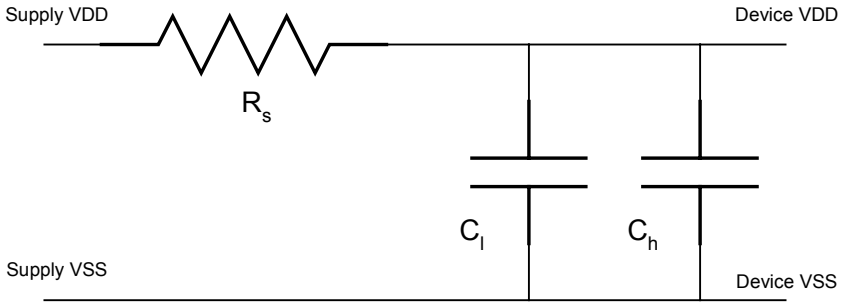
Table 1 Analog Power Filters

	Rs	Cl	Ch	Notes
CSU AVDH (2 balls)	3.3 ohm	100 nF	10 nF	One Filter network per VDD ball.
CSU AVDL (6 balls)	0.47 ohm	4.7 µF	10 nF	One Filter network per VDD ball.
AVDH (34 balls)	3.3 ohm	1.0 µF	10 nF	Two VDD balls per filter network[1].
AVDL (8 balls)	0 ohm	100 nF	10 nF	One Filter network per VDD ball.

Note

1. Two power-gnd pairs can use the same filter.

Figure 7 Analog Power Filter Circuit



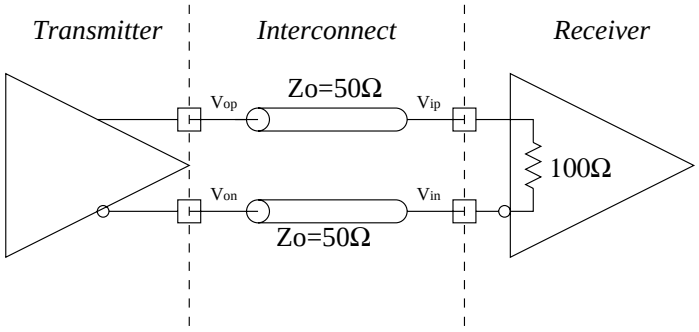
9 Functional Description

9.1 LVDS Overview

The LVDS family of cells allow the implementation of 777.6 Mbit/s LVDS links. A reference clock of 77.76 MHz is required.

A generic LVDS link according to IEEE 1596.3-1996 is illustrated in Figure 7 below. The transmitter drives a differential signal through a pair of 50 Ω characteristic interconnects, such as board traces, backplane traces, or short lengths of cable. The receiver presents a 100 Ω differential termination impedance to terminate the lines. Included in the standard is sufficient common-mode range for the receiver to accommodate as much as 925 mV of common-mode ground difference.

Figure 8 Generic LVDS Link Block Diagram



Complete SERDES transceiver functionality is provided. Ten-bit parallel data is sampled by the line rate divided-by-10 clock (77.76 MHz SYSCLK) and then serialized at the line rate on the LVDS output pins by a 777.6 MHz clock synthesized from SYSCLK. Serial line rate LVDS data is sampled and de-serialized to 10-bit parallel data. Parallel output transfers are synchronized to a gated line rate divided-by-10 clock. The 10-bit data is passed to an 8B/10B decoding block. The gating duty cycle is adjusted such that the throughput of the parallel interface equals the receive input data rate (Line Rate +/- 100 ppm). It is expected that the clock source of the transmitter is the same as the clock source of the receiver to ensure the data throughput at both ends of the link are identical.

Data is guaranteed to contain sufficient transition density to allow reliable operation of the data recovery units by 8B/10B block coding and decoding provided by the T8TE and R8TD blocks.

At the system level, reliable operation will be obtained if proper signal integrity is maintained through the signal path and the receiver requirements are respected. Namely, a worst case eye opening of 0.7UI and 100 mV differential amplitude is needed. These conditions should be achievable with a system architecture consisting of board traces, two sets of backplane connectors, up to 1m of backplane interconnect. This assumes proper design of 100 Ω differential lines and minimization of discontinuities in the signal path. Due to power constraints, the output differential amplitude is approximately 350 mV

The LVDS system is comprised of the LVDS Receiver (RXLV), LVDS Transmitter (TXLV), Transmitter reference (TXREF), data recovery unit (DRU), parallel to serial converter (PISO and Clock Synthesis Unit (CSU).

9.1.1 LVDS Receiver (RXLV)

The RXLV block is a 777.6 Mbit/s Low Voltage Differential Signaling (LVDS) Receiver according to the IEEE 1596.3-1996 LVDS Specification.

The RXLV block is the receiver in Figure 7, accepting 777.6 Mbit/s LVDS signals from the transmitter, over RP[X]/RN[X] pins, amplifying them and converting them to digital signals, then passing them to a data recovery unit (DRU). Holding to the IEEE 1596.3-1996 specification, the RXLV has a differential input sensitivity better than 100 mV, and includes at least 25 mV of hysteresis. There are 12 RXLV blocks in the NSE.

9.1.2 LVDS Transmitter (TXLV)

The TXLV block is a 777.6 Mbit/s Low Voltage Differential Signaling (LVDS) Transmitter according to the IEEE 1596.3-1996 LVDS Specification.

The TXLV accepts 777.6 Mbit/s differential data from a “parallel-in, serial-out” (PISO) circuit and then transmits the data off-chip as a LVDS on TP[X]/TN[X] pins.

The TXLV uses a reference current and voltage from the TXREF block to control the output differential voltage amplitude and the output common-mode voltage.

There are 12 instances of the TXLV block in the NSE-8G.

9.1.3 LVDS Transmit Reference (TXREF)

The TXREF provides an on-chip bandgap voltage reference (1.20 V ±5%) and a precision current to the TXLV (777.6 Mbit/s LVDS Transmitter) block’s. The reference Voltage is used to control the common-mode level of the TXLV output, while the reference current is used to control the output amplitude.

The precision currents are generated by forcing the reference Voltage across an external, off-chip 3.16 kΩ(±1%) resistor. The resulting current is then mirrored through several individual reference current outputs, so each TXLV receives its own reference current.

There is one instance of the TXREF in the NSE-8G.

9.1.4 Data Recovery Unit (DRU)

The DRU is a fully integrated data recovery and serial to parallel converter that can be used for 777.6 Mbit/s NRZ data. 8B/10B block code is used to guarantee transition density for optimal performance.

The DRU recovers data and outputs a ten-bit word synchronized with a line rate divided by ten, gated clock to allow frequency deviations between the data source and the local oscillator. The output clock is not a recovered clock. The DRU accumulates 10 data bits and outputs them on the next clock edge. If 10-bits are not available for transfer at a given clock cycle, the output clock is gated.

The DRU provides moderate high frequency jitter tolerance suitable for inter-chip serial link applications. It can support frequency deviations up to ±100 ppm.

There are 12 instances of the DRU on the NSE-8G.

9.1.5 Parallel to Serial Converter (PISO)

The PISO is a parallel-to-serial converter designed for high-speed transmit operation, supporting up to 777.6 Mbit/s.

There are 12 instances of the PISO on the NSE-8G.

9.1.6 Clock Synthesis Unit (CSU)

The CSU is a fully integrated clock synthesis unit. It generates low jitter multi-phase differential clocks at 777.6 MHz for the use by the transmitter.

There is one instance of the CSU on the NSE-8G.

9.2 Receive 8B/10B Frame Aligner (R8TD)

The Receive 8B/10B serial SBI336S Bus frame aligner, R8TD, frames to the receive stream to find 8B/10B character boundaries. It also contains a FIFO to bridge between the timing domain of the receive LVDS links and the system clock timing domain. The R8TD blocks perform framing and elastic store functions on data retrieved from the receive LVDS links, RP[x]/RN[x].

9.2.1 FIFO Buffer

The FIFO buffer sub-block provides isolation between the timing domains of the associated receive LVDS link and that of the system clock, SYSCLK. Data with arbitrary alignment to the 8B/10B characters, are written into a 10-bit by 24-word deep FIFO at the link clock rate. Data is read from the FIFO at every SYSCLK cycle.

9.3 Transmit 8B/10B Encoder (T8TE)

The Transmit 8B/10B Encoder blocks, T8TE, construct an 8B/10B character stream from an incoming translated SBI336 or TelecomBus carrying an STS-12/STM-4 equivalent channelized stream. The T8TE block corrects the running disparity of an 8B/10B character stream and buffers data in a FIFO before transmission to the transmit serializer block. A total of 32 T8TE blocks are instantiated in the NSE-20G device.

In SBI mode, these blocks encode the SBI336S stream as shown in Table 2. When configured for Synchronous mode for DS0 switching, the 8B/10B encoder transmits CAS signaling multiframe alignment across the SBI336S interface by generating a C1FP character every 48 frame times. When not configured for DS0 switching, the C1FP character is sent every four frames.

9.3.1 SBI336S 8B/10B Character Encoding

Table 2 shows the mapping of SBI336S bus control bytes and signals into 8B/10B control characters. The linkrate octet in location V4, V1 and V2, the in-band programming channel, the V3 octet when it contains data are all carried as data. Justification requests for master timing are carried in the V5 character so there are three V5 characters used, nominal, negative timing adjustment request, positive timing adjustment request.

Table 2 SBI336S Character Encoding

Code Group Name	Curr. RD- abcdei fghj	Curr. RD+ abcdei fghj	Encoded Signals Description
Common to All Link Types			
K28.5	001111 1010	110000 0101	IC1FP='b1 C1FP frame and multiframe alignment
K23.7-	111010 1000	-	Overhead Bytes (columns 1-60 or 1-72 except for C1 and in-band programming channel), V3 or H3 byte except during negative justification, byte after V3 or H3 byte during positive justification, unused bytes in fraction rate links
Asynchronous T1/E1 Links			
K27.7-	110110 1000	-	V5 byte, no justification request
K28.7-	001111 1000	-	V5 byte, negative justification request
K29.7-	101110 1000	-	V5 byte, positive justification request
Synchronous T1/E1 Links			
K27.7-	110110 1000	-	V5 byte
Asynchronous DS3/E3 Links			
K27.7-	110110 1000	-	V5 byte, no justification request
K28.7-	001111 1000	-	V5 byte, negative justification request*
K29.7-	101110 1000	-	V5 byte, positive justification request*
Fractional Rate Links			
K28.7-	001111 1000	-	V5 byte, send one extra byte request**

Code Group Name	Curr. RD- abcdei fghj	Curr. RD+ abcdei fghj	Encoded Signals Description
K29.7-	101110 1000	-	V5 byte, send one less byte request**
Floating Transparent Virtual Tributaries			
K27.7-	110110 1000	-	V5 byte IV5=1, IDATA[0,4] = ERDI[1:0] = ‘b00, IDATA[5] = REI = ‘b0
K27.7+	-	001001 0111	V5 byte IV5=1, IDATA[0,4] = ERDI[1:0] = ‘b00, IDATA[5] = REI = ‘b1
K28.7-	001111 1000	-	V5 byte IV5=1, IDATA[0,4] = ERDI[1:0] = ‘b01, IDATA[5] = REI = ‘b0
K28.7+	-	110000 0111	V5 byte IV5=1, IDATA[0,4] = ERDI[1:0] = ‘b01, IDATA[5] = REI = ‘b1
K29.7-	101110 1000	-	V5 byte IV5=1, IDATA[0,4] = ERDI[1:0] = ‘b10, IDATA[5] = REI = ‘b0
K29.7+	-	010001 0111	V5 byte IV5=1, IDATA[0,4] = ERDI[1:0] = ‘b10, IDATA[5] = REI = ‘b1
K30.7-	011110 1000	-	V5 byte IV5=1, IDATA[0,4] = ERDI[1:0] = ‘b11, IDATA[5] = REI = ‘b0
K30.7+	-	100001 0111	V5 byte IV5=1, IDATA[0,4] = ERDI[1:0] = ‘b11, IDATA[5] = REI = ‘b1

* Note there can be multiple V5s per SBI frame when in DS3 or E3 mode but only one justification can occur per SBI frame. Positive and negative justification request through V5 required by the SBI336S interface should be limited to one per frame.

** Note fractional rate links are symmetric in the transmit and receive direction over SBI336S. When using clock slave mode with a fractional rate link the clock master makes single byte adjustments to the slaves rate once per frame.

9.3.2 Serial TelecomBus 8B/10B Character Encoding

Table 3 shows the mapping of TelecomBus control bytes and signals into 8B/10B control characters. When the TelecomBus control signals conflict each other, the 8B/10B control characters are generated according to the sequence of the table, with the characters at the top of the table taking precedence over those lower in the table.

Table 3 Serial TelecomBus Character Encoding

Code Group Name	Curr. RD- abcdei fghj	Curr. RD+ abcdei fghj	Encoded Signals Description
High Order Path Termination (HPT) Mode			
K28.5	001111 1010	110000 0101	IC1FP='b1 IPL='b0 C1FP frame and multiframe alignment
K28.0-	001111 0100	-	IPL='b0 High-order path H3 byte position, no negative justification event.
K28.0+	-	110000 1011	IPL='b0 High-order path PSO byte position, positive justification event.
K28.6	001111 0110	110000 1001	IC1FP='b1, IPL='b1 High-order path frame alignment (J1).
Low Order Path Termination (LPT) Mode			
K28.4+	-	110000 1101	ITAIS='b1 Low-order path AIS.
K27.7-	110110 1000	-	IV5='b1, IDATA[0,4] = ERDI[1:0] = 'b00, IDATA[5] = REI = 'b0 Low order path frame alignment. ERDI and REI are encoded in the V5 byte.
K27.7+	-	001001 0111	IV5='b1, IDATA[0,4] = ERDI[1:0] = 'b00, IDATA[5] = REI = 'b1 Low order path frame alignment. ERDI and REI are encoded in the V5 byte.
K28.7-	001111 1000	-	IV5='b1, IDATA[0,4] = ERDI[1:0] = 'b01, IDATA[5] = REI = 'b0 Low order path frame alignment. ERDI and REI are encoded in the V5 byte.
K28.7+	-	110000 0111	IV5='b1, IDATA[0,4] = ERDI[1:0] = 'b01, IDATA[5] = REI = 'b1 Low order path frame alignment. ERDI and REI are encoded in the V5 byte.
K29.7-	101110 1000	-	IV5='b1, IDATA[0,4] = ERDI[1:0] = 'b10, IDATA[5] = REI = 'b0 Low order path frame alignment. ERDI and REI are encoded in the V5 byte.
K29.7+	-	010001 0111	IV5='b1, IDATA[0,4] = ERDI[1:0] = 'b10, IDATA[5] = REI = 'b1 Low order path frame alignment. ERDI and

			REI are encoded in the V5 byte.
K30.7-	011110 1000	-	IV5='b1, IDATA[0,4] = ERDI[1:0] = 'b11, IDATA[5] = REI = 'b0 Low order path frame alignment. ERDI and REI are encoded in the V5 byte.
K30.7+	-	100001 0111	IV5='b1, IDATA[0,4] = ERDI[1:0] = 'b11, IDATA[5] = REI = 'b1 Low order path frame alignment. ERDI and REI are encoded in the V5 byte.
K23.7-	111010 1000	000101 0111	ITPL='b0 Non low-order path payload bytes.

9.3.3 Serial SBI336S and TelecomBus Alignment

The alignment functionality preformed by each receiver can be broken down into two parts, character alignment and frame alignment. Character alignment finds the 8B/10B character boundary in the arbitrarily aligned incoming data. Frame alignment finds SBI336S or TelecomBus frame and multiframe boundaries within the Serial link.

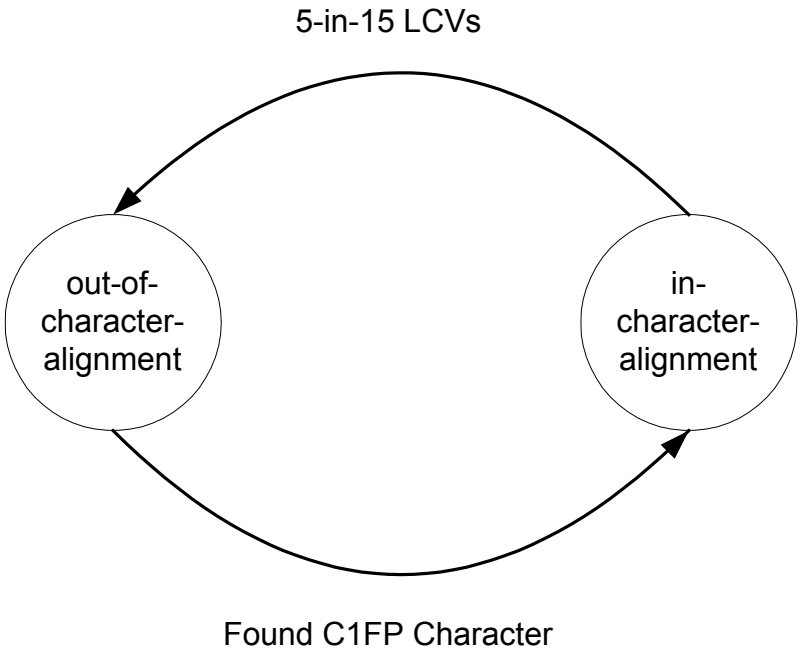
The character and frame alignment are expected to be robust enough for operation over a cabled interconnect.

9.3.4 Character Alignment Block

Character alignment locates character boundaries in the incoming 8B/10B data stream. The character alignment algorithm may be in one of two states, in-character-alignment state and out-of-character-alignment state. The two states of the character alignment algorithm is shown in Figure 8.

When the character alignment state machine is in the out-of-character-alignment state, it maintains the current alignment, while searching for a C1FP character. If it finds the C1FP character, it will re-align to the C1FP character and move to the in-character-alignment state. The C1FP character is found by searching for the 8B/10B C1FP character, K28.5+ or K28.5-, simultaneously in ten possible bit locations. While in the in-character-alignment state, the state machine monitors LCVs. If 5 or more LCVs are detected within a 15 character window the character alignment state machine transitions to out-of-character-alignment state. The special characters listed in Table 2 and Table 3 are ignored for LCV purposes. Upon return to in-character-alignment state the LCV count is cleared.

Figure 9 Character Alignment State Machine



9.3.5 Frame Alignment

Frame alignment locates SBI or TelecomBus frame and multiframe boundaries in the incoming 8B/10B data stream. The frame alignment state machine may be in one of two states, in-frame-alignment state and out-of-frame-alignment state. Each SBI336S frame is 125 µS in duration.

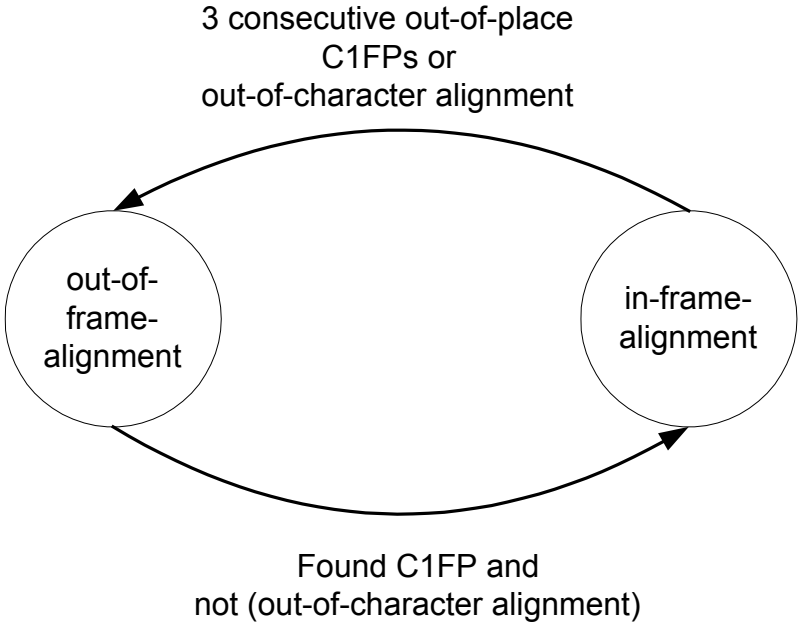
In SBI mode: Encoded over the SBI336S frame alignment is SBI336S multiframe alignment which is every four SBI336S frames or 500 µS. When carrying DS0 traffic in synchronous mode, signaling multiframe alignment is also necessary and is also encoded over SBI336S alignment. Signaling multiframe alignment is every 24 frames for T1 links and every 16 frames for E1 links, therefore signaling multiframe alignment covering both T1 and E1 multiframe alignment is every 48 SBI336S frames or 6ms. Therefore C1FP characters are sent every four or every 48 frames.

In TelecomBus mode: Encoded over the serial link is the tributary multiframe alignment which is every 4 frames or 500 µS. Multiframe alignment is required so that a downstream device can extract the T1 or E1 data from the tributary. The multiframe information is preserved by only sending out C1FP characters every four frames.

The frame alignment state machine establishes frame alignment over the link and is based on the frame and not the multiframe alignments. When the frame alignment state machine is in the out-of-frame-alignment state, it maintains the current alignment, while searching for a C1FP character. When it finds the C1FP character the state machine transitions to the in-frame-alignment state. While in the in-frame-alignment state, the state machine monitors out-of-place C1FP characters. Out-of-place C1FP characters are identified by maintaining a frame counter based on the C1FP character. The counter is initialized by the C1FP character when in the out-of-character-alignment state, and is unaffected in the in-character-alignment state. If three consecutive C1FPs have been found that do not agree with the expected location as defined by the frame counter, the state will change to out-of-frame-alignment state.

The frame alignment state machine is also sensitive to character alignment. When the character alignment state machine is in the out-of-character-alignment state, the frame alignment state machine is forced out-of-alignment, and is held in that state until the character alignment state machine transitions to the in-character alignment state.

Figure 10 Frame Alignment State Machine



9.3.6 SBI336S Multiframe Alignment

SBI336S multiframe alignment is communicated across the link by controlling the frequency of the C1FP character. The most frequent transmission of the C1FP character is every four SBI336S frame times. This is the SBI336S multiframe and is used when there are no synchronous tributaries requiring signalling multiframe alignment on the SBI336S bus. When there are synchronous tributaries on the SBI336S bus the C1FP character is transmitted every 48 frame times. This is the CAS signaling multiframe and is the lowest common multiple of the 24 frame T1 multiframe and the 16 frame E1 multiframe.

The SBI336S multiframe and signaling multiframe alignment is based a free running multiframe counter that is reset with each C1FP character received. Under normal operating conditions each received C1FP character will coincide with the free running multiframe counter. SBI336S multiframe alignment is always required, SBI336S signaling multiframe alignment is optional and only required when synchronous tributaries are supported with DS0 level switching.

9.4 DS0 Cross Bar switch (DCB)

Each of 12 R8TD blocks provides an eight-bit data signal on each 77.76 MHz clock edge. These signals are the STS-12 frame aligned ingress octets. Likewise, each of 12 egress T8TE blocks expects to receive a STS-12 frame aligned signal on each clock edge. The DS0 Cross Bar switch (DCB) connects these inputs to these outputs.

The DCB constitutes a Space switch that connects each output to some input during each clock period in the STS-12 frame structure. The STS-12 frame structure consists of 12*9*90 = 9720 octets (of overheads and payload). Being a DS0 granularity space switch, the DCB must provide separate switch settings for each of these 9720 octet times.

These 9720 switch settings are stored in an on-chip SRAM. Each of twelve egress ports must be told which of each of twelve ingress ports it should read during each of the 9720 clock periods. Five bits are required to specify which ingress port should be read by each output. Thus, we require 9720 words of five bits each for twelve egress ports. Thus each clock period requires 12 *5 = 60 bits. To support controlled switchover from one set of switch settings to another, we require two banks of 9720 words each. The aggregate memory requirement is 2 X 9720 X 60b = 1,166,400b of SRAM. Table 4 illustrates the mapping of this memory. Each control page in the table is a vector of 60 bits containing five bits (specifying the source port) for each of 12 egress ports. One page will be on-line translating ports in the core switch while the other is offline for CPU update. When the new configuration is ready, and the appropriate system synchronized frame boundary arrives, the pages will be swapped.

Table 4 Switching Control RAM Layout

	Control Page 0			Control Page 1		
RAM Address	**STS**	**Row**	**Col**	**STS**	**Row**	Col
0	1	1	1	1	1	1
1	2	1	1	2	1	1
…	…					
9719	12	9	90	12	9	90

The multiplexers that select the inputs for each egress port are straight forward 12 to-1 multiplexers. They require five bits of control during each 77.76 MHz clock cycle. Their outputs go to the T8TEs. This design permits unicast, multicast, and broadcast.

9.5 Clock Synthesis and Transmit Reference Digital Wrapper (CSTR)

The CSTR contains the configuration registers for the CSU and TXREF LVDS analog locks.

9.6 Fabric Latency

The flow of octets from ingress LVDS to egress LVDS has variable latency, depending on the timing of the arriving LVDS stream, and the clock variation on the egress LVDS drivers. A reasonable estimate of the NSE's latency can be arrived at by making assumptions about the depths of the receive and transmit FIFOs: we assume the "C1" timing is set to maintain about four samples in the ingress FIFO; the egress FIFO is designed to be centered at four samples – so typically delay due to FIFOs will be 8 clock cycles. The latency through the space switch stage is three clock cycles. Data latency through the analog blocks is around 90 ns. The typical latency of the NSE-8 G is 24 clock cycles or 308 ns. With worst case conditions in both FIFOs, latency rises to 36 clock cycles or 463 ns.

9.7 JTAG Support

The NSE-8G provides JTAG support for testing device interconnection on a PC board.

9.8 Microprocessor Interface

The Microprocessor Interface Block provides the logic required to interface the normal mode and test mode registers within the NSE-8G to a generic microprocessor bus. The normal mode registers are used during normal operation to configure and monitor the NSE. The register set is accessed as shown in the Register Memory Map table below. Addresses that are not shown are not used and must be treated as Reserved.

9.9 In-band Link Controller (ILC)

In order to permit centralized control of distributed NSE/SBS fabrics from the NSE-8G microprocessor interface (for applications in which NSEs are located on fabric cards, and SBSs are located on multiple line cards), an in-band signaling channel is provided between the NSE-8G and the SBS over the Serial SBI336S interface. Each NSE-8G can control up to 12 SBSs that are attached by the LVDS links. The NSE-SBS in-band channel is full duplex, but the NSE-8G has active control of the link.

The in-band channel is carried in the first 36 columns of four rows of the SBI structure, rows 3, 6, 7 and 8. The overall in-band channel capacity is thus 36*4*64 kb/s = 9.216 Mbit/s. Each 36 bytes per row allocated to the in-band signaling channel is its own in-band message between the end points. Four bytes of each 36 byte inband message are reserved for end-to-end control information and error protection, leaving 8.192 Mbit/s available for data transfer between the end points.

The data transferred between the end points has no fixed format, effectively providing a clear channel for packet transfer between the attached microprocessors at each of the LVDS link terminating devices. The user is able to send and receive any packet up to 32 bytes in length. The last two reserved bytes of the 36 byte in-band message is a CRC-16 which detects errors in the message. This block provides a microprocessor interface to the in-band signaling channel.

This in-band channel is expected to be used almost entirely to carry out switching control changes in the SBSs. To configure a DS0 in an SBS device most often requires a local microprocessor to write to one memory location consisting of a 16-bit address and a 16-bit data. Using this as a baseline and assuming an efficient use of the in-band channel bandwidth, we can set a maximum of (32bytes/row * 4 rows/frame * 8000 frames/sec / 4 bytes/write) 256,000 DS0 configurations per second.

Considering that configuring a T1 when switching DS0s requires 27 DS0 writes, indicates that the in-band signaling channel bandwidth sets maximum limit of over 9000 T1 configurations per second. In real life these limits will not be achieved but this shows that the in-band link should not be the bottleneck. In TelecomBus mode, this same configuration will require only three writes per T1 link. Another more efficient communication scheme could be used to increase this performance.

In N+1 protected architectures, it is likely that full configuration of a port card will be necessary during the switchover. This would require the entire connection memory be reconfigured. Assuming connections for overhead bytes are also reconfigured, the fastest that a complete reconfiguration can take place is 9720 register writes for each of the two configuration pages in the SBS. This equates to (2 * 9720 writes * 4 bytes/write / (32 bytes/row * 4 rows/frame * 8000 frames/second)) 76 milliseconds. It is also possible that the spare card could hold all the connection configurations for all the port cards it is protecting locally, for even faster switch over.

9.9.1 In-Band Signaling Channel Fixed Overhead

The In-Band Link Controller block generates and terminates two bytes of fixed header and a CRC-16 per every 32 byte in-band message (total 36 bytes). The two byte header provides control and status between devices at the ends of the LVDS link. The CRC-16 is calculated over the 32 byte (and header - 34 bytes) in-band message and provides the terminating end the ability to detect errors in the in-band message. The format of the in-band message and header bytes is shown in Figure 11 and Figure 12.

Figure 11 In-Band Signaling Channel Message Format

1 byte	1 byte	32 bytes	2 bytes
Header1	Header2	Free Format Information	CRC-16

Figure 12 In-Band Signaling Channel Header Format

Header1							
Bit 7	Bit 6	Bit 5	Bit4	Bit3	Bit2	Bit1	Bit 0
VALID	LINK[1:0]		PAGE[1:0]		USER[2:0]		

Header2							
Bit 7	Bit 6	Bit 5	Bit4	Bit3	Bit2	Bit1	Bit 0
AUX[7:0]							

Table 5 In-band Message Header Fields

Field Name	NSE-8G to SBS	SBS to NSE
Valid	Message slot contains a message(1) or is empty(0). If empty this message will not be put into Rx Message FIFO (other header information processed as usual)	Message slot contains a message(1) or is empty(0). If empty this message will not be put into Rx Message FIFO (other header information processed as usual)
Link[1:0]#	These bits are optional for SBI336S devices, intended for devices which have multiple redundant links. Each bit either indicates which Link to use, Working(0) or Protect(1) when sourced from the master device, or which link is being used, when sourced from the slave device. Other algorithms are possible to indicate Working or Protect over these 2 bits but all SBI336S devices must be able to revert back to this meaning. Transmitted immediately.	These bits are optional for SBI336S devices, intended for devices which have multiple redundant links. Each bit either indicates which Link to use, Working(0) or Protect(1) when sourced from the master device, or which link is being used, when sourced from the slave device. Other algorithms are possible to indicate Working or Protect over these 2 bits but all SBI336S devices must be able to revert back to this meaning.Transmitted immediately.

Field Name	NSE-8G to SBS	SBS to NSE
Page[1:0]#	Each bit indicates which control page to use, page 1 or 0, two bits, bit 1 for the ingress MSU and bit 0 for the egress MSU. Only transmitted from the beginning of the first message of the frame	Each bit shows current control page in use, page 1 or 0, two bits, bit 1 for the ingress MSU and bit 0 for the egress MSU. Only transmitted from the beginning of the first message of the frame.
User[2:0]#	User defined register indication to SBS reflected in the SBS as external hardware signal outputs. Transmitted immediately.	User defined register indication to NSE-8G from external hardware inputs to the SBS. Transmitted immediately.
Aux[7:0]#	User defined auxiliary register indication to SBS. Transmitted immediately.	User defined auxiliary register indication to NSE. Transmitted immediately.

Change in these bits(received side) will not be processed if the received message CRC-16 indicates an error.

Interrupts can be generated when CRC errors are detected or the USER or LINK bits change state. There is no inherent flow control provided by the In-Band Link Controller. The attached microprocessor is able to provide flow control via interrupts when the in-band message FIFO overflows and via the USER and Auxiliary bits in the header.

As each message arrives, the CRC-16 and valid bit is checked; if the valid bit is not set the message is discarded, if it fails the CRC check it is flagged as being in error and an interrupt is generated if enabled. If the CRC-16 is OK, regardless of the valid bit, the Page Link, User and Aux bits are passed on immediately. If the FIFO erroneously overflows, an interrupt is generated.

9.10 Microprocessor Interface

The following register map, Table 6, shows the registers used to provide control of the NSE.

The first 100h addresses provide access to the top level NSE-8G configuration and control registers, the Clock synthesis units through the CSTR blocks and the DSO Crossbar (DCB) The DCB is the space switch at the core of the NSE. From 100h are 12 identical, 20h spaces used to control the ports of the NSE-8G on an individual basis. Each port has an In-Band Link Controller (ILC), an 8B/10B encoder (T8TE) and an 8B/10B decoder (R8TD). These blocks provide functions specific to the ports such as Line Code Violation counts (for data integrity monitoring) and receive and transmit in-band link message buffers. Only port 0 is fully described as the other ports are identical, being incrementally distributed from address 100h in 20h steps.

Table 6 NSE-8G Register Map

Address	Register
000	NSE-8G Master Reset
001	NSE-8G Individual Channel Reset
002	NSE-8G Master JTAG ID
003	SBS Page select – Page 0
004	SBS Page select – Page 1
005	NSE-8G Master Interrupt Source
006	NSE-8G Master ILC Interrupt Source
007	NSE-8G Master R8TD Interrupt Source
008	NSE-8G Master T8TE Interrupt Source
009	NSE-8G Master Clock Monitor
00A	NSE-8G DCB CMP select
00B	NSE-8G Master Interrupt Enable
00C	NSE-8G Subsystem Interrupt Enable
00D	NSE-8G R8TD TIP
00E	SBS User Bits 0
00F	SBS User Bits 1
010	SBS User Bits 2
011	NSE-8G FREE User Register
012	R8TD_RX_C1 Pulse Monitor Register
013	Unexpected R8TD_RX_ C1 Interrupt Register
014	Missing R8TD_RX_ C1 Interrupt Register
015	Unexpected R8TD_RX_ C1 Interrupt Enable Register
016	Missing R8TD_RX_ C1 Interrupt Enable Register
017	R8TD C1 Disable
01C-01F	Reserved
020	CSTR #1 Control
021	CSTR #1 Interrupt Enable and CSU Lock Status
022	CSTR #1 Interrupt Indication
023	Reserved
024	Reserved
025	Reserved
026	Reserved
027-03F	Reserved
040	Reserved
041	Reserved
042	Reserved
043	Reserved
044	DCB CONFIGURATION PORT 11-6 REGISTER
045	DCB CONFIGURATION PORT 5-0 REGISTER

Address	Register
046	DCB CONFIGURATION OUTPUT REGISTER
047	DCB ACCESS MODE REGISTER
048	DCB C1 DELAY (RC1FP) REGISTER
04A	DCB FRAME SIZE REGISTER
04C	DCB CONFIGURATION REGISTER
04D	DCB INTERRUPT REGISTER
04E – 0FF	Reserved
100-1FF	Port Register Set 0 – Port 0 (Channel 0)
100	Port Register Set 0: R8TD Control and Status
101	Port Register Set 0: R8TD Interrupt Status
102	Port Register Set 0: R8TD LCV Count
103	Port Register Set 0: RXLV and DRU Control
0106 – 107	Port Register Set 0: Reserved
108	Port Register Set 0: T8TE Control and Status
109	Port Register Set 0: T8TE Interrupt Status
10A	Port Register Set 0: T8TE Time-slot Configuration #1
10B	Port Register Set 0: T8TE Time-slot Configuration #2
10C	Port Register Set 0: T8TE Test Pattern
10D	Port Register Set 0: TXLV and PISO Control
10F	Port Register Set 0: Reserved
110	Port Register Set 0: ILC Transmit Message FIFO Data
111	Port Register Set 0: ILC Transmit Control
112	Port Register Set 0: ILC Transmit Status and FIFO Synch
113	Port Register Set 0: ILC Receive Message FIFO DATA
114	Port Register Set 0: ILC Receive Control
115	Port Register Set 0: ILC Receive Status and FIFO Synch
116	Port Register Set 0: ILC Interrupt enable and Control
117	Port Register Set 0: ILC Interrupt reason Register
118-11F	Reserved
120-13F	Port Register Set 1 – Port 1 (Channel 1)
140-15F	Port Register Set 2 – Port 2 (Channel 2)
160-17F	Port Register Set 3 – Port 3 (Channel 3)
180-19F	Port Register Set 4 – Port 4 (Channel 4)
1A0-1BF	Port Register Set 5 – Port 5 (Channel 5)
1C0-1DF	Port Register Set 6 – Port 6 (Channel 6)
1E0-1FF	Port Register Set 7 – Port 7 (Channel 7)
200-21F	Port Register Set 8 – Port 8 (Channel 8)
220-23F	Port Register Set 9 – Port 9 (Channel 9)
240-25F	Port Register Set 10 – Port 10 (Channel 10)
260-27F	Port Register Set 11 – Port 11 (Channel 11)

Address	Register
280-29F	Port Register Set 12 – Port 12 (Channel 12
2A0-7FF	Reserved
800 – FFF	Reserved for Test

Notes on Register Memory Map:

1. For all register accesses, CSB must be low.
2. Addresses that are not shown must be treated as Reserved.

10 Normal Mode Register Description

Normal mode registers are used to configure and monitor the operation of the NSE. Normal mode registers (as opposed to test mode registers) are selected when A[11] is set low.

Notes on Normal Mode Register Bits:

1. Writing values into unused register bits has no effect. However, to ensure software compatibility with future, feature-enhanced versions of this product, unused register bits must be written with logic 0. Reading back unused bits can produce either a logic one or a logic 0; hence, unused register bits should be masked off by software when read.

2. All configuration bits that can be written into can also be read back. This allows the processor controlling the TSB to determine the programming state of the block.

3. Writeable normal mode register bits are cleared to logic 0 upon reset unless otherwise noted.

4. Writing into read-only normal mode register bit locations does not affect NSE-8G operation unless otherwise noted.

5. For registers above 100H, only a one port set of the 12 ports are shown. The Register addresses are shown for example as: 0100H + N*20H, N here is the port number between 0 and 11. This is done to prevent unnecessary duplication of otherwise identical register sets.

Register 000H: NSE-8G Master Reset

Bit	Type	Function	Default
Bit 31	R/W	DRESET	0
Bit 30	R/W	ARESET	0
Bit 29:0	R	Unused	X

This register allows separate software reset of digital and analog circuitry on the NSE.

ARESET

The ARESET bit allows the analog circuitry in the NSE-8G to be reset under software control. If the ARESET bit is a logic one, all the NSE-8G analog circuitry is held in reset. ARESET must be held at logic one for at least 100us to ensure correct reset of the CSU. This bit is not self-clearing. Therefore, a logic zero must be written to bring the NSE-8G out of reset. Holding the NSE-8G in a reset state places it into a low power, analog stand-by mode. A hardware reset clears the ARESET bit, thus negating the analog software reset.

DRESET

The DRESET bit allows the digital circuitry in the NSE-8G to be reset under software control. If the DRESET bit is a logic one, all the NSE-8G digital circuitry is held in reset. This bit is not self-clearing. Therefore, a logic zero must be written to bring the NSE-8G out of reset. Holding the NSE-8G in a reset state places it into a low power, digital stand-by mode. A hardware reset clears the DRESET bit, thus negating the digital software reset.

Register 001H: NSE-8G Individual Channel Reset

Bit	Type	Function	Default
Bit 11:0	R/W	RESET[11:0]*	1

This register allows power saving by holding individual channels in reset.

RESET[n]

The RESET[n] bit allows the channel circuitry in the NSE-8G to be reset under software control. If the RESET[n] bit is a logic one, the NSE-8G channel circuitry for a particular channel is held in reset. RESET[n] does not affect the reset of the CSU. This bit is not self-clearing. Therefore, a logic zero must be written to bring the channel out of reset. Holding the channel in a reset state places it into a low power, analog stand-by mode. A hardware reset or software DRESET bit 000h sets the RESET[n] bit.

Register 002H: NSE-8G Master JTAG ID

Bit	Type	Function	Default hex
Bit 31:28	R	ID[3:0]	0h
Bit 27:12	R	DEVID[15:0]	8621h
Bit 11:0	R	JTAG Identification Code [MID[10:0] & JID]	0CDh

The NSE-8G Master JTAG ID registers hold the jtag identification code for the device. The device revision number and device id are available through these registers.

ID[3:0]

The ID bits can be read to provide a binary NSE-8G revision number.

DEVID[15:0]

The DEVID bits can be read to distinguish the NSE-8G from other members of the NSE-8G family of devices.

MID[10:0]

The MID bits provide the manufacturer identity field in the JTAG identification code.

JID

The JID bit is bit 0 in the JTAG identification code.

Register 003H: SBS Page select – Page 0

Bit	Type	Function	Default
Bit 11:0	R/W	Page0_SBS[11:0]*	0

Page0_SBS[n]

This bit will be the Page 0 bit sent to SBSn over the In-Band channel – where n is any SBS connected to LVDS links numbered from 0 to 11*.

Register 004H: SBS Page select – Page 1

Bit	Type	Function	Default
Bit 11:0	R/W	Page1_SBS[11:0]*	0

Page1_SBS[n]

This bit will be the Page 1 bit sent to SBSn over the In-Band channel – where n is any SBS connected to LVDS links numbered from 0 to 11*.

Register 005H: NSE-8G Master Interrupt Source

Bit	Type	Function	Default
Bit 31:8	R	Unused	X
Bit 7	R	R8C1EXTRAINT	0
Bit 6	R	R8C1MISSINT	0
Bit 5	R	Reserved	0
Bit 4	R	CSU1INT	0
Bit 3	R	R8TDINT	0
Bit 2	R	T8TEINT	0
Bit 1	R	ILCINT	0
Bit 0	R	DCBINT	0

R8C1EXTRAINT

If the R8C1EXTRAINT bit is a logic one, an interrupt of unexpected C1 character in one of the R8TD_C1_INT blocks has occurred. The source of the R8C1EXTRAINT bit comes from the Register 013h.

R8C1MISSINT

If the R8C1MISSINT bit is a logic one, an interrupt of missing C1 characters in one of the R8TD_C1_INT blocks has occurred. The source of the R8C1MISSINT bit comes from the Register 014h.

CSU1INT

If the CSU1INT bit is a logic one, an interrupt has been generated by CSU #1. The CSTR Interrupt register must be read to clear this interrupt.

R8TDINT

If the R8TDINT bit is a logic one, an interrupt has been generated by one of the R8TD blocks. The internal R8TD Interrupt register must be read to clear this interrupt. Which R8TD caused the interrupt can be ascertained by reading the NSE-8G R8TD Interrupt Source Register.

T8TEINT

If the T8TEINT bit is a logic one, an interrupt has been generated by one of the T8TE blocks. The internal T8TE Interrupt register must be read to clear this interrupt. Which T8TE caused the interrupt can be ascertained by reading the NSE-8G T8TE Interrupt Source Register.

ILCINT

If the ILCINT bit is a logic one, an interrupt has been generated by one of the ILC blocks. The relevant ILC Interrupt register must be read to clear this interrupt. Which ILC caused the interrupt can be ascertained by reading the NSE-8G ILC Interrupt Source Register.

DCBINT

If the DCBINT bit is a logic one, an interrupt has been generated by the DCB block. The DCB Interrupt register must be read to clear this interrupt.

Register 006H: NSE-8G Master ILC Interrupt Source

Bit	Type	Function	Default
Bit 11:0	R	ILCINT[11:0]*	0

ILCINT[n]

If the ILCINT[n] bit is a logic one, an interrupt has been generated by that ILC block. The relevant ILC Interrupt register must be read to clear this interrupt.

Register 007H: NSE-8G Master R8TD Interrupt Source

Bit	Type	Function	Default
Bit 11:0	R	R8TDINT[11:0]*	0

R8TDINT[n]

If the R8TDINT[n] bit is a logic one, an interrupt has been generated by that R8TD block. The relevant R8TD Interrupt register must be read to clear this interrupt.

Register 008H: NSE-8G Master T8TE Interrupt Source

Bit	Type	Function	Default
Bit 11:0	R	T8TEINT[11:0]*	0

T8TEINT[n]

If the T8TEINT[n] bit is a logic one, an interrupt has been generated by that T8TE block. The relevant T8TE Interrupt register must be read to clear this interrupt

Register 009H: NSE-8G Master Clock Monitor

Bit	Type	Function	Default
Bit 31:2	R	Unused	X
Bit 1	R	RC1FPA	X
Bit 0	R	SYSCLKA	X

When a monitored clock signal makes a low to high transition, the corresponding register bit is set high. The bit will remain high until this register is read, at which point all the bits in this register are cleared. A lack of transitions is indicated by the corresponding register bit reading low. This register should be read at periodic intervals to detect clock failures.

SYSCLKA

The SYSCLK active bit (SYSCLKA) detects low to high transitions on the SYSCLK input. SYSCLKA is set high on a rising edge of SYSCLK, and is set low when this register is read.

RC1FPA

The RC1FP active bit (RC1FPA) detects low to high transitions on the RC1FP input. RC1FPA is set high on a rising edge of RC1FP, and is set low when this register is read.

Register 00AH: NSE-8G DCB CMP select

Bit	Type	Function	Default
Bit 31:2	R	Unused	X
Bit 1	R/W	CMP_SRC	0
Bit 0	R/W	CMP_VAL	0

The connection memory page select signal (CMP) controls the selection of the connection memory page in the NSE. When CMP is set high, connection memory page 1 is selected. When CMP is set low, connection memory page 0 is selected. Changes to the connection memory page selection are synchronized to the boundary of the next C1FP multiframe.

This Register controls a software override to the CMP pin.

CMP_SRC

This bit dictates whether CMP is to be sourced from software when set to ‘1’ or from the external CMP pin when set to 0.

CMP_VAL

CMP_VAL is used to provide the CMP signal when CMP_SRC is set to ‘1’ other wise this bit is ignored.

Register 00BH: NSE-8G Interrupt Enable Register

Bit	Type	Function	Default
Bit 31:1	R	Unused	X
Bit 0	R/W	INTE	0

This register allows the CPU to disable or enable NSE-8G interrupts with a single write.

INTE

This bit, when '1', enables the INTB pin on the NSE. When set to '0' INTB is held '1'.

Register 00CH: NSE-8G Subsystem Interrupt Enable Register

Bit	Type	Function	Default
Bit 31:6	R	Unused	X
Bit 5	R/W	TOPINTE	0
Bit 4	R/W	CSUINTE	0
Bit 3	R/W	R8TDINTE	0
Bit 2	R/W	T8TEINTE	0
Bit 1	R/W	ILCINTE	0
Bit 0	R/W	DCBINTE	0

This register allows the CPU to disable or enable NSE-8G Subsystem interrupts with a single write.

TOPINTE

This bit, when ‘1’, enables the generation of interrupts from the Top_level i.e. R8C1EXTRAINT and R8C1MISSINT interrupts. When set to ‘0’ R8C1EXTRAINT and R8C1MISSINT interrupts are disabled .

CSUINTE

This bit, when ‘1’, enables the generation of interrupts from CSU1 control. When set to ‘0’ CSU1 control interrupts are disabled .

R8TDINTE

This bit, when ‘1’, enables the generation of interrupts from R8TD blocks. When set to ‘0’ all R8TD interrupts are disabled .

T8TEINTE

This bit, when ‘1’, enables the generation of interrupts from T8TE blocks. When set to ‘0’ all T8TE interrupts are disabled .

ILCINTE

This bit, when ‘1’, enables the generation of interrupts from ILC blocks. When set to ‘0’ all ILC interrupts are disabled .

DCBINTE

This bit, when ‘1’, enables the generation of interrupts from the DCB block. When set to ‘0’ DCB interrupts are disabled .

Register 00DH: NSE-8G R8TD TIP register

Bit	Type	Function	Default
Bit 31:1	R	Unused	X
Bit 0	R	TIP	0

This register allows the CPU to determine if the TIP signals from all the R8TDs are inactive indicating no transfers in progress.

TIP

This bit, when '1', indicates one or more of the TIP signals from each of the R8TDs is active. It is the result of all TIP signals ORed together.

Register 00EH: SBS User Bit 0

Bit	Type	Function	Default
Bit 11:0	R/W	SBS_USER_0[11:0]*	0

SBS_USER_0[n]

This bit will be the USER 0 bit sent to SBSn over the In-Band channel – where n is any SBS connected to LVDS links numbered from 0 to 11*

Register 00FH: SBS User Bit 1

Bit	Type	Function	Default
Bit 11:0	R/W	SBS_USER_1[11:0]*	0

SBS_USER_1[n]

This bit will be the USER 1 bit sent to SBSn over the In-Band channel – where n is any SBS connected to LVDS links numbered from 0 to 11*

Register 010H: SBS User Bit 2

Bit	Type	Function	Default
Bit 11:0	R/W	SBS_USER_2[11:0]*	0

SBS_USER_2[n]

This bit will be the USER 2 bit sent to SBSn over the In-Band channel – where n is any SBS connected to LVDS links numbered from 0 to 11*.

Register 011H: NSE-8G FREE User Register

Bit	Type	Function	Default
Bit 31:8	R	Unused	X
Bit 7:0	R/W	FREE[7:0]	0

FREE[7:0]

The software ID register (FREE) holds whatever value is written into it. Reset clears the contents of this register. This register has no impact on the operation of the NSE.

Register 012H: Correct R8TD_RX_C1 Pulse Monitor

Bit	Type	Function	Default
Bit 11:0	R	R8C1_OK_INT[11:0]	0

R8C1_OK_INT[n]

This bit will be set to '1' if both oc1fp[n] and r8_rx_c1[n] have occurred at the same time. Otherwise, it will be stay at '0'. Read access will clear this bit.

Section 12.5 describes the proper use of this register.

Register 013H: Unexpected R8TD_RX_C1 Interrupt

Bit	Type	Function	Default
Bit 11:0	R	R8C1_EXTRA_INT[11:0]	0

R8C1_EXTRA_INT[n]

This bit will be set to '1' if oc1fp[n] has not occurred at the time when r8_rx_c1[n] has occurred. Otherwise, it will stay at '0'. Read access will clear this bit.

Section 12.5 describes the proper use of this register.

Register 014H: Missing R8TD_RX_C1 Interrupt

Bit	Type	Function	Default
Bit 11:0	R	R8C1_MISS_INT[11:0]	0

R8C1_MISS_INT[n]

This bit will be set to '1' if r8_rx_c1[n] has not occurred at the time when oc1fp[n] has occurred. Otherwise, it will stay at '0'. Read access will clear this bit.

Section 12.5 describes the proper use of this register.

Register 015H: Unexpected R8TD_RX_C1 Interrupt Enable

Bit	Type	Function	Default
Bit 11:0*	R/W	R8C1_EXTRA_INTE[11:0]	0

R8C1_EXTRA_INTE[n]

R8C1_EXTRA_INTE[n] is used to enable/disable ('1' for enable; '0' for disable) the R8C1_EXTRA_INT[n] (defined in Reg 013h) to cause interrupt. This is on per channel* basis.

*Any unused ports must be set to '0'.

Register 016H: Missing R8TD_RX_C1 Interrupt Enable

Bit	Type	Function	Default
Bit 11:0*	R/W	R8C1_MISS_INTE[11:0]*	0

R8C1_MISS_INTE[n]

R8C1_MISS_INTE[n] is used to enable/disable ('1' for enable; '0' for disable) the R8C1_MISS_INT[n] (defined in Reg 014h) to cause interrupt. This is on per channel* basis.

*Any unused ports must be set to '0'

Register 020H, 024H: CSTR #1 – 2 Control*

Bit	Type	Function	Default
Bit 31-16		Unused	X
Bit 15	R/W	Reserved[11]	0
Bit 14	R/W	Reserved[10]	0
Bit 13	R/W	Reserved[9]	0
Bit 12	R/W	Reserved[8]	0
Bit 11	R/W	Reserved[7]	0
Bit 10	R/W	Reserved[6]	1
Bit 9	R/W	Reserved[5]	0
Bit 8	R/W	Reserved[4]	0
Bit 7	R/W	Reserved[3]	0
Bit 6	R/W	Reserved[2]	0
Bit 5	R/W	Reserved[1]	0
Bit 4	R/W	CSU_ENB	0
Bit 3	R/W	CSU_RSTB	1
Bit 2		Unused	X
Bit 1		Unused	X
Bit 0	R/W	Reserved[0]	1

This register provides reset control and enable control for CSTR block #1

Reserved[11:0]

The Reserved bits must be set to the indicated default value for correct operation of the NSE.

CSU_RSTB

The CSU_RSTB signal is a software reset signal that forces the CSU into reset. The CSU is reset when the CSU_RSTB is logic 0. The CSU is also reset by the NSE-8G master analog reset signal. When the CSU is reset, the reset signal should be held for at least 100us.

CSU_ENB

The CSU enable control signal (CSU_ENB) bit forces the CSU into low power configuration. The CSU is disabled when CSU_ENB is logic one. The CSU is enabled when CSU_ENB is logic 0.

Register 021H, 025H: CSTR #1 – 2* Interrupt Enable and CSU Lock Status

Bit	Type	Function	Default
Bit 31-2	R	Unused	X
Bit 1	R	LOCKV	X
Bit 0	R/W	LOCKE	0

This register configures the operation of CSTR block #1.

LOCKE

The CSU lock interrupt enable bit (LOCKE) controls the contribution of CSU lock state interrupts by the CSTR to the device interrupt INTB. When LOCKE is high, INTB is asserted low when the CSU lock state changes. Interrupts due to CSU lock state are masked when LOCKE is set low.

LOCKV

The CSU lock status bit (LOCKV) indicates whether the clock synthesis unit has successfully locked with the system clock. LOCKV is set low when the CSU has not successfully locked with the reference clock. LOCKV is set high if when the CSU has locked with the reference clock.

Register 022H, 026H: CSTR #1 – 2* Interrupt Indication

Bit	Type	Function	Default
Bit 31-1	R	Unused	X
Bit 0	R	LOCKI	X

LOCKI

The CSU lock interrupt status bit (LOCKI) reports and acknowledges changes in the CSU lock state. LOCKI is set high when the CSU achieves lock with the reference clock or loses its lock to the reference clock. LOCKI is cleared on a read to this register when WCIMODE is logic 0. LOCKI is cleared on a write (of any value) to this register when WCIMODE is logic one. INTB is asserted low when both LOCKE and LOCKI are high. If LOCKE is asserted, LOCKI must be cleared before INTB will be reasserted.

Register 044H: DCB Configuration port 11-6 Register

Bit	Type	Function	Default
Bit 31-30	R	Unused	X
Bit 29-25	R/W	Port11[4:0]	0
Bit 24-20	R/W	Port10[4:0]	0
Bit 19-15	R/W	Port9[4:0]	0
Bit 14-10	R/W	Port8[4:0]	0
Bit 9-5	R/W	Port7[4:0]	0
Bit 4-0	R/W	Port6[4:0]	0

Port11[4:0]

This register selects the input port number to map to output port 11 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port10[4:0]

This register selects the input port number to map to output port 10 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port9[4:0]

This register selects the input port number to map to output port 9 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port8[4:0]

This register selects the input port number to map to output port 8 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port7[4:0]

This register selects the input port number to map to output port 7 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port6[4:0]

This register selects the input port number to map to output port 6 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Register 045H: DCB Configuration port 5-0 Register

Bit	Type	Function	Default
Bit 31-30	R	Unused	X
Bit 29-25	R/W	Port5[4:0]	0
Bit 24-20	R/W	Port4[4:0]	0
Bit 19-15	R/W	Port3[4:0]	0
Bit 14-10	R/W	Port2[4:0]	0
Bit 9-5	R/W	Port1[4:0]	0
Bit 4-0	R/W	Port0[4:0]	0

Port5[4:0]

This register selects the input port number to map to output port 5 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port4[4:0]

This register selects the input port number to map to output port 4 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port3[4:0]

This register selects the input port number to map to output port 3 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port2[4:0]

This register selects the input port number to map to output port 2 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port1[4:0]

This register selects the input port number to map to output port 1 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Port0[4:0]

This register selects the input port number to map to output port 0 of the DCB for an arbitrary position in the SBI336/TelecomBus frame.

Register 046H: DCB Configuration Output Register.

Bit	Type	Function	Default
Bit 31-30	R	Unused	X
Bit 29-0	R	CFG_O[29:0]	0

CFG_O[29:0]

This field contains configuration data read from the offline connection memory page. Configuration data in this field is read from the location specified by the WORDADDR and PORTADDR fields in the Access Mode register. There is a 6 SYSCLK cycle latency from when an indirect read is requested until when correct data appears in this register.

Register 047H: DCB Access Mode Register

Bit	Type	Function	Default
Bit 31	R/W	WRB	1
Bit 30	R/W	ACCMDE	0
Bit 29	R	Unused	X
Bit 28-24	R/W	PORTCFG[4:0]	0
Bit 23-21	R	Unused	X
Bit 20-16	R/W	PORTADDR[4:0]	0
Bit 15-14	R	Unused	X
Bit 13-0	R/W	WORDADDR [13:0]	0

Writing to this register with the WRB register bit set high initiates an indirect read from the offline connection memory page. WORDADDR selects the offline connection memory page to read from. There is a latency of 6 SYSCLK cycles from when this register is written to with the WRB bit set high until when valid data appears on the Configuration Output register. Indirect reads should be spaced at least 6 SYSCLK cycles apart to permit valid data to appear in the Configuration Output register.

Writing to this register with the WRB register bit set low initiates an indirect write to the offline connection memory page. WORDADDR selects the offline connection memory page to write to. Indirect writes should be spaced at least 4 SYSCLK cycles apart to ensure the writes complete successfully.

While page copy is in progress (UPDATEV register bit = ‘1’), writing to this register will NOT cause data to be updated to/from the offline connection memory page.

While a page swap is pending (SWAPV register bit = ‘1’), writing to this register MAY cause unpredictable results as data may be transferred while a page swap is occurring, causing data to be updated to a different connection memory page from the intended.

WRB

The indirect access control bit selects between a write (0) or read (1) access to the offline connection memory page.

ACCMDE

These bits indicate the access mode of the offline connection memory page.

0 : PORT transfer mode.

1 : WORD transfer mode.

In port transfer mode, one port is updated per word of the offline connection memory page.

PORTCFG: new port mapping to be updated to the connection memory page.

WORDADDR: specifies the address of the offline connection memory page.

PORTADDR: port address of the offline connection memory page.

In word transfer mode, an entire word of the offline connection memory page is updated.

PORTCFG: is ignored.

WORDADDR: specifies the address to the offline connection memory page.

PORTADDR: is ignored.

In either mode, the contents read from the off-line connection memory page can be read by the microprocessor through the Configuration Output register..

PORTCFG[4:0]

This field contains the input port mapping to a particular output port specified in PORTADDR. Used only in PORT transfer mode. At all other modes, this field is ignored.

PORTADDR[4:0]

When performing writes to the offline connection memory page, this field indicates the output port to be updated with new mapping in PORTCFG. A PORTADDR of 0 relates to output port 0 of the DCB.

This field is valid in PORT transfer mode and during reading from the Configuration Output register and is ignored in WORD transfer mode. Valid values are 0-31 when performing writes.

When performing reads through the Configuration Output register, PORTADDR indicates the ports being read as follows:

000xx : ports 5-0

001xx : ports 11-6

010xx : ports 17-12

011xx : ports 23-18

100xx : ports 29-24

101xx : ports 31-30 on least significant bits

110xx : ports 5-0

111xx : ports 5-0

WORDADDR[13:0]

This field indicates the address of the update connection memory page to be accessed. This field relates to the time location within the SBI/TeleCombus frame. I.e. Location 0 would be the first A1 byte of the frame and location 24 is the C1 character.

This field is ignored in page copy mode. Valid values are 0-9719.

Register 048H: DCB C1 delay (RC1DLY) register.

Bit	Type	Function	Default
Bit 31-6	R	Unused	X
Bit 13-0	R/W	RC1DLY[13:0]	0

RC1DLY

This value, equaling the delay (in 77.76 MHz clock periods), between RC1FP and the arrival of the C1 characters in the R8TD. This delay will synchronize the C1 input to the R8TD blocks assuming all the C1 characters have arrived. As the delay on those links is dependent on the system design, backplane propagation delays, cable lengths etc. This value will have to be arrived at empirically. And will have an upper an lower limit for which the middle value should be selected. The Operations section for more detail and some recommended starting values.

MF_SWAP Legal Range (clock cycles)

00 26 – 9716

01 26 – 16383

10 26 – 16383

11 26 – 16383

Register 04AH: DCB Frame size Register

Bit	Type	Function	Default
Bit 31-14	R	Unused	X
Bit 13-0	R/W	FRMSZ[13:0]	9719

This register specifies the frame size of the SBI or TelecomBus frame.

FRMSZ[13:0]

This register specifies the size of the connection memory page in the various switching modes. Legal values:

1079: TelecomBus switching.

1079: SBI column switching.

9719: SBI DS0 switching.

9719: SBI DS0 switching with CAS.

Register 04CH: DCB Configuration Register

Bit	Type	Function	Default
Bit 31- 8	R	Unused	X
Bit 7-6	R/W	MF_SWAP[1:0]	0
Bit 5	R/W	AUTO	0
Bit 4	R/W	SWAP_PE	0
Bit 3	R/W	UPDATEE	0
Bit 2	R/W	FRAMEE	0
Bit 1	R	SWAPPV	0
Bit 0	R	UPDATEV	0

MF_SWAP [1:0]

This bit selects when RC1FP is expected and synchronizes when page swaps can occur. Table below relates MFSWAP to all vital variables from the DCB:

MFSWAP	Configuration Page Size	Frame Switching @ (9720 byte frame)	Frame Interrupt	RC1FP expected every	Switching Mode
00	1080	1 frame	1 frame	4 frame	TelecomBus
01	1080	4 frame	4 frame	4 frame	SBI column mode
10	9720	4 frame	4 frame	4 frame	SBI DS0 mode
11	9720	48 frame	48 frame	48 frame	SBI DS0 with CAS

AUTO

This bit enables an automatic copy of the online connection memory page to the offline connection memory page after the connection memory page is switched. Toggling the AUTO bit to '0' while a page copy is in progress will terminate the page copy process.

0: automatic update disabled.

1: automatic update enabled.

If automatic page copying is used, the page copy will take place automatically whenever the connection memory page swaps. This means that the UPDATEV register bit will be asserted immediately following a change from 1 to 0 in the SWAPV register bit. When the AUTO bit is set, access to the offline connection memory page is restricted from when a page swap is pending until when the page copy is complete.

SWAP_PE

This bit enables the propagation of interrupt to the INT output due to a change in state of SWAPV. This bit does not have an impact on SWAPI bit.

0: disables interrupt propagation to the INT output.

1: enables interrupt propagation to the INT output.

UPDATEE

This bit enables the propagation of interrupt to the INT output when UPDATEV changes state from 1 to 0. This bit does not have impact on UPDATEI bit

0: disables interrupt propagation to the INT output.

1: enables interrupt propagation to the INT output.

FRAMEE

This bit enables the propagation of interrupt to the INT output when CMP is sampled at the expected RC1FP position. This bit does not have an impact on FRAMEI bit.

0: disables interrupt propagation to the INT output.

1: enables interrupt propagation to the INT output.

SWAPV

The SWAPV bit contains the current state of the page swap. This bit is logic one when a switch to the connection memory page (CMP) input has been recognized but the page swap has not yet happened. This bit is a logic 0 when page swap is not pending.

When a page swap is pending, writing to the offline page or initiating a page copy may cause corruption of the memory pages.

UPDATEV:

This bit is updated when the active connection memory page is copied to the offline connection memory page.

0: copying completed.

1: copying in progress.

The duration of a page copy is highly dependent on MF_SWAP.

MF_SWAP	SYSCLK Clock cycles required
“00”	1083

“01”	1083
“10”	9723
“11”	9723

When a page copy is in progress, attempting to write to the offline connection memory page will be ignored and attempting to read from the offline connection memory page will return unpredictable results.

Register 04DH: DCB Interrupt status Register.

Bit	Type	Function	Default
Bit 31-3	X	Unused	0
Bit 2	I	SWAPI	X
Bit 1	I	UPDATEI	X
Bit 0	I	FRAMEI	X

Writing to this register initiates copying of the active connection memory page to the offline connection memory page. When a page swap is pending (SWAPV ='1') writing to this register may cause a corruption of the connection memory pages.

SWAPI

This bit reports and acknowledges a change of state in the SWAPV bit of the Configuration register. This bit is cleared when this register is read. When enabled by SWAPE, the INT output reflects the state of this bit.

UPDATEI

The offline page copy interrupt status bit, UPDATEI reports and acknowledges a change of state from 1 to 0 in the UPDATEV bit of the Configuration register. This signifies that a page copy is complete. This bit is cleared when read. When enabled by the UPDATEE bit, the INT output reflects the state of this bit.

FRAMEI

The frame interrupt status bit reports the sampling of the CMP bit at the expected RC1FP position. When enabled by FRAMEE, frequency of occurrence of FRAMEI is dependent on MF_SWAP. When enabled by the FRAMEE bit, the INT output reflects the state of this bit.

MF_SWAP FRAMEI occurs every

00	1 frame
01	4 frame
10	4 frame
11	48 frame

This bit is cleared when read.

A change in the CMP input should be sequenced to occur as soon as possible after the occurrence of FRAMEI. Changing CMP prior to the occurrence of FRAMEI may cause unpredictable behavior as it may cause CMP to be sampled later than expected.

Register 100H + N*20H, R8TD Control and Status

Bit	Type	Function	Default
Bit 31:16	R	Unused	X
Bit 15	R/W	Reserved[1]	0
Bit 14	R/W	Reserved[0]	0
Bit 13:10	R	Unused	X
Bit 9	R/W	RXINV	0
Bit 8	R/W	Reserved[2]	0
Bit 7	R/W	FUOE	0
Bit 6	R/W	LCVE	0
Bit 5	R/W	OFAE	0
Bit 4	R/W	OCAE	0
Bit 3	R	OFAV	X
Bit 2	R	OCAV	X
Bit 1	R/W	FOFA	0
Bit 0	R/W	FOCA	0

This register provides control and reports the status of the R8TD blocks.

FOCA

The force out-of-character-alignment bit (FOCA) control the operation of the character alignment block in the R8TD block. A 0-1 transition on this bit forces the character alignment block to the out-of-character-alignment state where it will search for the transport frame alignment character (K28.5). Before another force operation can be performed, FOCA must first be set to logic 0.

FOFA

The force out-of-frame-alignment bit (FOFA) controls the operation of the frame alignment block in the R8TD block. A 0-1 transition on this bit forces the frame alignment block to the out-of-frame-alignment state where it will search for the transport frame alignment character (K28.5). Before another force operation can be performed, FOFA must first be set to logic 0.

OCAV

The out-of-character-alignment status bit (OCAV) reports the state of the character alignment block in the R8TD block. OCAV is set high when the character alignment block is in the out-of-character-alignment state. OCAV is set low when the character alignment block is in the in-character-alignment state.

OFAV

The out-of-frame-alignment status bit (OFAV) reports the state of the frame alignment block in the R8TD block. OFAV is set high when the frame alignment block is in the out-of-frame-alignment state. OFAV is set low when the frame alignment block is in the in-frame-alignment state.

OCAE

The out of character alignment interrupt enable bit (OCAE) masks the contribution of the change of character alignment event indication bit (OCAI) in the R8TD block to INTB. When OCAE is high, INTB is asserted low when OCAI is high. INTB is not affected by the value of OCAI when OCAE is low.

OFAE

The out of frame alignment interrupt enable bit (OFAE) masks the contribution of the change of frame alignment event indication bit (OFAI) in the R8TD block to INTB. When OFAE is high, INTB is asserted low when OFAI is high. INTB is not affected by the value of OFAI when OFAE is low.

LCVE

The line code violation interrupt enable bit (LCVE) masks the contribution of the line code violation event indication bit (LCVI) in the R8TD block to INTB. When LCVE is high, INTB is asserted low when LCVI is high. INTB is not affected by the value of LCVI when LCVE is low.

FUOE

The FIFO underrun/overrun status interrupt enable bit (FUOE) masks the contribution of the FIFO underrun/overrun event indication bit (FUOI) in the R8TD block to INTB. When FUOE is high, INTB is asserted low when FUOI is high. INTB is not affected by the value of FUOI when FUOE is low.

RXINV

The receive data invert bit (RXINV) controls the active polarity of the incoming data stream. When RXINV is set high, the data is complemented before any processing by the R8TD. When RXINV is set low, data is not complemented before R8TD processing.

Reserved[2:0]

The Reserved[2:0] bits must be set to the indicated default value for correct operation of the NSE.

Register 101H + N*20H, R8TD Interrupt Status

Bit	Type	Function	Default
Bit 31:8		Unused	X
Bit 7	R	FUOI	X
Bit 6	R	LCVI	X
Bit 5	R	OFAI	X
Bit 4	R	OCAI	X
Bit 3:0		Unused	X

These registers reports interrupt status due to change of character alignment events and detection of line code violations for the R8TD block.

OCAI

The out-of-character-alignment interrupt status bit (OCAI) reports and acknowledges change of character alignment state events for the R8TD block. OCAI is set high when the character alignment block changes state to the out-of-character-alignment state or to the in-character-alignment state since the last clear for the register. OCAI is cleared on a read to this register when WCIMODE is logic 0. OCAI is cleared on a write (of any value) to this register when WCIMODE is logic one. INTB is asserted low when both OCAE and OCAI are high. If OCAE is asserted, OCAI must be cleared before INTB will be reasserted.

OFAI

The out-of-frame-alignment interrupt status bit (OFAI) reports and acknowledges change of frame alignment state events for the R8TD block. OFAI is set high when the frame alignment block changes state to the out-of-frame-alignment state or to the in-frame-alignment state. OFAI is cleared on a read to this register when WCIMODE is logic 0. OFAI is cleared on a write (of any value) to this register when WCIMODE is logic one. INTB is asserted low when both OFAE and OFAI are high. IF OFAE is asserted, OFAI must be cleared before INTB will be reasserted.

LCVI

The line code violation event interrupt status bit (LCVI) reports and acknowledges line code violation events for the R8TD block. LCVI is set high when the character alignment block detects a line code violation in the incoming data stream. LCVI is cleared on a read to this register when WCIMODE is logic 0. LCVI is cleared on a write (of any value) to this register when WCIMODE is logic one. INTB is asserted low when both LCVE and LCVI are high. IF LCVE is asserted, LCVI must be cleared before INTB will be reasserted.

FUOI

The FIFO underrun/overrun event interrupt status bit (FUOI) reports and acknowledges the FIFO underrun/overrun events for the R8TD block. FUOI is set high when R8TD detects a that the FIFO read and write pointers are within one slot of each other. FUOI is cleared on a read to this register when WCIMODE is logic 0. FUOI is cleared on a write (of any value) to this register when WCIMODE is logic one. INTB is asserted low when both FUOE and FUOI are high. IF FUOE is asserted, FUOI must be cleared before INTB will be reasserted.

Register 102H + N*20H, R8TD Line Code Violation Count

Bit	Type	Function	Default
Bit 31:16		Unused	X
Bit 15:0	R	LCV[15:0]	X

These registers reports the number of line code violations in the previous accumulation period for the R8TD blocks.

LCV[15:0]

The LCV[15:0] bits reports the number of line code violations that have been detected since the last time the LCV registers were polled. The LCV registers are polled by writing to the TIP register or by writing to this register. Within 10 us of either event, the internally accumulated error count is transferred to the LCV registers and the internal error counter is simultaneously reset to begin a new cycle of error accumulation.

Register 103H + N*20H, RXLV and DRU Control

Bit	Type	Function	Default
Bit 31:16	R	Unused	X
Bit 15	R/W	DRU_DTMSB	1
Bit 14	R/W	Reserved	1
Bit 13	R/W	DRU_ENB	0
Bit 12	R/W	RX_ENB	0
Bit 11	R/W	Reserved	0
Bit 10	R/W	A_RSTB	1
Bit 9	R/W	Reserved[4]	0
Bit 8	R/W	Reserved[3]	0
Bit 7	R/W	Reserved[2]	0
Bit 6	R/W	Reserved[1]	0
Bit 5	R/W	DRU_CTRL[3]	0
Bit 4	R/W	DRU_CTRL[2]	0
Bit 3	R/W	DRU_CTRL[1]	0
Bit 2	R/W	DRU_CTRL[0]	0
Bit 1	R/W	Reserved[0]	0
Bit 0		Unused	X

These registers drives the control signals for the RXLV and DRU blocks.

After Chip reset, this Register must be set to CC34h for proper operation, See below.

DRU_CTRL[3:0]

The DRU_CTRL[3:0] bits control the DRU CTRL[3:0] inputs. **The DRU_CTRL[3:0] bus is reset to 0000, but needs to be set to 1101 following a reset for correct operation of the NSE.**

A_RSTB

The A_RSTB bit is a soft-reset for the Data Recovery Unit Analog block. Setting A_RSTB to logic 0 will reset the block.

Reserved

The Reserved bit is set to logic 0 on reset, but needs to be set to logic one following reset for correct operation of the NSE.

RX_ENB

The RXLV enable bit (RX_ENB) bit controls the operation of RXLV block #X. Setting RX_ENB to logic 0 enables the block. Setting RX_ENB to logic one disables the block.

DRU_ENB

The DRU enable bit (DRU_ENB) bit controls the operation of Data Recovery Unit Analog block #X. Setting DRU_ENB to logic 0 enables the block. Setting DRU_ENB to logic one disables the block.

Register 108H + N*20H, T8TE Control and Status

Bit	Type	Function	Default
Bit 31:6		Unused	X
Bit 5	R/W	Reserved[1]	0
Bit 4	R/W	FIFOERRE	0
Bit 3	R/W	TPINS	0
Bit 2	R/W	Reserved[0]	0
Bit 1	W	CENTER	1
Bit 0	R/W	DLCV	0

These registers provide, control and report the status of the T8TE blocks.

Reserved[1:0]

The Reserved bit must be set to the indicated default value for correct operation of the NSE.

DLCV

The diagnose line code violation bit (DLCV) controls the insertion of line code violation in the outgoing data stream. While DLCV is logic one and TCBMODE is logic 0, the transmitted 8B/10B codes are inverted. This will result in at least one disparity error at a receive 8B/10B decoder. When the NSE-8G is configured with TCBMODE logic one, and DLCV logic one, 8B/10B data characters are inverted while the TeleCombus control characters are not inverted. When DLCV is logic 0, no code inversion is performed.

CENTER

The FIFO centering control bit (CENTER) controls the separation of the T8TE FIFO read and write pointers. CENTER is a write only bit. When a logic high is written to CENTER, and the current FIFO depth is not in the range of 3, 4 or 5 characters, the FIFO depth is forced to be four 8B/10B characters deep, with a momentary data corruption. Writing to the CENTER bit when the FIFO depth is in the 3, 4 or 5 character range produces no effect. CENTER always returns a logic low when read.

This bit must be set after CSU lock has been achieved to properly center the FIFO.

TPINS

The Test Pattern Insertion (TPINS) controls the insertion of test pattern in the outgoing data stream for jitter testing purpose. When this bit is set high, TP[9:0] in the T8TE Test Pattern register is selected for output.

FIFOERRE

The FIFO underrun/overrun error interrupt enable bit (FIFOERRE) masks the contribution of the FIFO underrun/overrun event indication bit (FIFOERRI) in the T8TE block to INTB. When FIFOERRE is high, INTB is asserted low when FIFOERRI is high. INTB is not affected by the value of FIFOERRI when FIFOERRE is low.

Register 109H + N*20H, T8TE Interrupt Status

Bit	Type	Function	Default
Bit 31:5		Unused	X
Bit 4	R	FIFOERRI	0
Bit 3:0		Unused	X

These registers report the interrupt status for T8TE blocks #0 through #11.

FIFOERRI

The FIFO overrun/underrun error interrupt indication bit (FIFOERRI) reports a FIFO overrun/underrun error event. FIFOERRI is set high when when FIFO logic detects FIFO read and write pointers in close proximity to each other. FIFOERRI is cleared on a read to this register when WCIMODE is logic 0. FIFOERRI is cleared on a write (of any value) to this register when WCIMODE is logic one. INTB is asserted low when both FIFOERRE and FIFOERRI are high. IF FIFOERRE is asserted, FIFOERRI must be cleared before INTB will be reasserted.

Register 10AH + N*20H: T8TE Time-slot Configuration #1

Bit	Type	Function	Default
Bit 31:16	R	Unused	X
Bit 15	R/W	TMODE8[1]	0
Bit 14	R/W	TMODE8[0]	0
Bit 13	R/W	TMODE7[1]	0
Bit 12	R/W	TMODE7[0]	0
Bit 11	R/W	TMODE6[1]	0
Bit 10	R/W	TMODE6[0]	0
Bit 9	R/W	TMODE5[1]	0
Bit 8	R/W	TMODE5[0]	0
Bit 7	R/W	TMODE4[1]	0
Bit 6	R/W	TMODE4[0]	0
Bit 5	R/W	TMODE3[1]	0
Bit 4	R/W	TMODE3[0]	0
Bit 3	R/W	TMODE2[1]	0
Bit 2	R/W	TMODE2[0]	0
Bit 1	R/W	TMODE1[1]	0
Bit 0	R/W	TMODE1[0]	0

Register 02H configures the path termination mode of time-slots 1 to 8 of the T8TE.

TMODE1[1:0]-TMODE8[1:0]

The time-slot path termination mode select register bits (TMODE1[1:0]-TMODE8[1:0]) configures the mode settings for time-slots 1 to 8 of the T8TE. Time-slots are numbered in order of transmission in the Incoming TeleCombus stream (ID[7:0]). Time-slot #1 is the first byte transmitted and time-slot #12 is the last byte transmitted. The setting stored in TMODEx[1:0] (x can be 1-12) determines which set of TeleCombus control signals are to be encoded in 8B/10B characters.

In TeleCombus mode, the T8TE encodes TeleCombus control signals such as transport frame and payload boundaries, pointer justification events and alarm conditions into three levels of an extended set of 8B/10B characters as well as performing the IEEE mode conversion on data. The two hierarchical levels are High-order Path Termination (HPT) and Low-order Path Termination (LPT). For correct operation see table below:

TMODEx[1]	TMODEx[0]	Mode	Functional Description
0	0	Reserved	invalid
0	1	HPT level	Use for TelecomBus where V1/V2 pointers must be preserved
1	0	LPT level	Use for SBI336S or TelecomBus with valid V5 but without valid V1/V2 pointers
1	1	Reserved	invalid

Register 10BH + N*20H: T8TE Time-slot Configuration #2

Bit	Type	Function	Default
Bit 31:8		Unused	X
Bit 7	R/W	TMODE12[1]	0
Bit 6	R/W	TMODE12[0]	0
Bit 5	R/W	TMODE11[1]	0
Bit 4	R/W	TMODE11[0]	0
Bit 3	R/W	TMODE10[1]	0
Bit 2	R/W	TMODE10[0]	0
Bit 1	R/W	TMODE9[1]	0
Bit 0	R/W	TMODE9[0]	0

Register 03H configures the path termination mode of time-slots 9 to 12 of the T8TE.

TMODE9[1:0]-TMODE12[1:0]

The time-slot path termination mode select register bits (TMODE9[1:0]-TMODE12[1:0]) configures the mode settings for time-slots 9 to 12 of the T8TE. Time-slots are numbered in order of transmission in the Incoming TeleCombus stream (ID[7:0]). Time-slot #1 is the first byte transmitted and time-slot #12 is the last byte transmitted. The setting stored in TMODEx[1:0] (x can be 1-12) determines which set of TeleCombus control signals are to be encoded in 8B/10B characters.

In TeleCombus mode, the T8TE encodes TeleCombus control signals such as transport frame and payload boundaries, pointer justification events and alarm conditions into three levels of an extended set of 8B/10B characters as well as performing the IEEE mode conversion on data. The two hierarchical levels are High-order Path Termination (HPT) and Low-order Path Termination (LPT). For correct operation see the table below:

TMODEx[1]	TMODEx[0]	Mode	Functional Description
0	0	Reserved	invalid
0	1	HPT level	Use for TelecomBus where V1/V2 pointers must be preserved
1	0	LPT level	Use for SBI336S or TelecomBus with valid V5 but without valid V1/V2 pointers
1	1	Reserved	invalid

Register 10CH + N*20H, T8TE Test Pattern

Bit	Type	Function	Default
Bit 31:10		Unused	X
Bit 9	R/W	TP[9]	1
Bit 8	R/W	TP[8]	0
Bit 7	R/W	TP[7]	1
Bit 6	R/W	TP[6]	0
Bit 5	R/W	TP[5]	1
Bit 4	R/W	TP[4]	0
Bit 3	R/W	TP[3]	1
Bit 2	R/W	TP[2]	0
Bit 1	R/W	TP[1]	1
Bit 0	R/W	TP[0]	0

These registers store the test pattern for test pattern insertion for the T8TE blocks.

TP[9:0]

The Test Pattern register (TP[9:0]) for T8TE block #X contains the test pattern conditionally inserted into output data stream #X. TP[9:0] is inserted into the output data stream when the TPINS bit is set high.

Register 10DH + N*20H, TXLV and PISO Control

Bit	Type	Function	Default
Bit 31:12		Unused	X
Bit 11		Reserved[8]	0
Bit 10	R/W	Reserved[7]	0
Bit 9	R/W	Reserved[6]	0
Bit 8	R/W	TXLV_ENB	0
Bit 7	R/W	PISO_ENB	0
Bit 6	R/W	Reserved[5]	0
Bit 5	R/W	Reserved[4]	0
Bit 4	R/W	Reserved[3]	0
Bit 3	R/W	Reserved[2]	0
Bit 2	R/W	Reserved[1]	1
Bit 1	R/W	Reserved[0]	1
Bit 0	R/W	ARSTB	1

ARSTB

The analog reset bit (ARSTB) resets the associated TXLV and PISO blocks. When ARSTB is set to logic 0, the TXLV and PISO are reset.

Reserved[5:0]

The Reserved[3:0] bits must be set to the indicated default value for correct operation of the NSE.

PISO_ENB

The PISO enable bit (PISO_ENB) controls the operation of the PISO block. PISO_ENB is set to logic one to disable the PISO block. PISO_ENB is set to logic 0 to enable the PISO block.

TXLV_ENB

The TXLV enable bit (TXLV_ENB) controls the operation of TXLV block. TXLV_ENB is set to logic one to disable the TXLV block. TXLV_ENB is set to logic 0 to enable the TXLV block.

Reserved[8]

The Reserved[8] bit must be set to the indicated default value for correct operation of the NSE.

Register 110H + N*20H, ILC Transmit FIFO Data

Bit	Type	Function	Default
Bit 31-0	R/W	TDAT[31:0]	0

TDAT[31:0]

TDAT[31: 0]Transmit FIFO form the 32 bit wide data word to be written to the register file FIFO. A single 32 bit write to this register will update TDAT[31:0]. A write to this address initiates a FIFO write sequence.

Register 111h + N*20H, ILC Transmit Control Register

Bit	Type	Function	Default
Bit 31:16		Unused	X
Bit 15:8	R/W	TX_AUX[7:0]	00000000
Bit 7:6	R	Reserved	00
Bit 5:4	R/W	TX_LINK[1:0]	00
Bit 3:2	R	Reserved	00
Bit 1	R/W	TX_CRC_SWIZ_EN	0
Bit 0	R/W	TX_BYPASS	0

TX_BYPASS

When this bit is set to '1', the blocks message transmit functions are bypassed. No messages are inserted into the Transmit Bus data The respective signals are passed through the block's pipeline unmodified. Transmit message FIFO RAM is disabled and thus message data writes are ignored.

TX_CRC_SWIZ_EN

When this bit is set to '1', the calculated CRC-16 is bit reversed before being transmitted. This facility can be used for diagnostic testing of CRC-16 generation and checking functionality.

TX_LINK[1:0]

These bits are transmitted in the LINK bits of the message header of the next available message. On reads these bit return the last written value.

TX_AUX[7:0]

These bits form the input to an Auxiliary channel between CPUs at each end of the link. Their use is at the Software developers' discretion. Data written to this register will be transmitted in the AUX header byte of each subsequent message to the other end of the inband link. A new value of TX_AUX will be transmitted at the next available message.

Data read from this register will be the data previously written.

Register 112h + N*20H, ILC Transmit Misc.Status and FIFO Synch Register

Bit	Type	Function	Default
Bit 31:16		Unused	X
Bit 15	R	TX_MSG_LVL_VALID	N/A
Bit 14:13	R	TX_LINK[1:0]	00
Bit 12:11	R	IPAGE[1:0]	N/A
Bit 10:8	R	IUSER[2:0]	N/A
Bit 7:6	R	Reserved	00
Bit 5:2	R	TX_MSG_LVL[3:0]	0000
Bit 1	R	TX_FI_BUSY	0
Bit 0	W	TX_XFER_SYNC	0

This register serves a dual purpose dependant on whether it is being read or written.

When it is read it returns the status for the Message Transmit Channel.

When it is written (with 0001h) to it is used it synchronize the Transmit FIFO to the start of a message boundary.

TX_XFER_SYNC

Writing ‘1’ to this bit initializes the next write sequence to be to the beginning of the next message. After a ‘1’ had been written successive writes to the Transmit FIFO will be to location zero of the next available slot. If a partial message has been written, TX_XFER_SYNC indicates that the current message is complete and that subsequent writes will be to the next message. If more than 32 bytes are written, the 33rd byte will be the first byte of the next message. The purpose of this bit is to unambiguously align the message boundaries. Another use would be to abandon the current write and move the write pointer to the beginning of the next message. (Previous message data will remain in the unwritten portion of the message being abandoned, which will have to be ignored by the receiving software).

If the message FIFO pointers are already at a message boundary then writing this bit to a ‘1’ will have no affect.

On reads this bit is always returned as a ‘0’.

TX_FI_BUSY

This bit indicates that the internal hardware is transferring the data from the Transmit FIFO registers (TDAT) into the internal RAM. This bit need not be read by software if the time interval between successive 32 bit transfers is greater than 3 SYSCLK cycles.

User and Page bits are a copy of the User bits received, and being transmitted in 0Ch. These allow one read in the 32 bit device to gain a snapshot of the entire ILC.

TX_MSG_LVL[3:0]

This indicates the current number of messages in the TXFIFO.

Table 7 TX FIFO Message Level

TX_MSG_LVL[3:0]	Number of messages
0000	0
:	:
1000	8

Values greater than 1000 will not occur. The number of free messages available in the FIFO is given by (8 – TX_MSG_LVL).

IUSER

This bits are a reflection of the IUSER[2:0]. These reflect the USER bits that will be transmitted in subsequent message headers. There is no default value for these bits as they are device dependant.

IPAGE

This bits are a reflection of the IPAGE[1:0. These reflect the PAGE bits that will be transmitted in subsequent message headers. There is no default value for these bits as they are device dependant.

TX_LINK[1:0]

These bits reflect the last written value of the TX_LINK field of the TX Control register. The upper byte of this register therefore reflects all of the configurable bits of the message Header1 byte.

TX_MSG_LVL_VALID

This bit indicates that the value of TX_MSG_LVL is valid. When read with a ‘0’ this register should be re-read until TX_MSG_LVL_VALID is a ‘1’. This bit will be clear for only approximately 0.3% of time.

Register 113h + N*20H, ILC Receive FIFO Data Register

Bit	Type	Function	Default
Bit 31:0	R	RDAT[31:0]	00000000h

RDAT[31:0]

RDAT[31: 0] is the 32 bit wide data word read from the FIFO.

A single read from this register will update RDAT[31:0]. A read from this address initiates a FIFO read sequence.

Register 114h + N*20H, ILC Receive Control Register

Bit	Type	Function	Default
Bit 31:16		Unused	X
Bit 15:3	R	Reserved[15:3]	00000000000000
Bit 2	R/W	Reserved[2]	0
Bit 1	R/W	RX_CRC_SWIZ_EN	0
Bit 0	R/W	RX_BYPASS	0

RX_BYPASS

When this bit is set to '1', the blocks message receive functions are bypassed. No messages are extracted from the Receive TelecomBus. The RXTPL , RXPL and RXDATA signals are passed through the blocks pipeline unmodified. Receive message FIFO RAM is disabled and thus message data reads return undefined data.

RX_CRC_SWIZ_EN

When this bit is set to '1', the calculated CRC-16 is bit reversed before being compared with CRC-16 bytes of the received message. This facility can be used for diagnostic testing of CRC-16 generation and checking functionality.

Reserved[15:3]

Reserved for future use.

Reserved[2]

Default Value is '0', but should be set to '1' for correct operation.

Register 115h + N*20H, ILC Receive Auxiliary, Status and FIFO Synch Register

Bit	Type	Function	Default
Bit 31:24	R	Reserved	00000000
Bit 23:16	R	RX_AUX[7:0]	00h
Bit 15	R	RX_STTS_VALID	N/A
Bit 14:13	R	RX_LINK[1:0]	00
Bit 12:11	R	RX_PAGE[1:0]	00
Bit 10:8	R	RX_USER[2:0]	000
Bit 7	R	CRC_ERR	0
Bit 6	R	HDR_CRC_ERR	0
Bit 5:2	R	RX_MSG_LVL[3:0]	0000
Bit 1	R	RX_FI_BUSY	0
Bit 0	W	RX_XFER_SYNC	N/A
Bit 0	R	RX_SYNC_DONE	0

This register serves a dual-purpose dependant on whether it is being read or written.

When it is read it returns the status for the Message Receive Channel.

When it is written (with 00000001h) to it is used it synchronize the Receive FIFO to the start of a message boundary or perform a message skip.

RX_AUX[7:0]

These bits constitute the output from an Auxiliary channel between CPUs at each end of the link. Their use is at the Software developers' discretion. A read from this register will return the AUX header byte of the last message received (without a CRC-16 error).

RX_XFER_SYNC

Rx Transfer sync writing Writing '1' to this bit initiates a read sequence from the start of the next *unread* message. The hardware

- Aligns the message read buffer address to the start of the next *unread* message
- Prefetches the 1st Dword from the *unread* message buffer so that it is ready for a s/w read from the Receive FIFO Data register(s)

An *unread* message in this context means that the s/w has not read any of the message payload data by reading the Receive FIFO Data register(s)

After the RX XFER SYNC process has been completed successive reads from the Receive FIFO return the last Dword read from the Receive FIFO and prefetch the next Dword (when available).

This bit must be written to a '1' at the start of a message read sequence.

When multiple complete messages are being read (software knows that there is more than one message in the FIFO using the RX_MSG_LVL bits) this bit does not need to be written between individual message reads. It must be written for the 1st message.

When software uses a variable length message protocol it may want to abandon reading a message buffer before reading the entire message buffer of 8 DWords (16 Words). In this case this bit must be written with a '1' to move the message pointer to the start of the next message buffer before starting the read of that buffer.

After writing this bit with a '1' software should not start reading the FIFO until the RX_FI_BUSY bit has cleared.

In the worst case this will take 5 SYSCLK cycles when FAST_RD_EN = '1' and 4 SYSCLK cycles when FAST_RD_EN = '0'.

At this point the 1st DWORD of the message is available for reading and the CRC_ERR bit is valid. Software may abandon a CRC errored message without reading the message buffer by writing this bit with a '1' again.

On reads this bit is always returns the RX_SYNC_DONE status.

A suggested s/w procedure for accessing the Receive Message Buffer is outlined in section 12.12.2 (Accessing the Receive Message FIFO)

RX_SYNC_DONE

This bit indicates the status of an RX_XFER_SYNC operation. When '1' it indicates that an RX_XFER_SYNC has been done. S/W should check this bit at the start of a message read sequence or when attempting to perform a message skip sequence.

RX_FI_BUSY

This bit indicates that the internal hardware is transferring data from the Receive FIFO RAM into the Receive FIFO registers. The bits is set following

- A write to the Receive FIFO Synch Register (Bh) with the RX_XFER_SYNC bit set.
- A read from the Receive FIFO Data Low register.

Following an RX_XFER_SYNC write this bit need not be read by software if the time interval to the successive Receive FIFO DATA register read is greater than approximately 5 SYSCLK cycles when FAST_RD_EN = '1' or approximately 4 SYSCLK cycles when FAST_RD_EN = '0'.

This bit need not be read by software if the time interval between successive Receive FIFO DATA register reads greater than approximately 4 SYSCLK cycles when FAST_RD_EN = '1' or approximately 3 SYSCLK cycles when FAST_RD_EN = '0'.

RX_MSG_LVL[3:0]

This indicates the current number of messages in the Receive FIFO.

Table 8 RX FIFO Message Level

RX_MSG_LVL[3:0]	Number of messages
0000	0
:	:
1000	8

Values greater than 1000 will not occur.

HDR_CRC_ERR

If this bit is set to '1', the last message slot received was received with an errored CRC-16 field. This bits is updated every message slot. This bit is provided as status only.

CRC_ERR

If this bit it set to '1', the message at the head of the Receive FIFO has an errored CRC-16 field.

The usual sequence would be to read this register before reading the message buffer to check if the message buffer that will be read from next has been received with a CRC error. If a Receive FIFO Synchronization has been started the value of this bit is invalid until the RX_XFER_SYNC operation has completed. When FAST_RD_EN is a '1' this bit is valid when RX_FI_BUSY is a '0' following a Receive FIFO Synchronization. When FAST_RD_EN is a '0' the values of RX_FI_BUSY and CRC_ERR change concurrently and a further read should be made after RX_FI_BUSY is sampled as a '0' before checking the value of this bit.

OUSER[2:0]

These bits are a reflection of the OUSER[2:0] bits output from the far end and indicate the value of the received USER bits in the received message header of the last message received (without a CRC-16 error). These bits are available in the three top level SBS User bits registers at a bit position equal to the link number.

OPAGE[1:0]

These bits are a reflection of the OPAGE[1:0] bits output from the far end and indicate the value of the received PAGE bits in the received message header of the last message received (without a CRC-16 error). These bits are available in the two top level SBS page bits registers at a bit position equal to the link number.

RX_LINK[1:0]

These bits indicate the value of the LINK bits from the message header of the last message received (without a CRC-16 error).

RX_STTS_VALID

This bit indicates that the values of RX_MSG_LVL , RX_LINK, OPAGE, OUSER are valid. In 32 bit mode this bit also indicates whether the value of RX_AUX (returned in bits 16 to 23) is valid. When read with a ‘0’ this register should be re-read until RX_STTS_VALID is a ‘1’. This bit will be set for only approximately 0.15% of time.

Register 116h + N*20H, ILC Interrupt Enable and Control Register

Bit	Type	Function	Default
Bit 31:16		Unused	X
Bit 15:13	R	Reserved	000
Bit 12:11	R/W	RX_TIMEOUT_VAL[1:0]	00
Bit 10:8	R/W	RX_THRESHOLD_VAL[2:0]	101
Bit 7	R	Reserved	0
Bit 6	R/W	RX_TIMEOUTE	0
Bit 5	R/W	RX_THRSHLDE	0
Bit 4	R/W	RX_OVFLWE	0
Bit 3	R/W	RX_LINK_CHGE	0
Bit 2:1	R/W	RX_PAGE_CHGE[1:0]	0
Bit 0	R/W	RX_USER0_CHGE	0

RX_USER0_CHGE

Writing a ‘1’ to the RX_OUSER0_CHGE bit enables the generation of an interrupt on a change of state from a ‘0’ to a ‘1’ of received message header bit RX_USER[0].

RX_PAGE_CHGE[1:0]

Writing a ‘1’ to the RX_PAGE_CHGE[n] bit enables the generation of an interrupt on a change of state of the received PAGE bits. The RX_PAGE bits that changed value are indicated by a ‘1’ in the corresponding RX_PAGE_CHGI[n].

RX_LINK_CHGE

Writing a ‘1’ to the RX_LINK_CHGE bit enables the generation of an interrupt on a change of state of the received LINK bits. When either of the received LINK bits has changed value the RX_LINK_CHGI bit will be set to a ‘1’.

If the RXFIFO level had reached the threshold value an interrupt will be generated if this bit is ‘1’. To disable set to ‘0’.

RX_OVFLWE

Writing a ‘1’ to the RX_OVFLWE bit enables the generation of an interrupt when RX_OVFLWI is a ‘1’.

RX_THRSHLDE

Writing a ‘1’ to the RX_THRSHLDE bit enables the generation of an interrupt when RX_THRSHLDI is a ‘1’

RX_TIMEOUTE

Writing a ‘1’ to the RX_TIMEOUTE bit enables the generation of an interrupt when RX_TIMEOUTI is a ‘1’

RX_THRSHLD_VAL[2:0]

Variable Threshold dictates the minimum number of messages required to be in the RXFIFO before an interrupt is generated. ‘000’ = 1 message ‘111’ = 8 messages.

Table 9 RXFIFO Threshold Values

RX_THRSHLD_VAL [2:0]	Messages
000	1
001	2
010	3
011	4
100	5
101	6
110	7
111	8

If the RXFIFO has not been read for the amount of time indicated by the RXFIFO_Timeout_delay AND the FIFO is not empty an interrupt will be generated if this bit is ‘1’. To disable set to ‘0’.

If the RXFIFO overflows an interrupt will be generated if this bit is ‘1’. To disable set to ‘0’.

If the Rx Message fails its CRC check, an interrupt will be generated if this bit is ‘1’. To disable set to ‘0’.

RX_TIMEOUT_VAL

These bits specify a variable delay, relative to a read from the receive message FIFO, in steps of 125 us, before an interrupt is generated, if the Receive FIFO level is greater than 0. The objective is to stop stale messages collecting in the RXFIFO.

Table 10 RXFIFO Timeout Delay

RX_TIMEOUT_VAL[1:0]	Nominal Delay In Frames	Minimum Delay from message reception	Maximum Delay from message reception	Minimum Delay from FIFO read	Maximum Delay from FIFO read
00	1	152us	222 us	125us	250 us
01	2	277us	347 us	250us	375us
10	3	402us	472 us	375us	500us
11	4	527us	597 us	500us	625 us

Register 117h + N*20H: ILC Interrupt Reason Register

Bit	Type	Function	Default
Bit 31:0		Unused	X
Bit 15:7	R	Reserved	000000000
Bit 6	R	RX_TIMEOUTI	0
Bit 5	R	RX_THRSHLDI	0
Bit 4	R	RX_OVFLWI	0
Bit 3	R	RX_LINK_CHGI	0
Bit 2:1	R	RX_PAGE_CHGI[1:0]	0
Bit 0	R	RX_OUSER0_CHGI	0

This register contains the status of events that may be enabled to generate interrupts.

All bits in this register are cleared on read

RX_OUSER0_CHGI

A ‘1’ in this bit indicates that the last received value of the RX_USER[0] header bit has changed from a ‘0’ to a ‘1’ from the previously received values. This bit is cleared on a read.

RX_PAGE_CHGI [1:0]

A ‘1’ in these bits indicates that the last received value of the corresponding RX_PAGE[1:0] header bits has changed from the previously received values. These bits are cleared on read.

RX_LINK_CHGI

A ‘1’ in this bit indicates that the last received value of the LINK[1:0] header bits has changed from the previously received values. This bit is cleared on a read.

RX_OVFLWI

This bit, when ‘1’, indicates a Receive FIFO Overflow. This bit is cleared on a read.

RX_THRSHLDI

This bit, when ‘1’, indicates a Receive FIFO Threshold reached. This bit is cleared on a read.

RX_TIMEOUTI

This bit, when ‘1’, indicates a Receive FIFO Timeout. This bit is cleared on read.

11 Test Features Description

The test mode registers, shown in Table 11, are used for production and board testing.

During production testing, the test mode registers are used to apply test vectors. In this case, the test mode registers (as opposed to the normal mode registers) are selected when A[10] is high.

During board testing, the digital output pins and the data bus are held in a high-impedance state by simultaneously asserting (low) the CSB, RDB, and WRB inputs. All of the TSBs for the NSE-8G are placed in test mode 0 so that device inputs may be read and device outputs may be forced through the microprocessor interface.

Note: The NSE-8G supports a standard IEEE 1149.1 five-signal JTAG boundary scan test port that can be used for board testing. All digital device inputs may be read and all digital device outputs may be forced through this JTAG test port.

Table 11 Test Mode Register Memory Map

Address	Register
000H-7FFH	Normal Mode Registers
800H	Master Test Register
801H – FFFH	Reserved For Test

11.1 Master Test and Test Configuration Registers

Notes on Test Mode Register Bits

1. Writing values into unused register bits has no effect. However, to ensure software compatibility with future, feature-enhanced versions of the product, unused register bits must be written with logic zero. Reading back unused bits can produce either a logic one or a logic zero; hence, unused register bits should be masked off by software when read.

2. Writeable test mode register bits are not initialized upon reset unless otherwise noted.

Register 800H: NSE-8G Master Test

Bit	Type	Function	Default
Bit 31:6	R	Unused	X
Bit 5	W	PMCATST	X
Bit 4	W	PMCTST	X
Bit 3	W	DBCTRL	0
Bit 2	R/W	IOTST	0
Bit 1	W	HIZDATA	0
Bit 0	R/W	HIZIO	0

This register is used to enable NSE-8G test features. All bits, except PMCTST, and PMCATST are reset to zero by a reset of the NSE-8G using the RSTB input. PMCTST is reset when CSB is high; PMCATST is reset when CSB is high and RSTB is low. PMCTST and PMCATST can also be reset by writing a logic 0 to the corresponding register bit.

Access to this register is not affected by the Test Mode Address Force functions in registers 1001H and 1002H.

HIZIO, HIZDATA

The HIZIO and HIZDATA bits control the tri-state modes of the NSE-8G . While the HIZIO bit is a logic one, all output pins of the NSE-8G except the data bus and output TDO are held tri-state. The microprocessor interface is still active. While the HIZDATA bit is a logic one, the data bus is also held in a high-impedance state which inhibits microprocessor read cycles. The HIZDATA bit is overridden by the DBCTRL bit.

IOTST

The IOTST bit is used to allow normal microprocessor access to the test registers and control the test mode in each TSB block in the NSE-8G for board level testing. When IOTST is a logic one, all blocks are held in test mode and the microprocessor may write to a block's test mode 0 registers to manipulate the outputs of the block and consequently the device outputs (refer to the "Test Mode 0 Details" in the "Test Features" section).

DBCTRL

The DBCTRL bit is used to pass control of the data bus drivers to the CSB pin. When the DBCTRL bit is set to logic one and PMCTST is set to logic one, the CSB pin controls the output enable for the data bus. While the DBCTRL bit is set, holding the CSB pin low causes the NSE-8G to drive the data bus and holding the CSB pin high tri-states the data bus. The DBCTRL bit overrides the HIZDATA bit. The DBCTRL bit is used to measure the drive capability of the data bus driver pads.

PMCTST

The PMCTST bit is used to configure the NSE-8G for PMC's manufacturing tests. When PMCTST is set to logic one, the NSE-8G microprocessor port becomes the test access port used to run the PMC "canned" manufacturing test vectors. As well, the analog blocks are placed in IDDQ mode = the digital circuitry within the analog blocks is held static The PMCTST can be cleared by setting CSB to logic one or by writing logic zero to the bit.

PMCATST

The PMCATST bit is used to configure the analog portion of the NSE-8G for PMC's manufacturing tests. The PMCATST bit can be cleared by setting both CSB to logic one and RSTB to logic zero. PMCATST can also be cleared by writing logic zero to the bit.

11.2 JTAG Test Port

The NSE JTAG Test Access Port (TAP) allows access to the TAP controller and the four TAP registers: instruction, bypass, device identification and boundary scan. Using the TAP, device input logic levels can be read, device outputs can be forced, the device can be identified and the device scan path can be bypassed. For more details on the JTAG port, please refer to the Operations section.

Table 12 Instruction Register (Length - 3 bits)

Instructions	Selected Register	Instruction Codes, IR[2:0]
EXTEST	Boundary Scan	000
IDCODE	Identification	001
SAMPLE	Boundary Scan	010
BYPASS	Bypass	011
BYPASS	Bypass	100
STCTEST	Boundary Scan	101
BYPASS	Bypass	110
BYPASS	Bypass	111

Table 13 Identification Register

Length	32 bits
Version Number	0H
Part Number – NSE-20G	8620H
Part Number – NSE–8G	8621H
Manufacturer's Identification Code	0CDH
Device Identification – NSE-20G	086200CDH
Device Identification – NSE-8G	086210CDH

Table 14 Boundary Scan Register

Pin/ Enable	Register Bit	Cell Type	I.D. Bit
OEB_INTB	121	OUT_CELL	0
INTB	120	OUT_CELL	0
ALE	119	IN_CELL	0
CSB	118	IN_CELL	0
WRB	117	IN_CELL	1
RDB	116	IN_CELL	0
A[11]	115	IN_CELL	0
A[10]	114	IN_CELL	0
A[9]	113	IN_CELL	0
A[8]	112	IN_CELL	1
A[7]	111	IN_CELL	1
A[6]	110	IN_CELL	0
A[5]	109	IN_CELL	0
A[4]	108	IN_CELL	0
A[3]	107	IN_CELL	1
A[2]	106	IN_CELL	0
A[1]	105	IN_CELL	0
A[0]	104	IN_CELL	0
OEB_D[31]	103	OUT_CELL	0
D[31]	102	IO_CELL	0
OEB_D[30]	101	OUT_CELL	0
D[30]	100	IO_CELL	0
OEB_D[29]	99	OUT_CELL	0
D[29]	98	IO_CELL	0
OEB_D[28]	97	OUT_CELL	1
D[28]	96	IO_CELL	1
OEB_D[27]	95	OUT_CELL	0
D[27]	94	IO_CELL	0
OEB_D[26]	93	OUT_CELL	1
D[26]	92	IO_CELL	1
OEB_D[25]	91	OUT_CELL	0
D[25]	90	IO_CELL	1
OEB_D[24]	89	OUT_CELL	-
D[24]	88	IO_CELL	-
OEB_D[23]	87	OUT_CELL	-
D[23]	86	IO_CELL	-
OEB_D[22]	85	OUT_CELL	-
D[22]	84	IO_CELL	-

Pin/ Enable	Register Bit	Cell Type	I.D. Bit
OEB_D[21]	83	OUT_CELL	-
D[21]	82	IO_CELL	-
OEB_D[20]	81	OUT_CELL	-
D[20]	80	IO_CELL	-
OEB_D[19]	79	OUT_CELL	-
D[19]	78	IO_CELL	-
OEB_D[18]	77	OUT_CELL	-
D[18]	76	IO_CELL	-
OEB_D[17]	75	OUT_CELL	-
D[17]	74	IO_CELL	-
OEB_D[16]	73	OUT_CELL	-
D[16]	72	IO_CELL	-
OEB_D[15]	71	OUT_CELL	-
D[15]	70	IO_CELL	-
OEB_D[14]	69	OUT_CELL	-
D[14]	68	IO_CELL	-
OEB_D[13]	67	OUT_CELL	-
D[13]	66	IO_CELL	-
OEB_D[12]	65	OUT_CELL	-
D[12]	64	IO_CELL	-
OEB_D[11]	63	OUT_CELL	-
D[11]	62	IO_CELL	-
OEB_D[10]	61	OUT_CELL	-
D[10]	60	IO_CELL	-
OEB_D[9]	59	OUT_CELL	-
D[9]	58	IO_CELL	-
OEB_D[8]	57	OUT_CELL	-
D[8]	56	IO_CELL	-
OEB_D[7]	55	OUT_CELL	-
D[7]	54	IO_CELL	-
OEB_D[6]	53	OUT_CELL	-
D[6]	52	IO_CELL	-
OEB_D[5]	51	OUT_CELL	-
D[5]	50	IO_CELL	-
OEB_D[4]	49	OUT_CELL	-
D[4]	48	IO_CELL	-
OEB_D[3]	47	OUT_CELL	-
D[3]	46	IO_CELL	-
OEB_D[2]	45	OUT_CELL	-
D[2]	44	IO_CELL	-

Pin/ Enable	Register Bit	Cell Type	I.D. Bit
OEB_D[1]	43	OUT_CELL	-
D[1]	42	IO_CELL	-
OEB_D[0]	41	OUT_CELL	-
D[0]	40	IO_CELL	-
Logic one	39	IN_CELL	-
Logic one	38	IN_CELL	-
Logic one	37	IN_CELL	-
Logic one	36	IN_CELL	-
Logic one	35	IN_CELL	-
Logic one	34	IN_CELL	-
Logic one	33	IN_CELL	-
Logic one	32	IN_CELL	-
Logic one	31	IN_CELL	-
Logic one	30	IN_CELL	-
Logic one	29	IN_CELL	-
Logic one	28	IN_CELL	-
Logic one	27	IN_CELL	-
Logic one	26	IN_CELL	-
CMP	25	IN_CELL	-
SYSCLK	24	IN_CELL	-
RC1FP	23	IN_CELL	-
OEB_TC1FP	22	OUT_CELL	-
TC1FP	21	OUT_CELL	-
RSTB	20	IN_CELL	-
Logic one	19	IN_CELL	-
Logic one	18	IN_CELL	-
Logic one	17	IN_CELL	-
Logic one	16	IN_CELL	-
Logic one	15	IN_CELL	-
Logic one	14	IN_CELL	-
Logic one	13	IN_CELL	-
Logic one	12	IN_CELL	-
Logic one	11	IN_CELL	-
Logic one	10	IN_CELL	-
Logic one	9	IN_CELL	-
Logic one	8	IN_CELL	-
Logic one	7	IN_CELL	-
Logic one	6	IN_CELL	-
Logic one	5	IN_CELL	-
Logic one	4	IN_CELL	-

Pin/ Enable	Register Bit	Cell Type	I.D. Bit
Logic one	3	IN_CELL	-
Logic one	2	IN_CELL	-
Logic one	1	IN_CELL	-
Logic one	0	IN_CELL	-

Notes

1. When set high, INTB will be set to high impedance.
2. Enable cell OEB_*pinname*, tristates pin *pinname* when set high.
3. OEB_INTB is the first bit of the boundary scan chain.
4. Cells 'Logic one' are Input Observation cells whose input pad is bonded to VDD internally.

11.2.1 Boundary Scan Cells

In the following diagrams, CLOCK-DR is equal to TCK when the current controller state is SHIFT-DR or CAPTURE-DR, and unchanging otherwise. The multiplexer in the center of the diagram selects one of four inputs, depending on the status of select lines G1 and G2. The ID Code bit is as listed in the Boundary Scan Register, Table 14.

Figure 13 Input Observation Cell (IN_CELL)

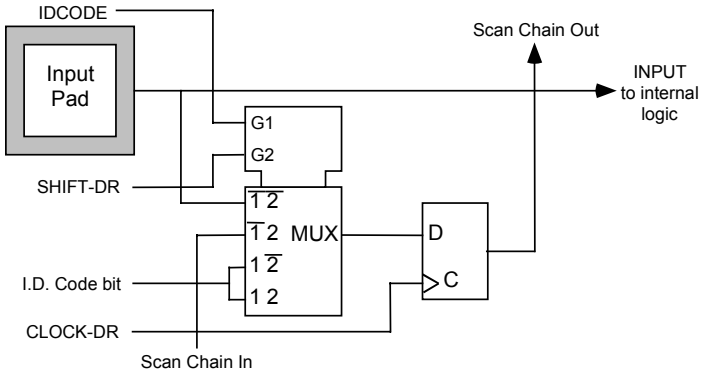


Figure 14 Output Cell (OUT_CELL)

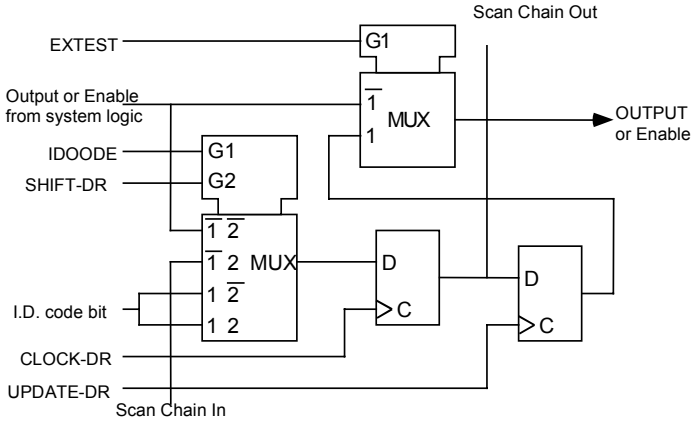


Figure 15 Bidirectional Cell (IO_CELL)

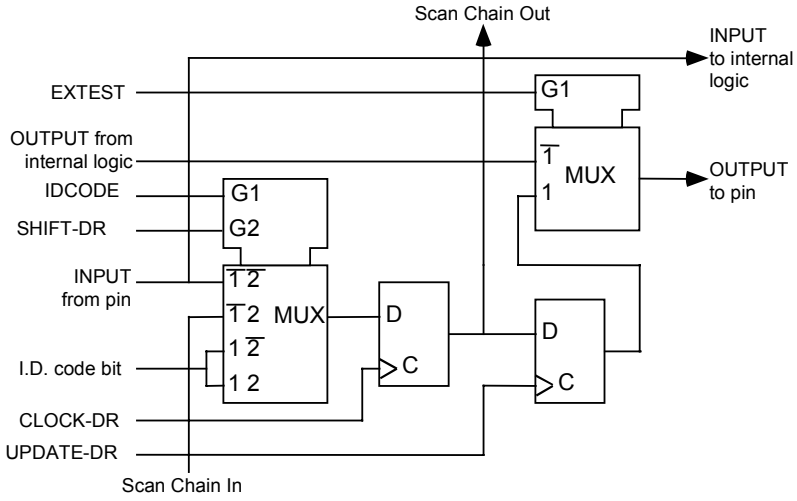
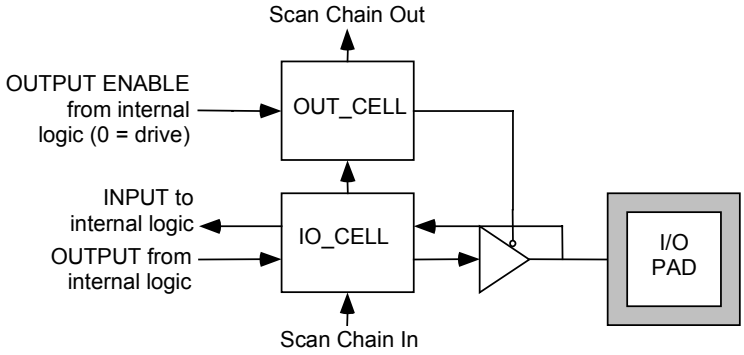


Figure 16 Layout of Output Enable and Bidirectional Cells



12 Operation

There are several important aspects regarding the operation of NSE-based switch fabrics; these are dealt with in turn in the following sections.

12.1 Software Default Settings

12.1.1 Setting the T8TE Time-slot Configuration #1 Register

Set the first T8TE Time-slot Configuration register to 0000AAAAh. This sets T8TE to LPT mode so that Low Order Path signals are encoded in outgoing 8B/10B characters. Or, set to 00005555h to set T8TE to HPT mode to ensure V1/V2 bytes are preserved.

12.1.2 Setting the T8TE Time-slot Configuration #2 Register

Set the second T8TE Time-slot Configuration register to 000000Aah. This sets T8TE to LPT mode so that Low Order Path signals are encoded in outgoing 8B/10B characters. Or, set to 00000055h to set T8TE to HPT mode to ensure V1/V2 bytes are preserved.

12.1.3 Configuring the NSE-8G to Use Fewer Links

The NSE-8G powers up with the software digital reset disabled, software analog reset disabled and individual link reset enabled. This means that only the digital blocks are enabled post hardware reset (since setting channel reset also disable the associated analog blocks). The CSU by default will be start upon NSE-8G powers up; it can only be reset by the firmware writing logic one to the ARESET bit in NSE-8G Master Reset register (000H). By writing logic zero to appropriate channels in NSE-8G Individual Channel Reset register (001H) will bring the associated link out of reset and operational for normal mode operation.

When fewer than 32/12 links are used in the NSE-8G 20G/8G, the unused links should be disabled individually by writing logic one to the appropriate NSE-8G Individual Channel Reset register (001H) bit. Writing logic one to bit N of NSE-8G Individual Channel Reset register will disable the R8TD, ILC, and T8TE of channel N. This reset controls both the digital as well as the analog reset inputs of the R8TD and T8TE. The analog reset input of R8TD and T8TE gates the analog reset and enable output that is used to disable the associated DRU/RXLV, PISO/TXLV analog blocks. This will cause the entire link from input N to output N to be disabled.

Reset States of Various Operation Modes

Post Hardware Reset:

Register 000H : DRESET ‘0’

Register 000H : ARESET ‘0’

Register 001H : RESET ‘0xFFFFFFFF’

Digital Test mode:

Register 000H : DRESET ‘0’

Register 000H : ARESET ‘0’

Register 001H : RESET ‘0x00000000’

Analog Test mode:

Register 000H : DRESET ‘0’

Register 000H : ARESET ‘1’

Register 001H : RESET ‘0xFFFFFFFF’

Holding pmcatstb will overide channel resets to analog blocks during analog tests. Software analog reset areset will override pmcatstb to reset analog blocks even when pmcatstb is active. Releasing pmcatstb will return all analog blocks to software control.

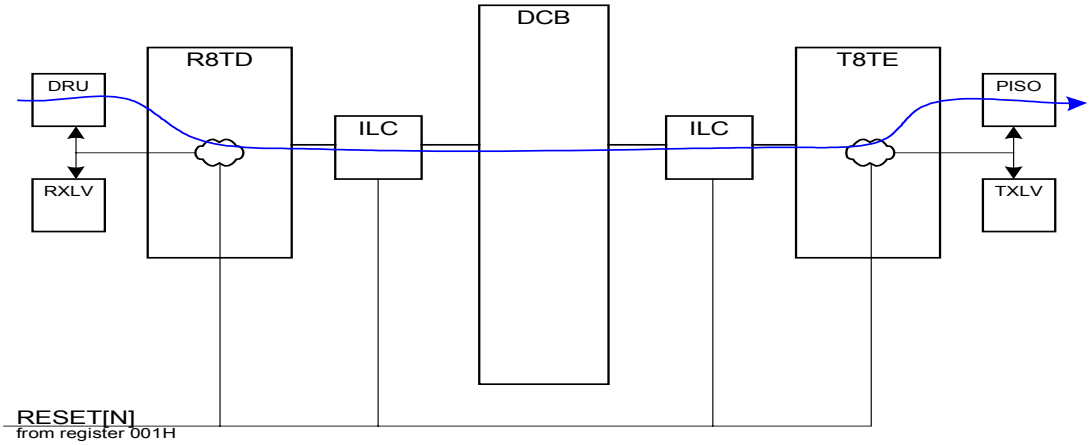
Normal mode:

Register 000H : DRESET ‘0’

Register 000H : ARESET ‘0’

Register 001H : RESET link dependent

Figure 17 Shutting Down a Link



12.1.4 PCB Design Notes

To maintain flexibility, all unused LVDS inputs and outputs should be left floating. This will prevent accidental damage caused by firmware enabling outputs, or releasing resets of inactive ports.

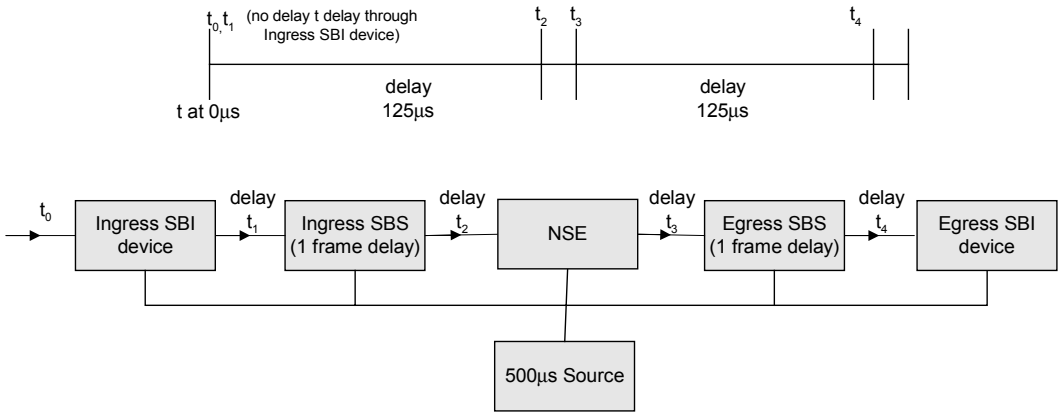
12.2 "C1" Synchronization

Any NSE/SBS fabric can be viewed as a collection of five "columns" of devices: column 0 consists of the ingress flow from the load devices (e.g., some SBI device); column 1 consists of the ingress flow through the SBS devices; column 2 consists of the NSE-8G device; column 3 consists of the egress flow through the SBS devices; and column 4 consists of the egress flow through the load devices (e.g. some SBI device). Note that the devices in columns 0 and 4 are SBI bus devices while columns 1 and 3 are SBS or SBS-lite devices. The dual column references refer to their two separate simplex flows. Path-aligned STS-12 frames are pipelined through this structure in a regular fashion, under control of a single clock source and frame pulse. There are latencies between these columns, and these latencies may vary from path to path. The following design is used to accommodate these latencies.

A timing pulse for SBI frames (2kHz, 500 µs) is generated and fed to each device in the fabric. Each chip has a *FrameDelay* register (RC1DLY) which contains the count of 77.76 MHz clock ticks that device should delay from the reference timing pulse before expecting the C1 characters of the ingress STS-12 frames to have arrived. The base timing pulse is called t. The delays from t based on the settings of the RC1DLY registers in the successive columns of the devices are called $t_0, \ldots t_4$. The first signal, t_1(equal to t_0), determines the start of an STS-12 frame; this signal is used to instruct the ingress load devices (column 0) to start emitting an STS-12 frame (with its special "C1" control character) at that time. t_i is determined by the customer, based on device and wiring delays to be approximately the earliest time that all "C1" characters will have arrived in the ingress FIFOs of the t_i column of devices. t_i is selected to provide assurance that all "C1" characters have arrived at the i^{th} column. The i^{th} column of devices use the t_i signal to synchronize emission of the STS-12 frames. The ingress FIFOs permit a variable latency in C1 arrival of up to 16 clock cycles.

Note: The SBS device, being a memory switch adds a latency of one complete frame or row plus a few clock ticks to the data.

Figure 18 "C1" Synchronization Control



12.3 Synchronized Control Setting Changes

The NSE-8G and SBS support dual switch control settings. These dual settings permit one bank of settings to be operational while the other bank is updated as a result of some new connection requests. The CMP input selects the current operational switch control settings. CMP is sampled by the NSE-8G on the RC1FP. The internal blocks sample the registered CMP value as they receive the next C1 character after a delay of RC1DLY. The new CMP value is applied on the first A1 character of the following STS-12 frame. This switchover is hitless; the control change does not disrupt the user data flow in any way. This feature is required for the addition of arbitrary new connections, as existing connections may need to be rerouted (see the discussion of the connection routing algorithm in this document).

The DS0-granularity switch settings RAM in organized into two control settings banks, these are switched by the above mechanisms on C1 boundaries. The NSE-8G also has to coordinate the switching of the connected SBS devices (if using the In-Band link facility), so a broader understanding of the issues is required.

The following sections provide examples.

12.3.1 SBS/NSE-8G Systems with DS0 and CAS Switching

When building a DS0 and Channel Associated Signaling switching system with the SBS, SBS-lite and NSE-8G devices the overall timing is based on the CAS signaling multiframe on the SBI bus. In this configuration the delay through the SBS devices is a single 125 µS SBI frame plus a few 77.76 MHz clocks and the delay through the NSE-8G is a few 77.76 MHz clocks. A single C1FP frame synchronization signal is distributed around the system. Internal to the SBS and NSE-8G devices are programmable offsets used to account for propagation delays through the system. The key constraint is that all SBI frames are aligned going into the NSE-8G device.

Compatible devices are TEMUX-84, FREEDM-336, FREEDM-336-84 bond-out, S/UNI-IMA-84, and other future SBI336 devices.

The SBS and NSE-8G devices have two configuration pages controlling the switching of each DS0 with CAS. The SBS has independent configuration pages for each direction of data flow through the device. The NSE-8G has one set of configuration pages. System configuration changes are made by writing to the offline configuration page in all affected devices and then swapping from the old configuration page to the new configuration page. The TCMP and OCMP signals control the current configuration page of the SBS and the CMP signal controls the current configuration page of the NSE. Swapping of configuration pages must be aligned to frame switching through the system to avoid any possible data corruption. The TCMP, OCMP and CMP signals are sampled with the SBS IC1FP and RC1FP signals and the NSE-8G RC1FP signals respectively. The CMP signals can be connected together at the expense of having to ensure all device configuration pages are current.

Figure 19 shows how the devices are connected together. The following timing diagrams show the external signals and the internal device frame alignment signal generated from the programmed delays. Although the CMP signals are sampled externally with the C1FP signals they are also delayed internally to coincide with the internally delayed frame signals. These are also shown in the timing diagram. All internal signals are identified by the .INT suffix.

Figure 19 TEMUX-84™/SBS/NSE/SBS/AAL1gator-32™ system DS0 Switching with CAS

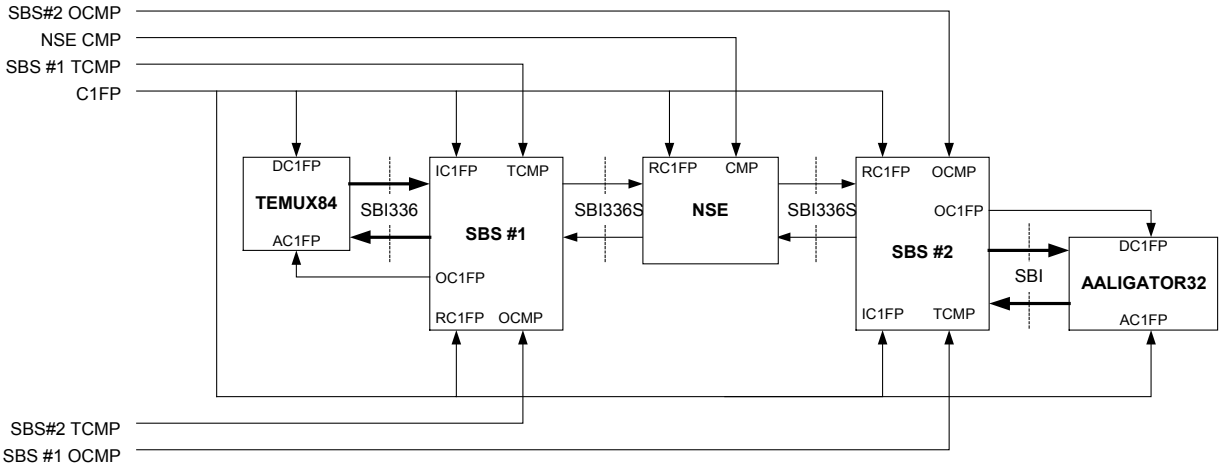


Figure 20 CAS Multiframe Timing

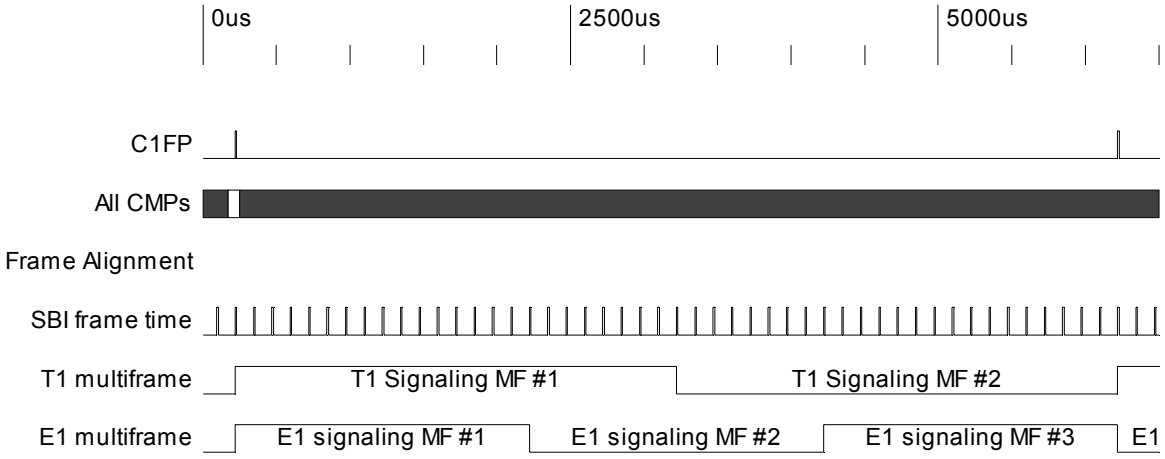
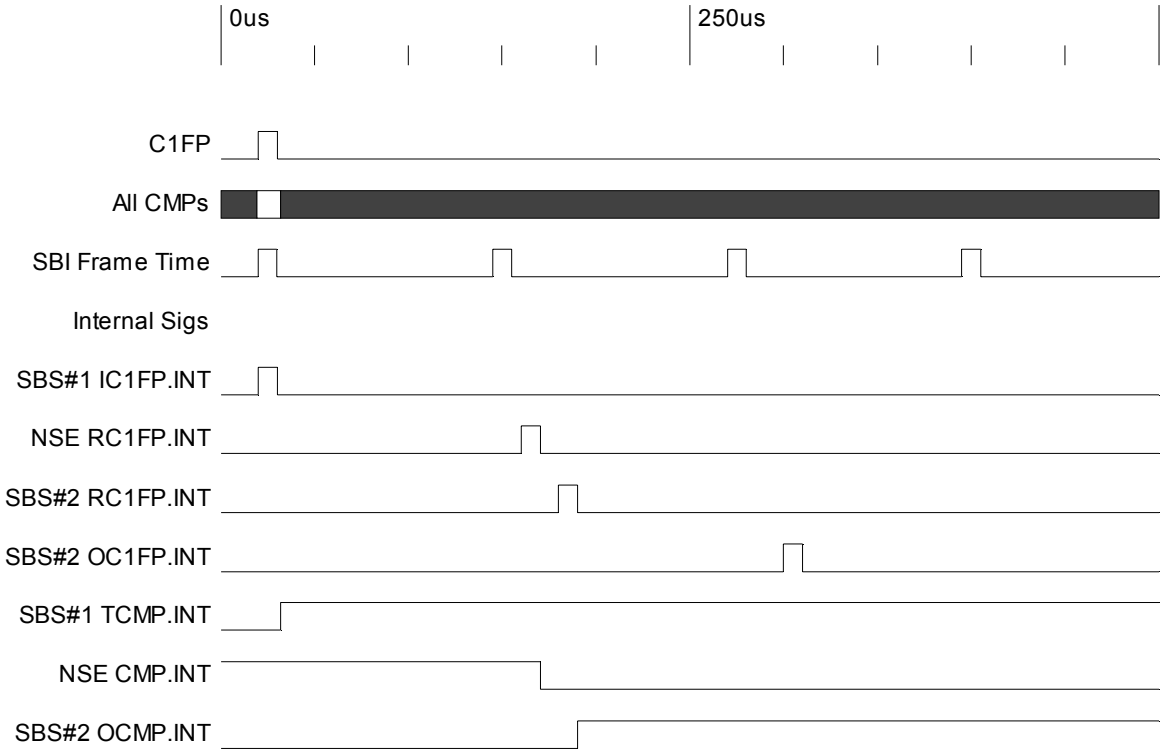


Figure 21 Switch Timing DSOs with CAS

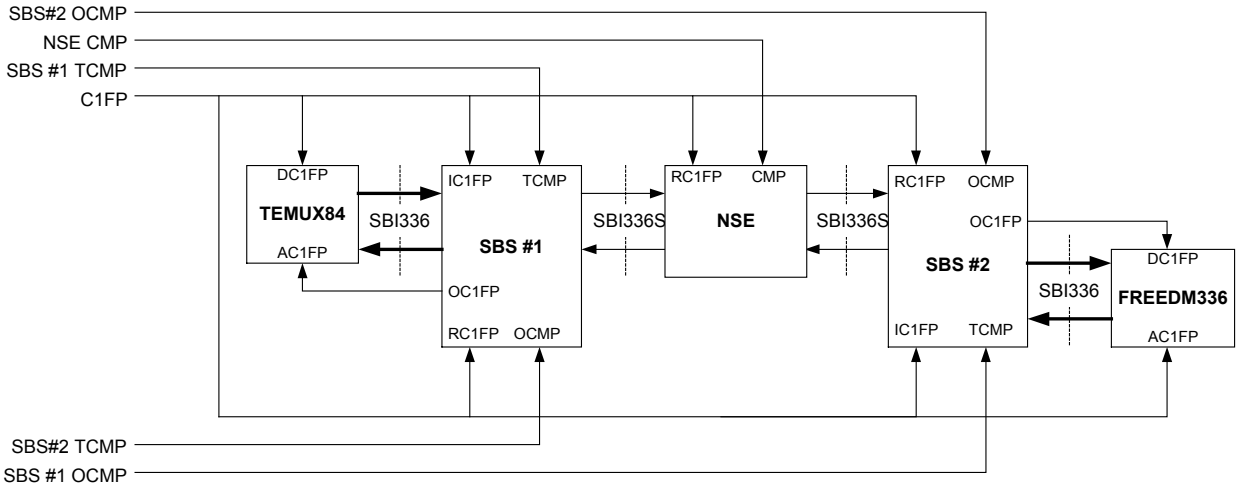


12.3.2 SBS/NSE-8G Systems Switching DS0s without CAS

This is very similar to the DS0 switching system configuration with CAS described in the previous section. The only difference is that in this system the global C1FP can be reduced to every SBI multiframe rather than the longer 48-frame SBI bus signaling multiframe. The advantage is that there is less latency when making switch configuration changes via the CMP signals.

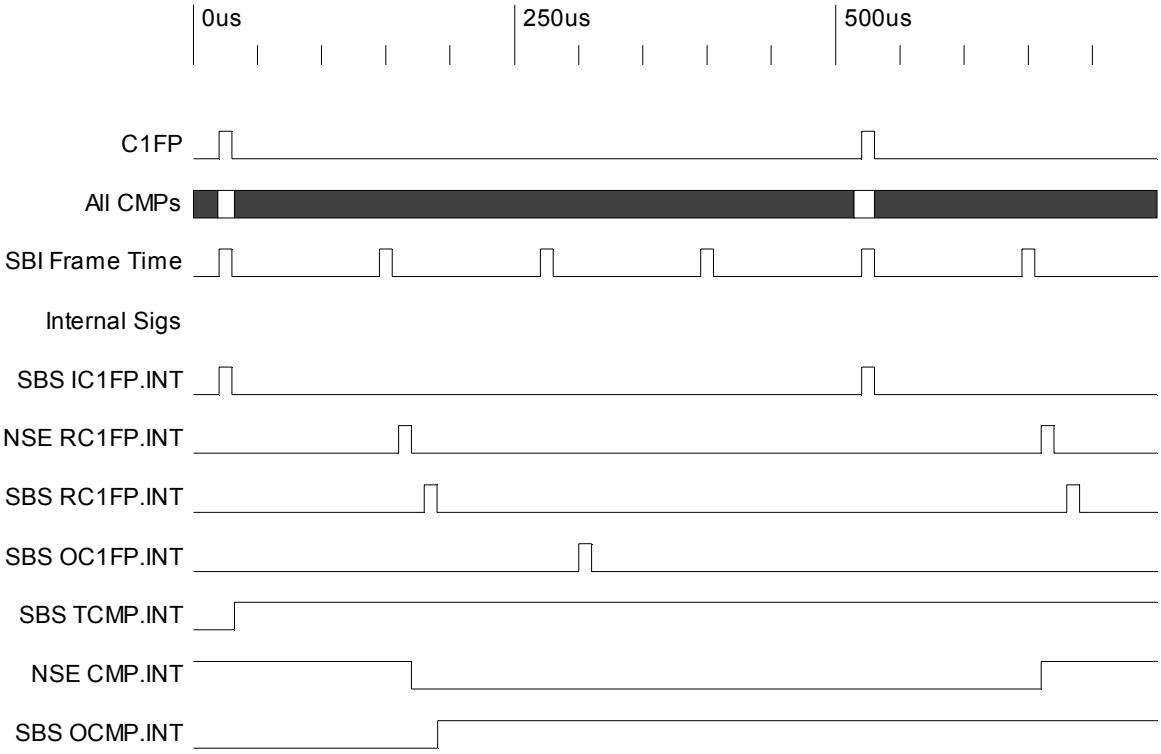
The following diagram shows the system with the FREEDM-336™, which does not require Channel Associated Signaling. Notice that the data latency through the system is the same as the case when switching DS0s with CAS.

Figure 22 TEMUX-84/SBS/NSE/SBS/FREEDM-336 System DS0 Switch no CAS



The following timing diagram shows the system timing when in this configuration.

Figure 23 Switch Timing - DSOs without CAS



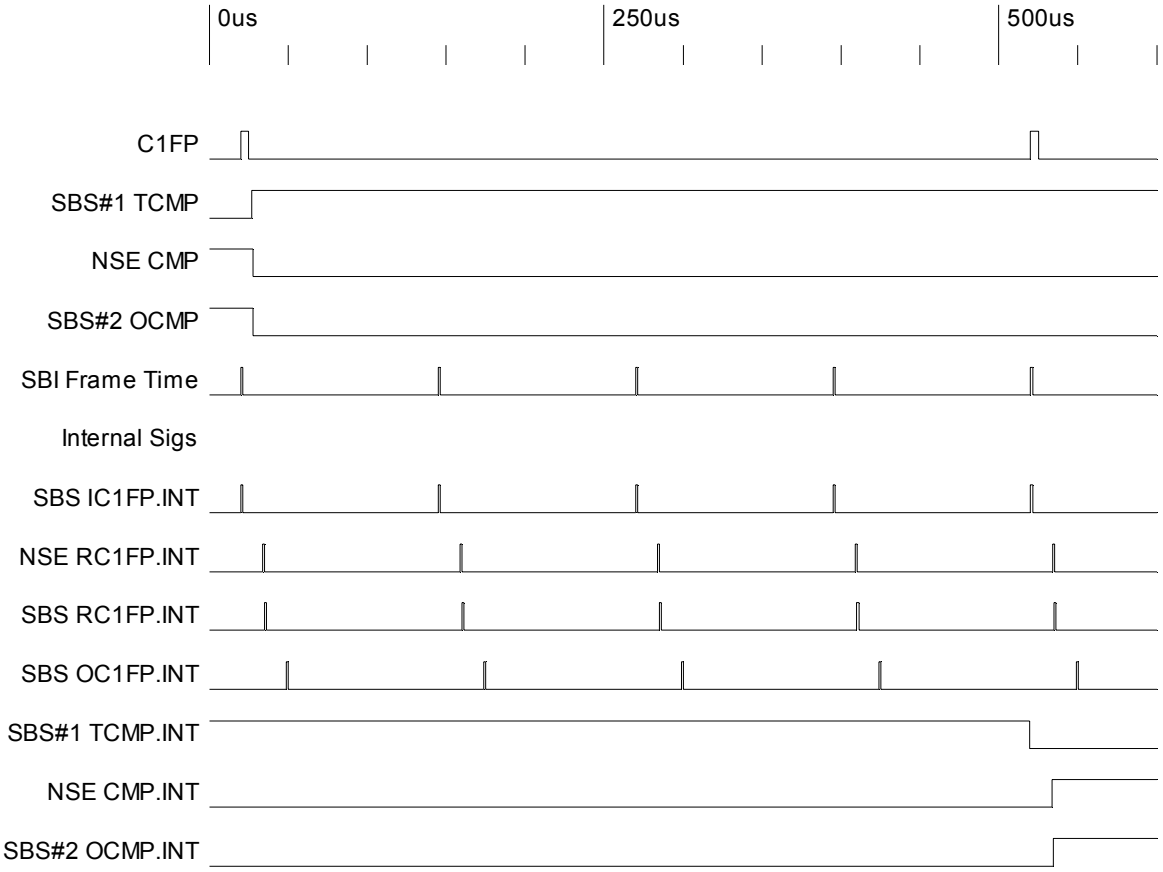
12.3.3 SBS/NSE-8G non-DS0 Level Switching with SBI336 Devices

The SBS and NSE-8G supports another mode of operation that has lower latency and lower power when not switching at the DS0 level. In this mode both of these devices become a column switch rather than a DS0 switch. This also saves SW configuration since only one row of the switch configuration rams has to be configured rather than all nine rows.

When switching DS0 through the system the SBS must store an entire frame of DS0s before routing them to the destination to allow for the last DS0 of a frame to be switched to the first DS0 of the output. When doing column switching only one row of the SBI structure needs to be stored before switching can take place.

Figure 22 from the previous section can be used here. The following timing diagram shows the system timing for this mode of operation.

Figure 24 Non DS0 Switch Timing



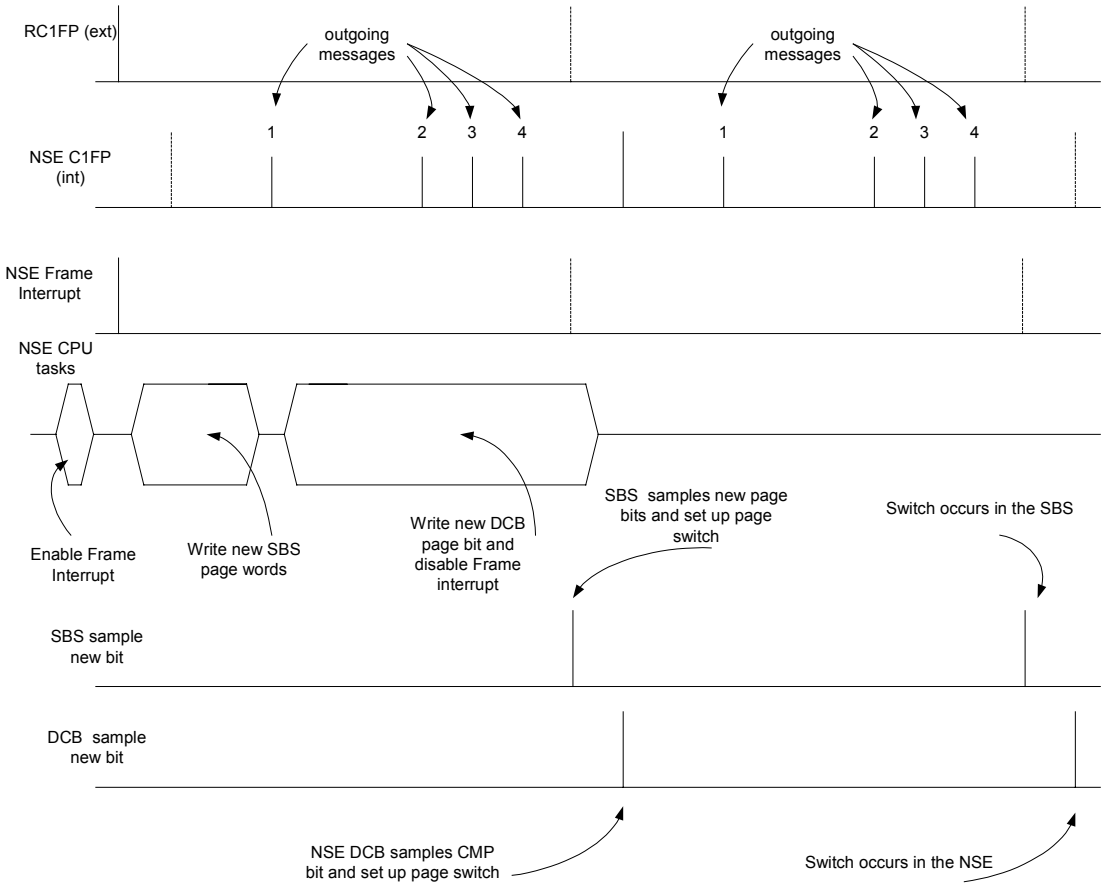
12.4 NSE-8G CPU Interaction with the Switching Cycle When Using the ILC

An interrupt is made available to the NSE-8G CPU called the Frame Interrupt this occurs at the start of the internal frame and marks a time in the NSE-8G where updates to the system page bits can occur. This interrupt is maskable and would normally be masked.

The CPU will need to enable this interrupt before a page switch is required, then respond to this interrupt immediately and complete writing the new page bit settings (a two double word operation) within 27 µs.

This is required as the ILC will sample the SBS page bits (in the ILCs) once during the frame before the first message is assembled and sent (starting at the beginning of row 3). If the page bits are updated late, the SBS pages will switch a frame late, which means the NSE-8G DCB may switch early giving disastrous results.

The NSE-8G CPU will have the rest of the frame to signal a page switch to the DCB as this is sampled on the next frame



12.5 Controlling Frame Alignment in the Receive Port

After external data corruption on any port it may be necessary to force OCA to reset the alignment of the R8TD block. In order to detect this out of alignment condition, three hardware functions are implemented for each port. The registers are:

- "Correct R8TD_RX_C1 Pulse Monitor", 012h
- "Unexpected R8TD_RX_C1 Pulse Interrupt", 013h
- "Missing R8TD_RX_C1 Pulse Interrupt", 014h

These are qualified against a delayed version of the RC1FP input, which should occur every 4 or 48 frames and in agreement with mf_swap mode (DCB Configuration Register, 04Ch). If all active ports are using carrying the same frequency of C1 frame pulses (1 in 4 or 1 in 48) then the unexpected interrupt (013h) should be used to signal that a C1 code word was detected at the wrong time, software can then poll the monitor register (012h) to see if the error condition is permanent.

If some links are switching DS0 traffic (“1 in 48” frame mode) and some are not (“1 in 4” frame mode), the input RC1FP and the qualifying signal from the DCB (from mf_swap), will be running at “1 in 48” frame mode. The links in “1 in 48” frame mode should use the unexpected interrupt while the others should use the missing interrupt.

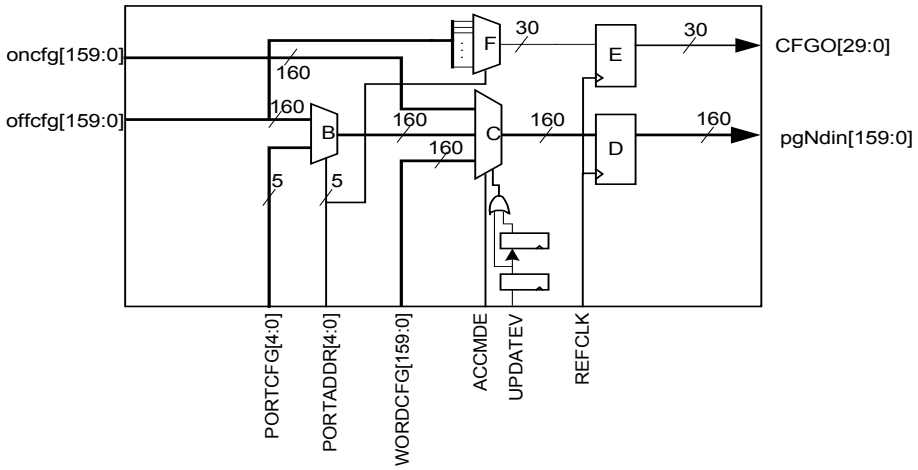
If a link no longer has any C1 activity, the firmware should assume the link has lost alignment, and should force R8TD OCA for the port.

These instructions assume the PMC NSE-8G Device driver is not being used. If the supplied driver is being used, this will all be handled within that driver.

12.6 DS0 Cross-Bar Switch (DCB) Operation

While the DCB is the Space Switch central to the NSE-8G it also contains the DCB C1 Delay Register. This register must be programmed with the delay (in 77.76 MHz clock cycles) between RC1FP (RC1DLY) and the expected arrival time in the R8TD SIPO of the C1 character. This value is expected to be in the order of 51 clock cycles + 9720 or +1080 for SBI mode or TeleCombus mode respectively. The 51 value is approximate and very dependant on the system architecture and transmission line lengths between the SBS and NSE-8G components. This must be obtained empirically by the system designer during product commissioning.

Figure 25 Architecture of the RAM Input Interface



12.6.1 Configuring the DCB using Port Transfer Mode

In port transfer mode, the microprocessor updates only one configuration entry within a word of offline connection memory page. The steps to perform a port transfer are shown in the following example:

Example: Suppose one wishes to change the cross bar to map DIN[10][12:0] to DOUT[6][12:0] for just the 4097th byte of the frame and wishes to keep all other mappings of the 4097th byte the same.

Steps:

1. CPU writes 0x8a061000 to the DCB Access Mode register (i.e., WRB=1, ACCMDE=0, PORTCFG[4:0]=0x0A, PORTADDR[4:0]=0x06, WORDADDR[13:0]=0x1000).
   - Causes MUX B to switch in PORTCFG at port 6.
   - Causes MUX C to switch in MUX B.
   - Triggers a read from offline memory connection page at location 0x1000.
2. Wait four SYSCLK cycles.
3. Writes 0x0a061000 to the DCB Access Mode register (i.e., WRB=0, ACCMDE=0, PORTCFG[4:0]=0x0A, PORTADDR[4:0]=0x06, WORDADDR[13:0]=0x1000).
   - Causes MUX B to switch in PORTCFG at port 6.
   - Causes MUX C to switch in MUX B.
   - Triggers a write from register D to offline connection memory page at location 0x1000.
4. Wait four SYSCLK cycles before returning to step 1 to perform another mapping change.

12.6.2 Configuring the DCB using Word transfer mode:

In word transfer mode, the microprocessor updates the entire word of offline connection memory page. The steps to perform a word transfer is shown in the following example:

Example: Suppose one wishes to change the entire cross bar mapping from DIN to DOUT for just the 4097th byte of the frame.

Steps:

1. CPU writes new mapping to the Configuration 31-30 Port register.
2. CPU writes new mapping to the Configuration 29-24 Port register.
3. CPU writes new mapping to the Configuration 23-18 Port register.
4. CPU writes new mapping to the Configuration 17-12 Port register.
5. CPU writes new mapping to the Configuration 11-6 Port register.
6. CPU writes new mapping to the Configuration 5-0 Port register .
7. CPU write 0x40001000 to the DCB Access Mode register.
   - Causes Mux C to switch in WORDCFG.

- Triggers a write to offline connection memory page at 0x1000.

Go to step 1 to begin the mapping change for a new byte in the frame.

12.6.3 Reading Configurations

It is possible to read configurations from the offline connection memory page. The following example shows this reading operation.

Example: Suppose one wishes to read which DIN ports map to DOUT[17:12][12:0] for the 4097th byte of the frame within the offline connection memory page.

Steps:

1. CPU writes 0x80081000 to the DCB Access Mode register. (A binary value of 010XX on PORTADDR[4:0] indicates to supply the mapping the Configuration Port 17-12 register.)

2. Wait for six SYSCLK cycles.

3. CPU reads the mapping from the DCB Configuration Output register.

Note:

1. The Access Mode register should NOT be accessed more frequently than once ever 4 SYSCLK cycles when initiating write transfers.
2. When initiating a read from the offline connection memory page to the Configuration Output register, there is a latency of 6 SYSCLK cycles from when a read is initiated till when valid data appears on CFG_O.
3. User should perform this operation only when there is no page swap pending (SWAPV = ‘0’) and page copy is inactive (UPDATEV = ‘0’).

12.6.4 DCB Online to Offline Memory Page Copy

There are two ways in which a connection memory page copy can occur; forced and automatic.

In forced mode, the CPU initiates a page copy by writing to the DCB Interrupt Status register. The page copy begins immediately after being initiated.

In automatic mode, the AUTO field must be set to logic one. When a connection memory page swap occurs, the online connection memory page is copied to the offline connection memory page.

Interrupt generation to signal the page copying status can be enabled to simplify software scheduling by setting the UPDATEE field in the DCB Configuration register to logic one. In this mode, the UPDATEI field in the DCB Interrupt Status register can be used as the interrupt signal to control the microprocessor.

Alternatively, the microprocessor can poll the UPDATEV field within the DCB Configuration register to detect the status of the connection memory page update logic. Logic one indicates copying in progress and logic zero indicates copying complete.

Warning: Attempting a page copy while a page swap is pending can lead to corruption of both online and offline memory pages if the page swap occurred while the page copy is in progress.

12.7 Telecombus Mode Operation

In Telecombus mode operation, only 1080 words of the configuration RAM are utilized. This same configuration is repeated nine times for switching the entire 9720 byte OC-12 frame. In this mode, RC1FP is flywheeled internally every frame so that page swaps can also occur at this frequency. The OC1FP output pulse that is used to resynchronize external TSBs occurs at every 4th frame.

To configure this mode of operation, use the following programming steps. Note, the order in which these steps are followed is irrelevant.

1. Program the DCB Frame Size register (0x4A) to 1079.
   - This programs the DCB to use just 1080 location of the RAMs.
2. Program the DCB MF_SWAP bits in Configuration register (0x4C) to 00.
   - This will program the DCB to effect page changes at every 9720 byte frame when a page swap request is received.
   - OC1FP will be output at every 4th frame.
   - CMP inputs will be sampled every frame at the internally flywheeled RC1FP location.
   - If enabled, FRAMEI will occur every frame at the internally flywheeled RC1FP location.

12.8 SBI Column Mode Operation

In SBI column mode operation, only 1080 rows of the configuration RAM is utilized, this same configuration is repeated nine times for switching the entire 9720 byte OC12 frame and 36 times to form the 4-frame multiframe. In this mode, RC1FP is flywheeled internally every fourth frame so that page swap can also occur at this frequency. OC1FP output that is used to resynchronize external TSBs also occurs at every fourth frame.

To configure this mode of operation, use the following programming steps. Note, the order in which these steps are followed is irrelevant.

1. Program the DCB Frame Size register (0x04A) to 1079.
   - This programs the DCB to use just 1080 location of the RAMs.
2. Program the MF_SWAP bits in DCB Configuration register (0x04C) to 01.
   - This will program the DCB to effect page changes at every 4 x 9720 frame when a page swap request is received.
   - OC1FP will be output at every 4 frame.
   - CMP inputs will be sampled every 4 frame at the internally flywheeled RC1FP location.

- If enabled, FRAMEI will occur every 4 frame at the internally flywheeled RC1FP location.

12.9 SBI DS0 Mode Operation

In SBI DS0 mode operation, all 9720 words of the configuration RAM are utilized. This same configuration is repeated four times to switch the 4-frame multiframe. In this mode, RC1FP is flywheeled internally every four frames so that page swaps can also occur at this frequency. A pulse on the OC1FP output that is used to resynchronize external TSBs also occurs every four frames.

To configure this mode of operation, use the following programming steps. Note, the order in which these steps are followed is irrelevant.

1. Program the DCB Frame Size register (0x4A) to 9719.
   - This programs the DCB to use all 9720 location of the RAMs.
2. Program the MF_SWAP bits in DCB Configuration register (0x4C) to 10.
   - This will program the DCB to effect page changes at every 4 x 9720 frame when a page swap request is received.
   - OC1FP will pulse high at every 4th frame.
   - CMP inputs will be sampled every 4th frame at the internally flywheeled RC1FP location.
   - If enabled, FRAMEI will occur every 4th frame at the internally flywheeled RC1FP location.

12.10 SBI DS0 with CAS Mode Operation

In SBI DS0 with CAS mode operation, all 9720 words of the configuration RAM are utilized. This same configuration is repeated 48 times to switch the 48 frame multiframe. In this mode, RC1FP is flywheeled internally every 48th frame so that page swaps can also occur at this frequency. A pulse on the OC1FP output that is used to resynchronize external TSBs also occurs at every 48th frame.

To configure this mode of operation, use the following programming steps. Note, the order in which these steps are followed is irrelevant.

1. Program the DCB Frame Size register (0x4A) to 9719.
   - This programs the DCB to use all 9720 location of the RAMs.
2. Program the MF_SWAP bits in DCB Configuration register (0x4C) to 11.
   - This will program the DCB to effect page changes at every 48 x 9720 frame when a page swap request is received.
   - OC1FP will pulse every 48th frame.

- CMP inputs will be sampled every 48th frame at the internally flywheeled RC1FP location.
- If enabled, FRAMEI will occur every 48 frame at the internally flywheeled RC1FP location.

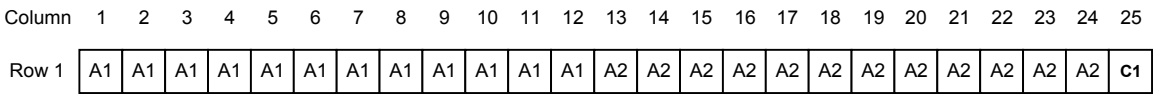
Note: It is vital to ensure that switching of the DS0 bytes containing CAS bits be performed correctly through software configuration. That is, these bytes should all be preserved and switched to the same output link to preserve the CAS for downstream devices.

12.11 ILC Operation

Operating each of the 32 ILC blocks requires the same procedure. Each ILC will be operating independently and wait states required for each ILC can be satisfied by interleaving the access cycles of several ILC blocks together.

The ILC is synchronized by the C1 pulse accompanying the input data stream on the TelecomBus. It preloads a 9720 counter using this C1 pulse. C1 as shown in Figure 26, will be high when the byte in column 25, row 1 is on the input data pins. A 2 bit counter is also kept to keep track of the 4-frame multiframe, ie. 4 x 9720 count. This is indicated by C1 being present only in the first frame of a multiframe. (could be 1in 4 or higher multiples).

Figure 26 C1 Position in the First Row



The ILC inserts and retrieves messages from the transport overhead of the SONET/SDH frame on the telecom-bus. Figure 27 illustrates the four rows carrying the four messages per frame.

The messages are inserted on an availability basis into the four message rows shown in Figure 27, rows 3,6,7 and 8. The header is always placed into columns 1 and 2. The Message itself is always placed MSByte first into columns 3 – 34, in FIFO order. The CRC-16, calculated over the header and message, is placed into columns 35 and 36.

If no message is available, internal hardware will automatically insert zeros into the message bytes. Even if no message is available (32 of 36 bytes), the header(2 of 36 bytes) can still be carrying valid bits for the far end, as such even if the message is invalid, the Header and CRC are still generated and inserted. The header's Valid bit is not set as an indication to the far end to discard the message (not insert the null message into its RxFIFO).

12.12 ILC CPU Operations

12.12.1 Accessing the Transmit Message FIFO

Access registers in the following order:

1. Write 00000001h to the ILC Transmit Misc. Status and FIFO Synch register to initiate a new message write sequence (TX_XFER_SYNC).

2. Loop through the next two operations until the message is transferred.

3. Write the Transmit FIFO Data register.

4. Wait for the TX_FI_BUSY bit to be logic zero (read ILC Transmit Misc. Status and FIFO Synch register) or wait for three SYSCLK Cycles and then continue. This wait time can be used to access other registers, just as long as the Transmit FIFO Data register is not changed during this time.

Notes on Using the TX_XFER_SYNC

If all messages transmitted are eight Dwords in length, then writing the TX_XFER_SYNC bit is optional.

When transmitting a message immediately following a short message, writing the TX_XFER_SYNC bit is mandatory.

When transmitting messages longer than eight Dwords, writing the TX_XFER_SYNC bit every eight Dwords is optional.

12.12.2 Accessing the Receive Message FIFO

Access the registers in the following order:

This procedure is outline only. Full operation is shown in the code snippet below.

1. Write 00000001h to the Receive Auxiliary, Status and FIFO Synch register to initiate a new message read sequence.

2. Loop through the next two operations until message is transferred.

3. Read the Receive FIFO Data register.

4. Wait for the busy bit to equal ‘0’ (Read Receive Status register Ah) or wait for three SYSCLK Cycles and then continue. This wait time can be utilized accessing other registers, just as long as the Receive FIFO Data register is not read again during this time.

The following psuedo code shows the recommended procedure for accessing the receive message FIFO.

```
    PROCEDURE Message_Receive IS
      VARIABLE dword_rd_cnt   : NATURAL := 0;
      VARIABLE msg_done       : BOOLEAN := false;
      VARIABLE polled_rx_mode : BOOLEAN := false;
      VARIABLE msg_lvl_loop   : NATURAL := 0;
      VARIABLE msgs_rd        : NATURAL := 0;
    BEGIN
--
```

```
--    Get the MSG_LVL
--
      rd(RX_STTS);
      WHILE rx_stts_valid = '0' LOOP
        rd(RX_STTS);
      END LOOP;
      msg_lvl_loop := rx_msg_lvl;
--
--    Now Check the RX_SYNC_DONE status to be sure we're not going to skip
--    the 1st message...
--
      IF rx_sync_done = '0' THEN
--
--    Do the RX_XFER_SYNC
--
        wr(RX_STTS, "00000000000000000000000000000001");
--
--    Poll the RX_FI_BUSY bit
--
        rd(RX_STTS);
        WHILE rx_fi_busy = '1' LOOP
          rd(RX_STTS);
        END LOOP;
      END IF;
--
--    Next RX MSG Buffer is now synched and we know how many messages we have.
--    Just read the messages out.
--
      msgs_rd := 0;
      tst_report("Processing " & to_str(msg_lvl_loop) & " messages.");
      WHILE msgs_rd < msg_lvl_loop LOOP    -- Loop on read value of rx_msg_lvl
--
--    Check the CRC_ERR bit for this message
--
        rd(RX_STTS);
        rxmsg.crc_err := crc_err_reg;            ---  Simulation Stuff

--
--    This code skips errored messages.
--
        WHILE crc_err_reg = '1' AND rndm_skip_msg_mode LOOP
          tst_report("Errored message received and being skipped...");
          IF rx_sync_done = '1' THEN
            wr(RX_STTS, "00000000000000000000000000000001");
          ELSE
--
-- We're in this loop 2nd time without a good msg inbetween. Need 2
-- RX_XFER_SYNC writes to do a message skip
--
            wr(RX_STTS, "00000000000000000000000000000001");
--
--   Poll the bit here before starting next message.
--
            rd(RX_STTS);
            WHILE rx_fi_busy = '1' LOOP
              rd(RX_STTS);
          END IF;
          rx_msg_cnt_o <= rx_msg_cnt_o + 1;
          msgs_rd := msgs_rd + 1;
--
```

```
--   Poll the bit here before startiong next message.
--
          rd(RX_STTS);
          WHILE rx_fi_busy = '1' LOOP
            rd(RX_STTS);
          END LOOP;
--
--    Check the CRC_ERR bit for the next message
--
          rd(RX_STTS);
          rxmsg.crc_err := crc_err_reg;
        END LOOP;

--
--    Now we should have a good message or an errored message we want to read
--    Note: we might have skipped the last message in  the code above.
--    Read the message IF we have any.
--
        IF msgs_rd < msg_lvl_loop THEN
          msg_done      := false;
          dword_rd_cnt  := 0;
          rxmsg.payload := (OTHERS => 0);
          WHILE dword_rd_cnt < 8 AND NOT msg_done LOOP
            rd(RX_DAT);          -- The returned data is part of the payload
---   Simulation Stuff. Should replace with S/W stuff
            FOR j IN 3 DOWNTO 0 LOOP
              rxmsg.payload((dword_rd_cnt*4)+(3-j)) :=
                              conv_integer(reg_rd_data(8*j+7 DOWNTO 8*j));
            END LOOP;
            IF dword_rd_cnt = 0 THEN
              rndm_rx_payload_len <= conv_integer(reg_rd_data(31 DOWNTO 24));
            END IF;
---   End Simulation Stuff.
            dword_rd_cnt := dword_rd_cnt + 1;

            IF polled_rx_mode THEN
              rd(RX_STTS);
              WHILE rx_fi_busy = '1' LOOP
                rd(RX_STTS);
              END LOOP;
            ELSE
--
-- Fixed delay to allow to ensure 4 SYSCLKs plus o/p delay between RDB edges
--
              WAIT FOR 36 ns;
            END IF;
--
--   Optional S/W algorithm to determine if the MSG is DONE before 8 Dwords
--   have been read. In this case for SIMULATION the test the msg_done is set
--   when we've read the number of dwords as indicated in the 1st message
--   payload byte.
--
            IF dword_rd_cnt = rndm_rx_payload_len THEN
              msg_done := true;
            END IF;

            IF msg_done AND dword_rd_cnt < 8 THEN
--
-- We're on a short message. Need to do RX_XFER_SYNC
--
```

```
              tst_report("Executing Receive buffer resync " & "
                         following short message of length " &
                           to_str(dword_rd_cnt) & ".");
              wr(RX_STTS, "00000000000000000000000000000001");
--
--    Poll the bit here before startiong next message.
--
              rd(RX_STTS);
              WHILE rx_fi_busy = '1' LOOP
                rd(RX_STTS);
              END LOOP;
            END IF;
          END LOOP;
---   Simulation Stuff
          Check_Rx_Msg(Payload_Chk,No_Header_Chk); -- Check payload only.
          rx_msg_cnt_o <= rx_msg_cnt_o + 1;
---   End Simulation Stuff
          msgs_rd := msgs_rd + 1;
        ELSE                          --    msgs_rd = msg_lvl_loop
--
--     When we think we're done check that we really are.
--     Update the current value of RX_MSG_LVL. Allows for the case where a
--     message has arrived while processing.
--     At this point we have read all the messages we thought we had at the
--     start of the loop so whatever number is returned now is the number of
--     new messages receieved while we've been reading messages.
--     Unless we're running real slow this should only be 1 or 0.
--     We adjust our loop variable by this amount to keep the loop going
--     long enough to get the extra messages received.
--
          rd(RX_STTS);
          WHILE rx_stts_valid = '0' OR rx_fi_busy = '1' LOOP
            rd(RX_STTS);
          END LOOP;
          IF rx_msg_lvl > 0 THEN
            tst_report("Message(s) Received while processing messages");
          END IF;
--
--     Note in this code I'm not checking for more messages received than
--     read. We'd be in real trouble in the system in this case! We
--     also can't get there in the sim!
--
          msgs_rd := msgs_rd - rx_msg_lvl;
        END IF;
      END LOOP;
    END Message_Receive;
```

12.12.3 Handling the Transmit Header

PAGE Bits

If the IPAGE bits are changed, they are not sent in the header bits until the next frame. They will be continually sent for each message in subsequent frames until they change again.

USER, LINK and AUX Bits

When any of these bits change they are sent in the header bits of the next message. They will be continually sent for each subsequent message until they change again.

12.12.4 Handling the Receive Header

12.12.5 Handling Interrupts

All interrupts are masked on startup, and should not be enabled until the link initializes.

12.12.6 Bypass Function

Functional block transmit and receive functions can be disabled independently (or together) by writing a ‘1’ to:

- The LSB (TX_BYPASS) of the Transmit Control register to enable transmit bypass.
- The LSB (RX_BYPASS) of the Receive Control register to enable receive bypass.

When in bypass mode, message FIFO rams are disabled and the ILC functions as a 2-stage pipeline. When TX_BYPASS is set, writes to the transmit FIFO are ignored and when RX_BYPASS is set, reads from the receive FIFO return random data.

Figure 27 Transport Overhead Affected by ILC

Row 1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A2	A2	A2	A2	A2	A2
Row 2	B1	B1	B1	B1	B1	B1	B1	B1	B1	B1	B1	B1	E1	E1	E1	E1	E1	E1
Row 3	**D1**	**D1**	**D1**	**D1**	**D1**	**D1**	**D1**	**D1**	**D1**	**D1**	**D1**	**D1**	**D2**	**D2**	**D2**	**D2**	**D2**	**D2**
Row 4	H1	H1	H1	H1	H1	H1	H1	H1	H1	H1	H1	H1	H2	H2	H2	H2	H2	H2
Row 5	B2	B2	B2	B2	B2	B2	B2	B2	B2	B2	B2	B2	K1	K1	K1	K1	K1	K1
Row 6	**D4**	**D4**	**D4**	**D4**	**D4**	**D4**	**D4**	**D4**	**D4**	**D4**	**D4**	**D4**	**D5**	**D5**	**D5**	**D5**	**D5**	**D5**
Row 7	**D7**	**D7**	**D7**	**D7**	**D7**	**D7**	**D7**	**D7**	**D7**	**D7**	**D7**	**D7**	**D8**	**D8**	**D8**	**D8**	**D8**	**D8**
Row 8	**D10**	**D10**	**D10**	**D10**	**D10**	**D10**	**D10**	**D10**	**D10**	**D10**	**D10**	**D10**	**D11**	**D11**	**D11**	**D11**	**D11**	**D11**
Row 9	S1/Z1	S1/Z1	S1/Z1	S1/Z1	S1/Z1	S1/Z1	S1/Z1	S1/Z1	S1/Z1	S1/Z1	S1/Z1	S1/Z1	M0 M1/ Z2	M0 M1/ Z2	M0 M1/ Z2	M0 M1/ Z2	M0 M1/ Z2	M0 M1/ Z2

Column	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
Row 1	A2	A2	A2	A2	A2	A2	**C1**	J0/Z0 (C1)	J0/Z0 (C1)	J0/Z0 (C1)	J0/Z0 (C1)	J0/Z0 (C1)	J0/Z0 (C1)	J0/Z0 (C1)	J0/Z0 (C1)	J0/Z0 (C1)	J0/Z0 (C1)	J0/Z0 (C1)
Row 2	E1	E1	E1	E1	E1	E1	F1	F1	F1	F1	F1	F1	F1	F1	F1	F1	F1	F1
Row 3	**D2**	**D2**	**D2**	**D2**	**D2**	**D2**	**D3**	**D3**	**D3**	**D3**	**D3**	**D3**	**D3**	**D3**	**D3**	**D3**	**D3**	**D3**
Row 4	H2	H2	H2	H2	H2	H2	H3	H3	H3	H3	H3	H3	H3	H3	H3	H3	H3	H3
Row 5	K1	K1	K1	K1	K1	K1	K2	K2	K2	K2	K2	K2	K2	K2	K2	K2	K2	K2
Row 6	**D5**	**D5**	**D5**	**D5**	**D5**	**D5**	**D6**	**D6**	**D6**	**D6**	**D6**	**D6**	**D6**	**D6**	**D6**	**D6**	**D6**	**D6**
Row 7	**D8**	**D8**	**D8**	**D8**	**D8**	**D8**	**D9**	**D9**	**D9**	**D9**	**D9**	**D9**	**D9**	**D9**	**D9**	**D9**	**D9**	**D9**
Row 8	**D11**	**D11**	**D11**	**D11**	**D11**	**D11**	**D12**	**D12**	**D12**	**D12**	**D12**	**D12**	**D12**	**D12**	**D12**	**D12**	**D12**	**D12**
Row 9	M0 M1/ Z2	M0 M1/ Z2	M0 M1/ Z2	M0 M1/ Z2	M0 M1/ Z2	M0 M1/ Z2	E2 Undef	E2 Undef	E2 Undef	E2 Undef	E2 Undef	E2 Undef	E2 Undef	E2 Undef	E2 Undef	E2 Undef	E2 Undef	E2 Undef

If a message is available, it will be inserted into the next message row available.

If the page bits are changed, they are not sent in the header bits until the first message in the next frame. They will be continually sent until they change again. All other header bits are sent immediately.

All ILC interrupts are masked on startup, and should not be enabled until the ILC links initialize.

12.13 Switch Setting Algorithm

NSE/SBS fabrics require an algorithm to map from customers' connection requirements to settings in the switch function control registers in these devices. Four constraints apply to this algorithm:

- The algorithm must succeed for arbitrary permutation requests (i.e., neither the fabric nor thealgorithm can fail to connect any one-to-one connection request).
- The algorithm must permit connection of 2-cast requests (port replication for either snooping or for advanced redundancy fabrics). In fabrics with spare capacity and multicast/broadcast servers, the algorithm must permit mapping of multicast/broadcast requests, up to the capacity of the fabric and the servers.
- This algorithm must be fast enough to satisfy requirements for response to operator requests for connection changes.
- This algorithm must be fast enough to satisfy requirements for protection responses to equipment failures.

There are several aspects of this problem:

- Reconnection requests may be made individually in which case an incremental connection-setting algorithm is desired, or as complete batches in which case a batch algorithm may be desirable.
- Reconnection requests may be pre-computed for fast protection fail-over mechanisms.

12.13.1 Problem Description

The basic scheduling problem is to find the switch settings to properly route a set of connections. This is more formally described using the definitions in the following paragraphs.

Port: An STS-12 input/output data stream. The serial ports on the SBS devices and the NSE-devices operate at STS-12 rates and utilize STS-12 frames. Since the intention of the NSE-8G is to serve as a DS0-granularity switch, these STS-12 frames must be treated as repeating on a cycle of 12*9*90 = 9720 octets. All connections considered by this algorithm are octet connections. Higher aggregations of traffic are handled as collections of octets, and are ignored for the purposes of describing this algorithm.

Timeslot: A specific octet location in the 9720 octet cyclic structure.

Spacetimeslot: A timeslot on a specific port, identified by a space component and a time component: for example, octet 9 on port 3 of SBS device 2

Connection: A mapping of an input spacetimeslot to an output spacetimeslot. Connections come in two varieties, multicast and unicast. Unicast connections are a mapping of a single input spacetimeslot to a single output spacetimeslot. Multicast connections are a mapping of a single spacetimeslot to multiple output spacetimeslots. This algorithm is only concerned with the unicast problem.

12.13.2 Naïve Algorithm

We begin by describing a simplified version of the algorithm, applied to a specific SBS/NSE-8G configuration. Four SBS devices are connected by one port each to an NSE, which is likewise connected by one port to the egress side of each SBS device. Only four ingress/egress ports on the NSE-8G are in use in this application, but the ideas generalize easily to larger fabrics.

Information flows from left to right. Each edge connects an egress port (on the left) to an ingress port (on the right); each such edge has a capacity of 9720 timeslots.

For present purposes, we consider the SBSs to be supporting a single P-SBI port (eight bits at 77.76 MHz, or STS-12). Also, we ignore the "standby" LVDS port. This reduces the SBS from a multi-ported Memory switch (which it in fact is) to a simpler two-ported (P-SBI and Active S-SBI) Time switch. This reduction in complexity makes the following discussion more straightforward, but does not reduce the algorithm's ability to deal with the more complex cases introduced by the use of the four slower P-SBI ports, or by concurrent use of the standby LVDS port. The nature of switching in this application is illustrated by Figure 19. The two dimensional 4-X-4 matrices represent octet slots in both space (vertical) and time (horizontal). We trace through the switching processing in the following steps:

Matrix I represents the arrival of the 16 octets from the SBI load devices.

The mapping from Matrix I to Matrix II represents the Time switching action of all four ingress SBSs. Each SBS carries out an arbitrary permutation (including 1-to-many) of the ingress Time slots within each Space row.

The mapping from Matrix II to mat Matrix rix III represents the Space switching action of the NSE. During each Time slot, the NSE-8G carries out an arbitrary permutation (including 1-to-many) of the ingress Space slots.

The mapping from Matrix III to Matrix IV represents the Time switching action of all four egress SBSs. Each SBS carries out an arbitrary permutation (including 1-to-many) of the ingress Time slots within each Space row.

It is known that any complete permutation from Matrix I to Matrix IV can be carried out in this way. Figure 19 illustrates two particular octets (α and β) being switched through the SBS-NSE-SBS Time:Space:Time switch.

Figure 28 Example Graph

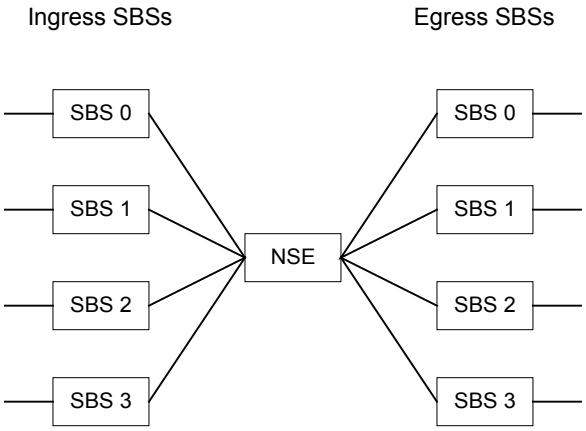
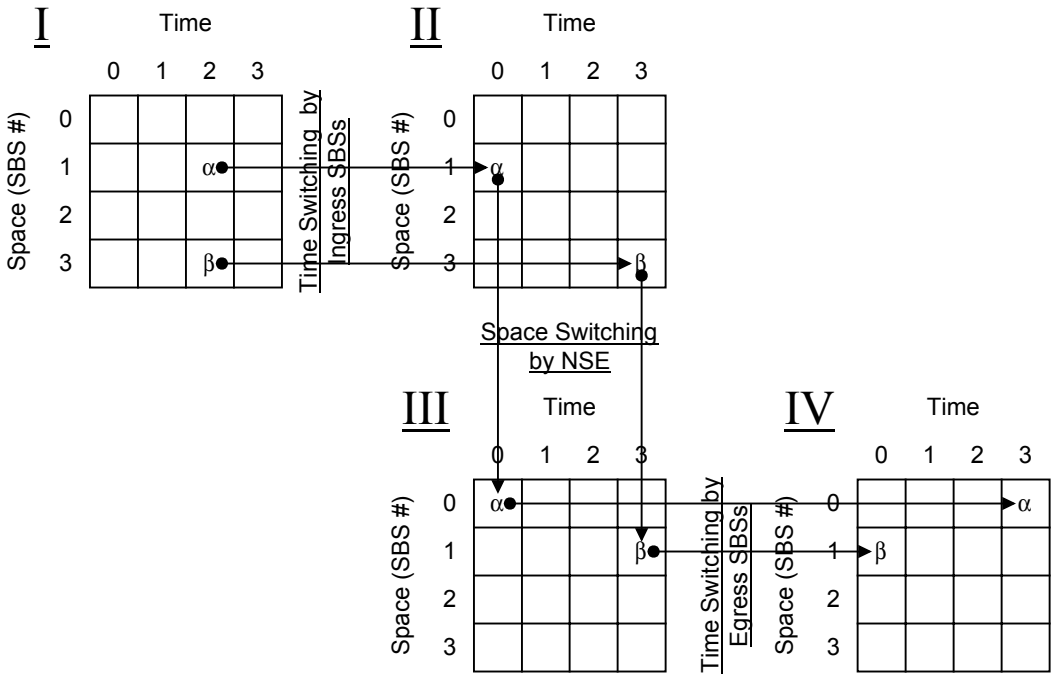


Figure 29 Time:Space:Time Switching in one NSE-8G and four Single-Ported SBSs



α: (S=1, T=2) => SBS => (S=1, T=0) => NSE => (S=0, T=0) => SBS => (S=0, T=3)

β: (S=3, T=2) => SBS => (S=3, T=3) => NSE => (S=1, T=3) => SBS => (S=1, T=0)

Consider a request to route an octet from ingress port *i* to egress port *j,* where *i* and j range from 0 to 3, over four ports corresponding to the four SBS devices. To make this connection, we must find a timeslot in the NSE-8G which can accept an octet from the ingress SBS and send an octet to the egress SBS. If the NSE-8G has these two slots free in the same timeslot, then the SBSs must also have the corresponding slot free. The actual routing of the sample is accomplished in several steps. The octet is:

1. mapped to the free timeslot by the ingress SBS port,

2. picked up by the NSE-8G in that timeslot on the port from the ingress SBS and mapped to the port which leads to the egress SBS,

3. picked up by the egress SBS in the expected timeslot.

It may not be possible to find a free time which connects the ingress SBS to the egress SBS, even though both SBS devices have unused capacity into the NSE-8G core (the ingress SBS may have a free timeslot at time *i* and the egress SBS may have a free timeslot at time *j,* but *i ~= j).* Such cases require a more complex algorithm which is capable of disconnecting and reconnecting other connections to make space for the new *i* to *j* connection. (Disconnection and reconnection of other connections is done hitlessly by NSE/SBS fabrics.) This more sophisticated algorithm is described in the remainder of this section.

12.13.3 Bi-partite graphs

A general solution to the connection problem is a schedule where each connection is assigned to one of the 9720 timeslots in each time stage such that no two connections conflict. This solution then maps to physical switch settings for the SBS and NSE-8G devices. The following definitions allow us to represent the problem as an abstract graph problem:

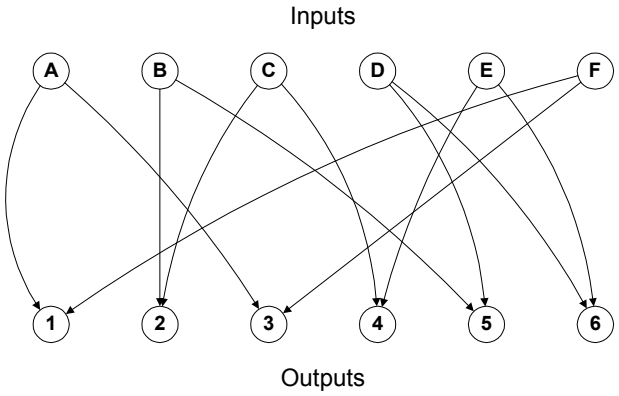
1. Draw a **graph** where each input and output port is represented as a **node**.

2. Partition the graph so that all of the input ports are in one **partition** and all the output ports are in the other.

3. Draw an **edge** from an input node to an output node if there is a connection from the corresponding input port to the corresponding output port.

This results in a **bipartite graph** where each node has a maximum degree of 9720 (the total number of possible connections from/to a port). A subset of this problem (6 nodes, 2 timeslots) is illustrated in Figure 29. We want to assign the edges (connections) to timeslots such that no coincident edges are assigned to the same timeslot. Notice that a solution to the problem consists of a permutation (or partial permutation) mapping of input nodes onto output nodes for each of the timeslots. These permutation mappings correspond to one set of switch settings for the NSE-8G devices.

Figure 30 Example Graph



12.13.4 Unicast

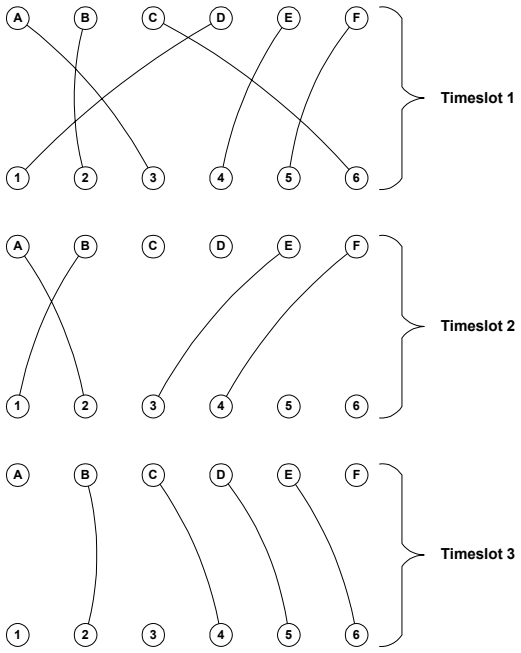
Scheduling unicast connections through the NSE-8G is a relatively simple problem: given *n* input ports, *n* output ports, *m* time slots and a guarantee that no port is oversubscribed, schedule the transfer of all input slots to output slots. This solution uses the time slot interchange on the SBS chips to schedule the flow of inputs to outputs through the NSE-8G fabric with no collisions.

Unicast connections have a perfect solution.

Example

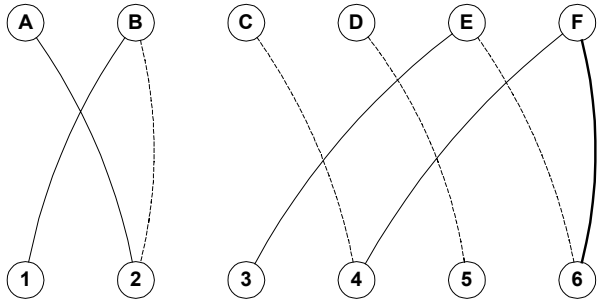
The algorithm is illustrated using an example with 3 timeslots and 6 input/output nodes. The original configuration is shown in Figure 30. The new connection originates at input node **F**, and terminates at output node **6**. This is edge **(F→6)** in the bipartite graph.

Figure 31 Example Problem



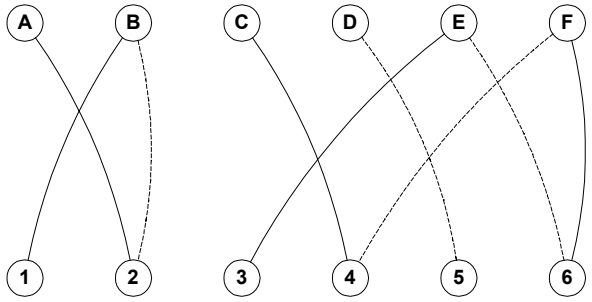
Input node F is available on timeslot 3 and output node 6 is available on timeslot 2. Merging these two timeslots and adding the edge **(F→6)** results in the graph shown in Figure 31. In this graph, the edges assigned to timeslot 3 are shown as dotted lines. The edge **(F→6)** is shown in bold.

Figure 32 Merged Graph



There are 3 maximal length paths in the merged graph, (A→2→B→1), (D→5), and (C→4→**F→6**→E→3). The last path mentioned requires re-labeling. If we start with edge (C→4) and traverse the path, alternately labeling with timeslot 2 and 3, we get the graph in Figure 32. The timeslot labeling in this graph replaces timeslots 2 and 3 in the original graph (and schedule).

Figure 33 Relabeled Graph



12.13.5 Experimental Results

The performance of PMC-Sierra's Open Path Algorithm has been studied by implementing it in C++ and running extensive random connection tests. Tests for NSE/SBS applications of this algorithm used a single NSE-8G connected to 12 SBSs, each carrying a full complement of DS0 connection. Many runs were completed in which an initially unloaded switch is presented with a sequence of random call establishment requests up to the point of 100% switching loads. These runs were carried out on a 600 MHz Alpha running Linux. In all of these runs, no otctet open path search took longer than 10μs, thus supporting up to 100,000[1] DS0 call establishments per second. T1s and other aggregates require the establishment of multiple octet open paths; complete T1s can be established at about 3,700 T1/sec. The reasons for this surprisingly good performance are explained in the separate open path algorithm document. It is our opinion that these rates are sufficiently high that the call establishment algorithm should not be a bottleneck in any application of the NSE/SBS, and that this rate is sufficiently high to permit the NSE/SBS to be used for PSTN call establishment rates (up to 100,000 calls/sec in a switch supporting 96,768 full-duplex calls, with the switching core implemented in 1 NSE-8G and 12 SBS chips).

12.13.6 Multicast

Scheduling general multicast connections is an entirely different class of problem. With unrestricted multicast, the underlying architecture is non-blocking up to capacity dictated by the number of slots in a frame, but finding the non-blocking schedule is NP-hard. There is no polynomial time running algorithm known to solve this class of problem.

There are two approaches to solving the multicast problem:

- Heuristic algorithms that have statistical probability of success for simple versions of the problem; (and)
- Restricted multicast, where the form of restriction provides a means to solve the scheduling problem.

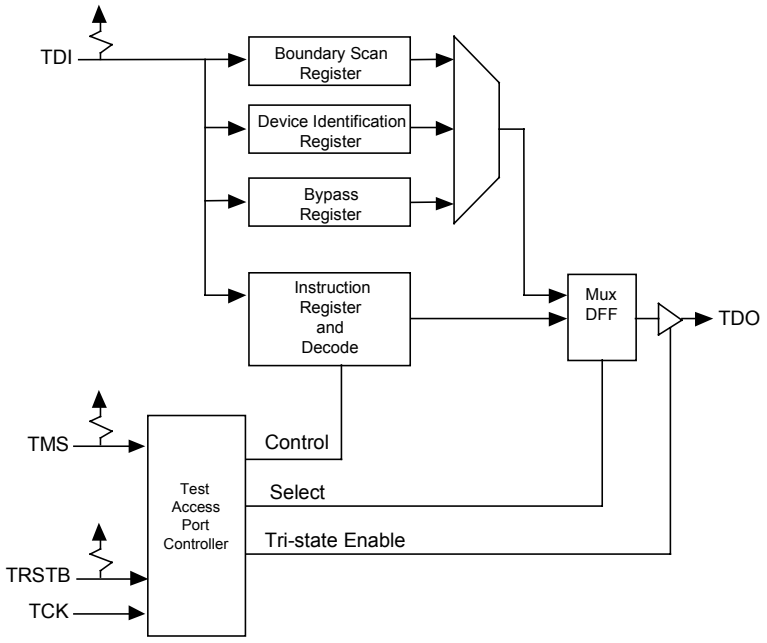
[1] This ignores inband or uP to NSE limitations.

The general multicast problem is not considered in this document. See the PMC NSE-8G documentation for descriptions of the use of multicast in a protection switching schemes; the same concepts apply to NSE/SBS fabrics.

12.14 JTAG Support

The NSE-8G supports the IEEE Boundary Scan Specification as described in the IEEE 1149.1 standards. The Test Access Port (TAP) consists of the five standard pins, TRSTB, TCK, TMS, TDI and TDO used to control the TAP controller and the boundary scan registers. The TRSTB input is the active-low reset signal used to reset the TAP controller. TCK is the test clock used to sample data on input, TDI and to output data on output, TDO. The TMS input is used to direct the TAP controller through its states. The basic boundary scan architecture is shown below.

Figure 34 Boundary Scan Architecture



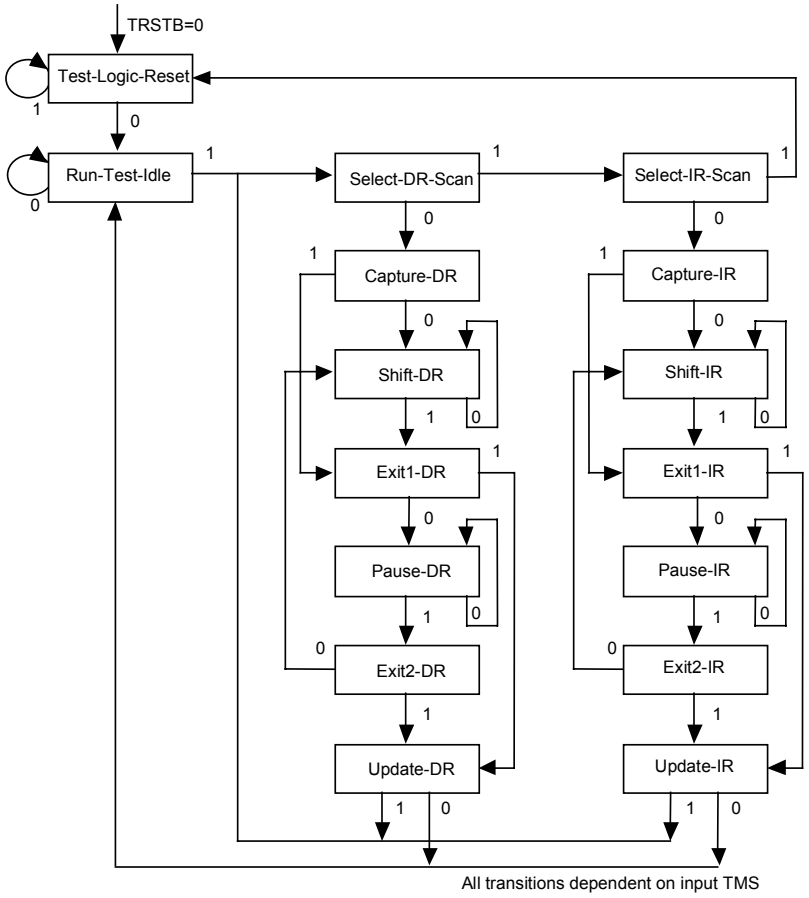
The boundary scan architecture consists of a TAP controller, an instruction-register with instruction-decode, a bypass register, a device identification register and a boundary scan register. The TAP controller interprets the TMS input and generates control signals to load the instruction and data registers. The instruction register with instruction decode block is used to select the test to be executed and/or the register to be accessed. The bypass register offers a single-bit delay from primary input, TDI to primary output, TDO. The device identification register contains the device identification code.

The boundary scan register allows testing of board inter-connectivity. The boundary scan register consists of a shift register place in series with device inputs and outputs. Using the boundary scan register, all digital inputs can be sampled and shifted out on primary output, TDO. In addition, patterns can be shifted in on primary input, TDI and forced onto all digital outputs.

12.14.1 TAP Controller

The TAP controller is a synchronous finite state machine clocked by the rising edge of primary input, TCK. All state transitions are controlled using primary input, TMS. The finite state machine is described below.

Figure 35 TAP Controller Finite State Machine



12.14.2 States

Test-Logic-Reset

The test logic reset state is used to disable the TAP logic when the device is in normal mode operation. The state is entered asynchronously by asserting input, TRSTB. The state is entered synchronously regardless of the current TAP controller state by forcing input, TMS high for 5 TCK clock cycles. While in this state, the instruction register is set to the IDCODE instruction.

Run-Test-Idle

The run test/idle state is used to execute tests.

Capture-DR

The capture data register state is used to load parallel data into the test data registers selected by the current instruction. If the selected register does not allow parallel loads or no loading is required by the current instruction, the test register maintains its value. Loading occurs on the rising edge of TCK.

Shift-DR

The shift data register state is used to shift the selected test data registers by one stage. Shifting is from MSB to LSB and occurs on the rising edge of TCK.

Update-DR

The update data register state is used to load a test register's parallel output latch. In general, the output latches are used to control the device. For example, for the EXTEST instruction, the boundary scan test register's parallel output latches are used to control the device's outputs. The parallel output latches are updated on the falling edge of TCK.

Capture-IR

The capture instruction register state is used to load the instruction register with a fixed instruction. The load occurs on the rising edge of TCK.

Shift-IR

The shift instruction register state is used to shift both the instruction register and the selected test data registers by one stage. Shifting is from MSB to LSB and occurs on the rising edge of TCK.

Update-IR

The update instruction register state is used to load a new instruction into the instruction register. The new instruction must be scanned in using the Shift-IR state. The load occurs on the falling edge of TCK.

The Pause-DR and Pause-IR states are provided to allow shifting through the test data and/or instruction registers to be momentarily paused.

Boundary Scan Instructions

The following is a description of the standard instructions. Each instruction selects a serial test data register path between input, TDI and output, TDO.

12.14.3 Instructions

BYPASS

The bypass instruction shifts data from input, TDI to output, TDO with one TCK clock period delay. The instruction is used to bypass the device.

EXTEST

The external test instruction allows testing of the interconnection to other devices. When the current instruction is the EXTEST instruction, the boundary scan register is place between input, TDI and output, TDO. Primary device inputs can be sampled by loading the boundary scan register using the Capture-DR state. The sampled values can then be viewed by shifting the boundary scan register using the Shift-DR state. Primary device outputs can be controlled by loading patterns shifted in through input TDI into the boundary scan register using the Update-DR state.

SAMPLE

The sample instruction samples all the device inputs and outputs. For this instruction, the boundary scan register is placed between TDI and TDO. Primary device inputs and outputs can be sampled by loading the boundary scan register using the Capture-DR state. The sampled values can then be viewed by shifting the boundary scan register using the Shift-DR state.

IDCODE

The identification instruction is used to connect the identification register between TDI and TDO. The device's identification code can then be shifted out using the Shift-DR state.

STCTEST

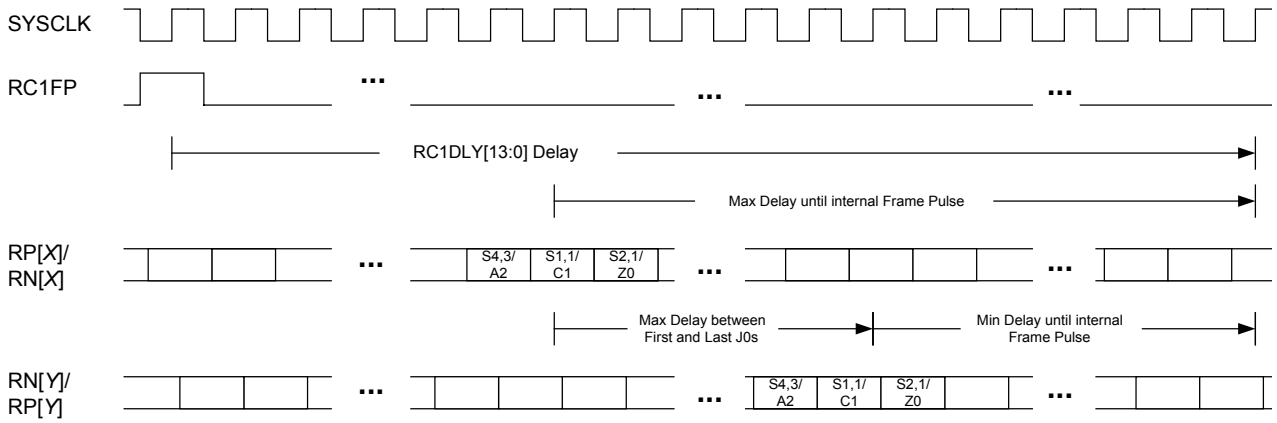
The single transport chain instruction is used to test out the TAP controller and the boundary scan register during production test. When this instruction is the current instruction, the boundary scan register is connected between TDI and TDO. During the Capture-DR state, the device identification code is loaded into the boundary scan register. The code can then be shifted out output, TDO using the Shift-DR state.

13 Functional Timing

13.1 Receive Interface Timing

Figure 35 below, shows the relative timing of the receive interface. The LVDS links carry SONET/SDH frame octets that are encoded in 8B/10B characters. Frame boundaries, justification events and alarm conditions are encoded in special control characters. The upstream devices sourcing the links share a common clock and have a common transport frame alignment that is synchronized by the Receive Serial Interface Frame Pulse signal (RC1FP). Due to phase noise of clock multiplication circuits and backplane routing or cable length discrepancies, the links will not phase aligned to each other but are frequency locked. The delay from RC1FP being sampled high to the first and last C1 character is shown in Figure 35 In this example, the first C1 is delivered on link RN[X]/RP[X]. The delay to the last C1 represents the time when the all the links have delivered their C1 character. In the example below, link RN[Y]/RP[Y] is shown to be the slowest. *The minimum value for the internal programmable delay (RC1DLY[13:0]) is the delay through the SBS[2] plus 15. The maximum value is the delay through the SBS plus 31. Consequently, the external system must ensure that the relative delays between all the receive LVDS links be less than 16 bytes.* The relative phases of the links in Figure 35 are shown for illustrative purposes only. Links may have different delays relative to other links than what is shown.

Figure 36 Receive Interface Timing



[2] This delay will be either one frame (9720 clock cycles) or one row (1080 clock cycles) depending on the mode employed.

13.2 Transmit Interface Timing

Figure 36 below shows the delay from assertion of RC1FP to the transmit serial data links. Due to the presence of FIFOs in the data path, the delay to the various links can differ by up to eight cycles. The minimum delay (RC1DLY + 43 SYSCLK cycles) is shown to be incurred by one of the transmit protect serial data links (TP[X]/TN[X]). The maximum delay (RC1DLY + 51 cycles) is shown to be incurred by one of the transmit auxiliary serial data links (TP[Y]/TN[Y]). The suggested setting for TC1DLY results in a TC1FP pulse at the time at which all the transmit serial links have transmitted their respective C1 characters. The maximum delay from RC1FP to the transmission of a C1 pulse is RC1DLY + 52 cycles. Therefore the suggested setting for TC1DLY is RC1DLY+ 52. Figure 36 shows the timing of TC1FP with the suggested setting for TC1DLY. The relative phases of the links in Figure 36 are shown for illustrative purposes only. Links may have different delays than what is shown.

Figure 37 Transmit Interface Timing

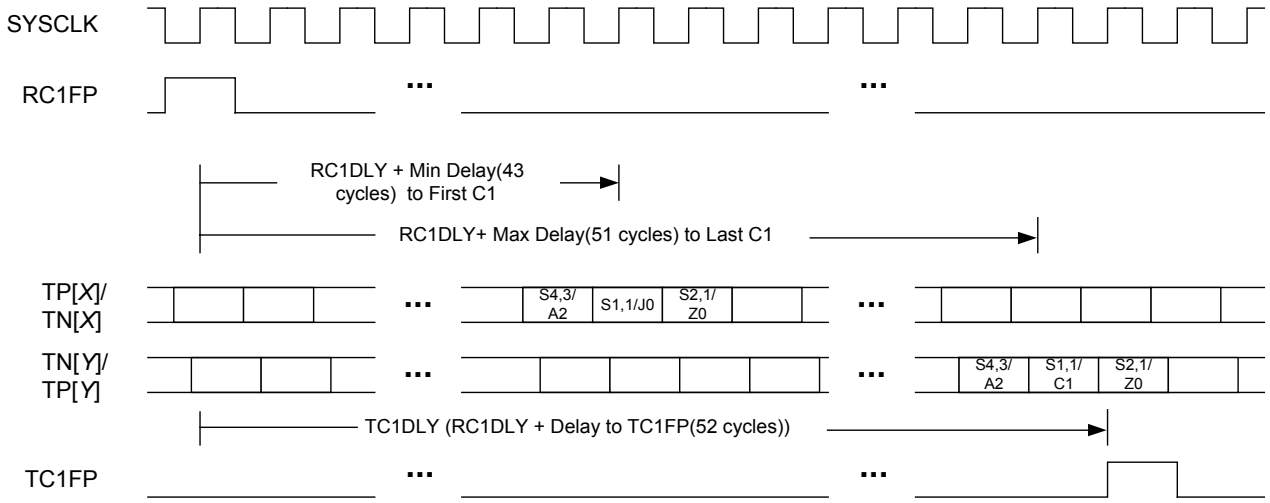
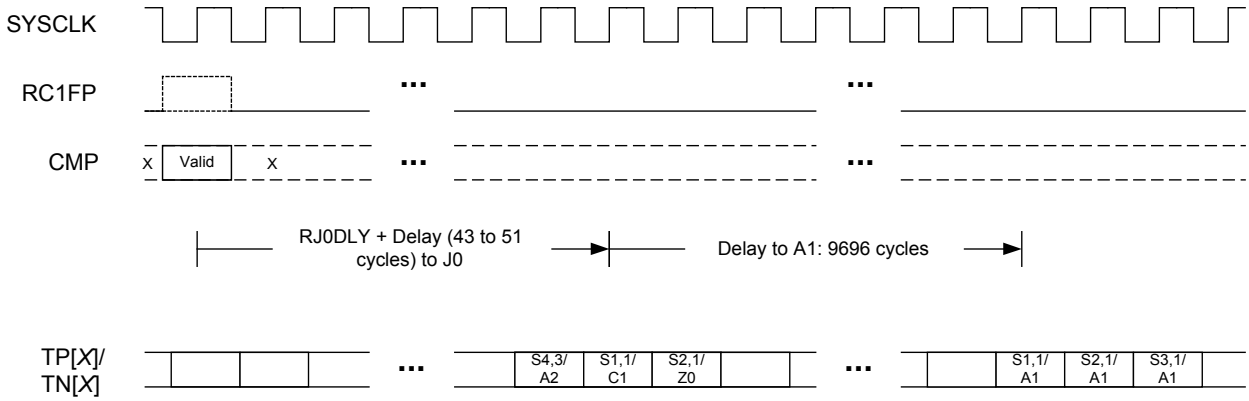


Figure 37 below shows the delay from CMP to the transmit serial data links. CMP is valid only at the RC1FP pulse time, whether RC1FP is pulsed or not. It is ignored at other locations in the transport frame. A change in value to the connection memory page signal (CMP) results in changing the active switch settings. Given that CMP is sampled on the RC1FP pulse time, the first data that is switched according to the newly selected connection memory page are the first A1 bytes of the frame following the C1 bytes transmitted by the NSE-8G before offset RC1DLY + 52 cycles. In more absolute terms, the first A1s transmitted by the NSE-8G between offset RC1DLY + 43 + 9696 cycles and RC1DLY + 51 + 9696 cycles, represent the first data switched according the connection memory page selected by CMP at the RC1FP pulse time.

Figure 38 CMP Timing



Note: RC1FP may not occur every frame – it may occur every 1, 4 or 48 frames. RC1FP synchronizes a 9720 count flywheel counter. The terminal count of this counter is used as an internal substitute for RC1FP. In this way the CMP signal is always sampled at the C1 position regardless of the RC1FP presence or absence at the C1 position.

14 Absolute Maximum Ratings

Maximum rating are the worst case limits that the device can withstand without sustaining permanent damage. They are not indicative of normal mode operation conditions.

Table 15 Absolute Maximum Ratings

Ambient Temperature under Bias	-40°C to +85°C
Storage Temperature	-40°C to +125°C
1.8V Supply Voltage (VDDI, AVDL)	-0.3V to +4.6V
3.3V Supply Voltage (VDDO, AVDH, CSU_AVDH)	-0.3 to +4.6V
Input pad tolerance	-2V < VDDO < +2V for 10ns, 100mA max
Output pad overshoot limits	-2V < VDDO < +2V for 10ns, 20mA max
Voltage on Any Digital Pin	-0.3V to VDDO+0.3V
Voltage on LVDS Pin	-0.3 V to AVDH + 0.3V
Static Discharge Voltage	±1000 V
Latch-Up Current on RN[I], RP[I], TN[I], TP[I] pins	±90 mA
Latch-Up Current on RESK pin	±50 mA
Latch-Up Current	±100 mA except RN[I], RP[I], TN[I], TP[I], and RESK
DC Input Current	±20 mA
Lead Temperature	+300°C
Absolute Maximum Junction Temperature	+150°C

Notes on Power Supplies

1. When powering up the NSE, the following power supply sequence should be observed
   VDDO, AVDH, CSU_AVDH
   VDDI, AVDL

   Powering down should be the reverse.

15 D.C. Characteristics

T_A = -40°C to +85°C, V_{DDO}3.3 V ±5%, V_{DDI}=1.8 V ±5%

(Typical Conditions: T_C = 25°C, V_{DDO}= 3.3 V, V_{DDI}= 1.8 V)

Table 16 D.C Characteristics

Symbol	Parameter	Min	Typ	Max	Units	Conditions
VDDI	Power Supply at 1.8V	1.66	1.8	1.94	Volts	
VDDO	Power Supply at 3.3V	3.14	3.3	3.47	Volts	
V_{IL}	Input Low Voltage	0	TBD	0.8	Volts	Guaranteed Input Low Voltage.
V_{IH}	Input High Voltage	2.0	TBD		Volts	Guaranteed Input High Voltage.
V_{OL}	Output or Bi-directional Low Voltage		TBD	0.4	Volts	Guaranteed output Low Voltage at VDD=2.97V and I_{OL}=maximum rated for pad.
V_{OH}	Output or Bi-directional High Voltage	2.4	TBD		Volts	Guaranteed output High Voltage at VDD=2.97V and I_{OH}=maximum rated current for pad.
V_{T+}	Reset Input High Voltage	2.2			Volts	Applies to RSTB and TRSTB only.
V_{T-}	Reset Input Low Voltage			0.8	Volts	Applies to RSTB and TRSTB only.
V_{TH}	Reset Input Hysteresis Voltage		TBD		Volts	Applies to RSTB and TRSTB only.
I_{ILPU}	Input Low Current	-200	-50	-4	µA	V_{IL} = GND. Notes 1 and 3.
I_{IHPU}	Input High Current	-10	0	+10	µA	V_{IH} = V_{DD}. Notes 1 and 3.
I_{IL}	Input Low Current	-10	0	+10	µA	V_{IL} = GND. Notes 2 and 3.
I_{IH}	Input High Current	-10	0	+10	µA	V_{IH} = V_{DD}. Notes 2 and 3.
C_{IN}	Input Capacitance		5		PF	t_A=25°C, f = 1 MHz
C_{OUT}	Output Capacitance		5		PF	t_A=25°C, f = 1 MHz

Symbol	Parameter	Min	Typ	Max	Units	Conditions
C_{IO}	Bi-directional Capacitance		5		PF	t_A=25°C, f = 1 MHz
I_{DDOP1}	Operating Current			TBD	MA	V_{DD} = 3.3V, Outputs Unloaded
V_{ICM}	LVDS Input Common-Mode Range	0		2.4	V	
$\|V_{IDM}\|$	LVDS Input Differential Sensitivity			100	mV	
R_{IN}	LVDS Differential Input Impedance	85	100	115	Ω	
V_{LOH}	LVDS Output Voltage high		1375	1475	mV	R_{LOAD}=100Ω ±1%
V_{LOL}	LVDS Output Voltage low	925	1025		mV	R_{LOAD}=100Ω ±1%
V_{ODM}	LVDS Output Differential Voltage	300	350	400	mV	R_{LOAD}=100Ω ±1%
V_{OCM}	LVDS Output Common-Mode Voltage	1125	1200	1275	mV	R_{LOAD}=100Ω ±1%
R_O	LVDS Output Impedance, Differential	85	110	115	Ω	
$\|\Delta V_{ODM}\|$	Change in $\|V_{ODM}\|$ between “0” and “1”			25	mV	R_{LOAD}=100Ω ±1%
ΔV_{OCM}	Change in V_{OCM} between “0” and “1”			25	mV	R_{LOAD}=100Ω ±1%
I_{SP}, I_{SN}	LVDS Short-Circuit Output Current			10	mA	Drivers shorted to ground
I_{SPN}	LVDS Short-Circuit Output Current			10	mA	Drivers shorted together

Notes on D.C. Characteristics:

1. Input pin or bi-directional pin with internal pull-up resistor.
2. Input pin or bi-directional pin without internal pull-up resistor
3. Negative currents flow into the device (sinking), positive currents flow out of the device (sourcing).

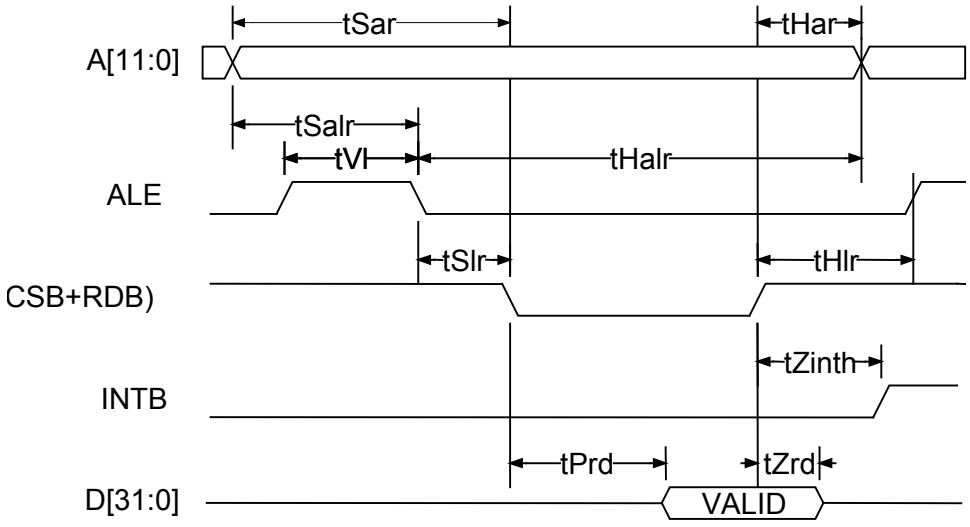
16 Microprocessor Interface Timing Characteristics

(T_C = -40°C to +85°C, V_{DDO}= 3.3V ± 5%)

Table 17 Microprocessor Interface Read Access

Symbol	Parameter	Min	Max	Units
tS_{AR}	Address to Valid Read Set-up Time	10		ns
tH_{AR}	Address to Valid Read Hold Time	5		ns
tS_{ALR}	Address to Latch Set-up Time	10		ns
tH_{ALR}	Address to Latch Hold Time	10		ns
tV_L	Valid Latch Pulse Width	5		ns
tS_{LR}	Latch to Read Set-up	0		ns
tH_{LR}	Latch to Read Hold	5		ns
tP_{RD}	Valid Read to Valid Data Propagation Delay		30	ns
tZ_{RD}	Valid Read Negated to Output Tri-state		20	ns
tZ_{INTH}	Valid Read Negated to INTB High		50	ns

Figure 39 Microprocessor Interface Read Timing



Notes on Microprocessor Interface Read Timing:

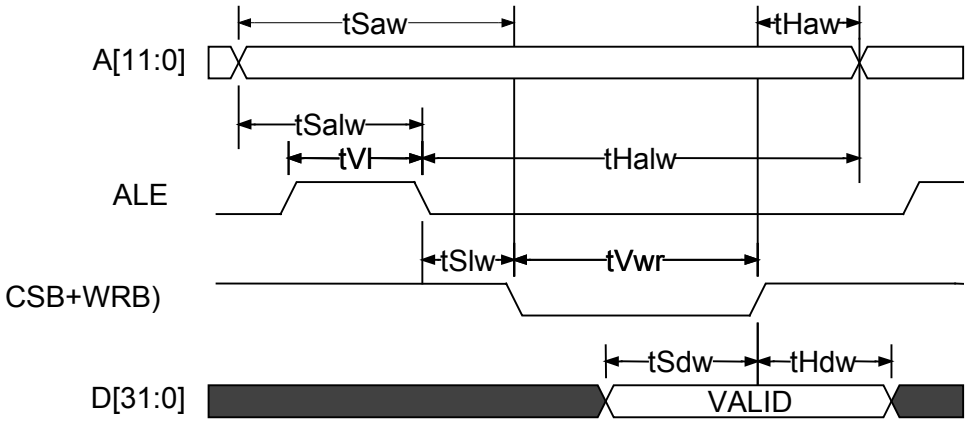
1. Output propagation delay time is the time in nanoseconds from the 1.4 V point of the reference signal to the 1.4 V point of the output.

2. Maximum output propagation delays are measured with a 100 pF load on the Microprocessor Interface data bus, (D[15:0]).
3. A valid read cycle is defined as a logical OR of the CSB and the RDB signals.
4. In non-multiplexed address/data bus architectures, ALE should be held high so parameters tS_{ALR}, tH_{ALR}, tV_L, tS_{LR}, and tH_{LR} are not applicable.
5. Parameter tH_{AR} is not applicable if address latching is used.
6. When a set-up time is specified between an input and a clock, the set-up time is the time in nanoseconds from the 1.4 V point of the input to the 1.4 V point of the clock.
7. When a hold time is specified between an input and a clock, the hold time is the time in nanoseconds from the 1.4 V point of the input to the 1.4 V point of the clock.

Table 18 Microprocessor Interface Write Access

Symbol	Parameter	Min	Max	Units
tS_{AW}	Address to Valid Write Set-up Time	10		ns
tS_{DW}	Data to Valid Write Set-up Time	20		ns
tS_{ALW}	Address to Latch Set-up Time	10		ns
tH_{ALW}	Address to Latch Hold Time	10		ns
tV_{L}	Valid Latch Pulse Width	5		ns
tS_{LW}	Latch to Write Set-up	0		ns
tH_{LW}	Latch to Write Hold	5		ns
tH_{DW}	Data to Valid Write Hold Time	5		ns
tH_{AW}	Address to Valid Write Hold Time	5		ns
tV_{WR}	Valid Write Pulse Width	40		ns

Figure 40 Microprocessor Interface Write Timing



Notes on Microprocessor Interface Write Timing:

1. A valid write cycle is defined as a logical OR of the CSB and the WRB signals.
2. In non-multiplexed address/data bus architectures, ALE should be held high so parameters tS_{ALW}, tH_{ALW}, tV_{L}, tS_{LW}, and tH_{LW} are not applicable.
3. Parameter tH_{AW} is not applicable if address latching is used.
4. When a set-up time is specified between an input and a clock, the set-up time is the time in nanoseconds from the 1.4 V point of the input to the 1.4 V point of the clock.
5. When a hold time is specified between an input and a clock, the hold time is the time in nanoseconds from the 1.4 V point of the input to the 1.4 V point of the clock.

17 A.C. Timing Characteristics

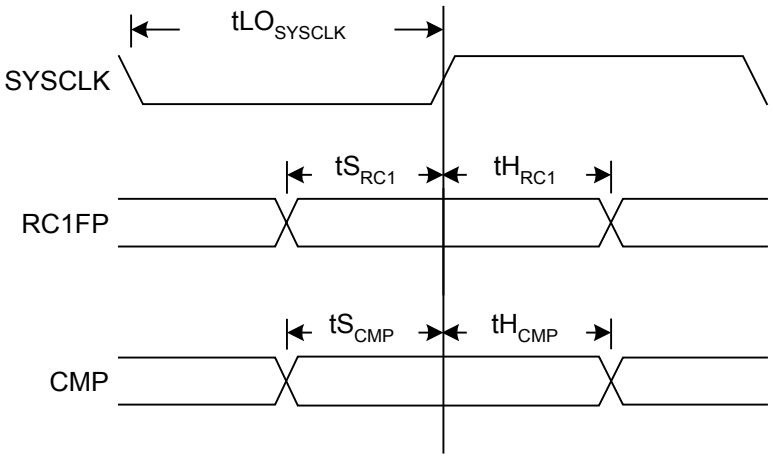
(T_A = -40°C to +85°C, V_{DDO}= 3.3 V ±5%, V_{DDI}= 1.8 V ±5%)

17.1 Input Timing

Table 19 NSE-8G Input Timing (Figure 40)

Symbol	Description	Min	Max	Units
F_{SYSCLK}	SYSCLK Frequency (nominally 77.76 MHz)	-50	+50	ppm
THI_{SYSCLK}	SYSCLK High Pulse Width	5		ns
TLO_{SYSCLK}	SYSCLK Low Pulse Width	5		ns
TS_{CMP}	CMP Set-Up Time	3		ns
TH_{CMP}	CMP Hold Time	0		ns
TS_{RC1}	RC1FP Set-Up Time	3		ns
TH_{RC1}	RC1FP Hold Time	0		ns

Figure 41 NSE-8G Input Timing



Notes on Input Timing:

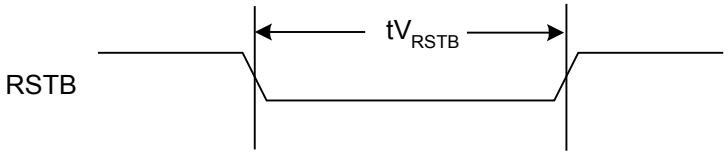
1. When a set-up time is specified between an input and a clock, the set-up time is the time in nanoseconds from the 1.4 V point of the input to the 1.4 V point of the clock.
2. When a hold time is specified between an input and a clock, the hold time is the time in nanoseconds from the 1.4 V point of the clock to the 1.4 V point of the input.

1.1 Reset Timing

Table 20 RSTB Timing (Figure 41)

Symbol	Description	Min	Max	Units
tV_{RSTB}	RSTB Pulse Width	100		ns

Figure 42 RSTB Timing



17.2 Serial SBI Bus Interface

Table 21 Serial SBI Bus Interface

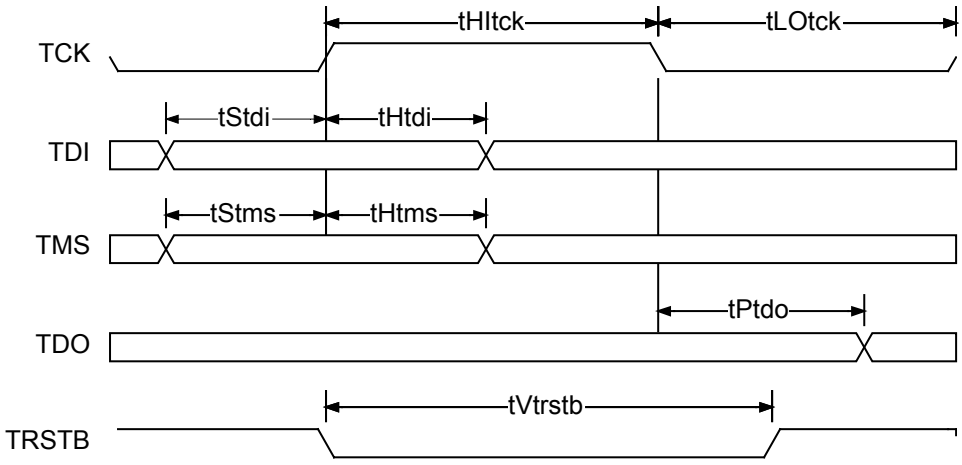
Symbol	Description	Min	Typical	Max	Units
F_{RLVDS}	RP[11:0], RN[11:0] Bit Rate	$10f_{SYSCLK}$ -100ppm	$10f_{SYSCLK}$	$10f_{SYSCLK}$+ 100ppm	MB/s
T_{FALL}	V_{ODM} fall time, 80%-20%, (R_{LOAD}=100Ω ±1%)	200	300	400	ps
T_{RISE}	V_{ODM} rise time, 20%-80%, (R_{LOAD}=100Ω ±1%)	200	300	400	ps
T_{SKEW}	Differential Skew			50	ps

17.3 JTAG Port Interface

Table 22 JTAG Port Interface (Figure 42)

Symbol	Description	Min	Max	Units
f_{TCK}	TCK Frequency		4	MHz
tHI_{TCK}	TCK HI Pulse Width	100		ns
tHI_{TCK}	TCK LO Pulse Width	100		ns
tS_{TMS}	TMS Set-up time to TCK	25		ns
tH_{TMS}	TMS Hold time to TCK	25		ns
tS_{TDI}	TDI Set-up time to TCK	25		ns
tH_{TDI}	TDI Hold time to TCK	25		ns
tP_{TDO}	TCK Low to TDO Valid	2	25	ns
tV_{TRSTB}	TRSTB Pulse Width	100		ns

Figure 43 JTAG Port Interface Timing



18 Ordering and Thermal Information

18.1 Packaging Information

Part No.	Description
PM8621-BIAP	480 Uni Ball Grid Array (UBGA) Package

18.2 Thermal Information

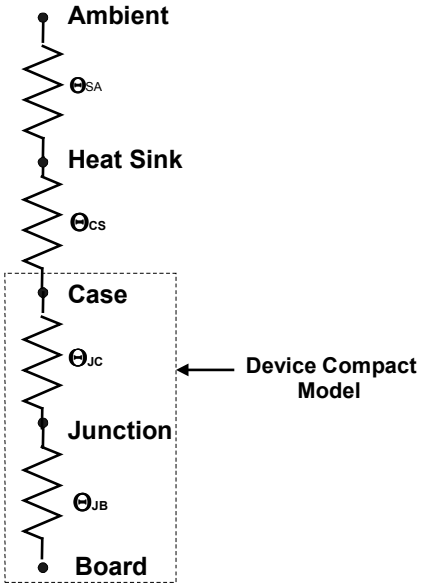
In full operation (10–12 ports), the NSE-8G is designed to operate over a wide temperature range when used with a heat sink with a worst case Junction to case Tj (Tjc) of 1 °C/W and is suited for industrial applications such as outside plant equipment

Maximum long-term operating junction temperature to ensure adequate long-term life	105 ^{0}C
Maximum junction temperature for short-term excursions with guaranteed continued functional performance.[1] This condition will typically be reached when local ambient reaches 70 Deg C.	125 ^{0}C
Minimum ambient temperature	-40^{0}C

Device Compact Model[2]	
Θ_{JC} (^{0}C/W)	# 1
Θ_{JB} (^{0}C/W)	# 4

Operating power is dissipated in package (watts) at worst case power supply	
Power (watts)	2.56

Θ_{SA} and Θ_{CS} required for long-term operation [3]	
Θ_{SA} + Θ_{CS} (^{0}C/W)[4]	#5



Notes

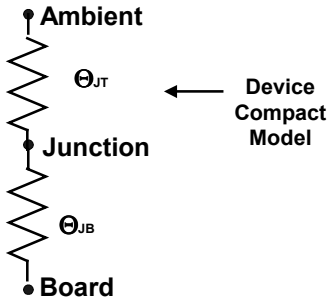
1. Short-term is understood as the definition stated in Telcordia Generic Requirements GR-63-Core
2. $\Theta_{JC,}$ the junction-to-case thermal resistance is a measured nominal value + 2 sigma. Θ_{JB}, the junction-to-board thermal resistance is obtained by simulating conditions described in JEDEC Standard, JESD 51
3. Θ_{SA} is the thermal resistance of the heat sink to ambient. Θ_{CS} is the thermal resistance of the heat sink attached material
4. The actual Θ_{SA} required may vary according to the air speed at the location of the device in the system with all the components in place

In partial operation (less than 10 ports), this product is designed to operate over a wide temperature range and is suited for industrial applications such as outside plant equipment.

Maximum long-term operating junction temperature to ensure adequate long-term life	105 ^{0}C
Maximum junction temperature for short-term excursions with guaranteed continued functional performance[1] This condition will typically be reached when local ambient reaches 85 Deg C.	125 ^{0}C
Minimum ambient temperature	-40 ^{0}C

Thermal Resistance vs Air Flow[2]			
Airflow	Natural Convection	200 LFM	400 LFM
Θ_{JA} (^{0}C/W)	13.5	12.2	11.8

Device Compact Model[3]	
Θ_{JT} (^{0}C/W)	# 1
Θ_{JB} (^{0}C/W)	# 4



Operating power is dissipated in package (watts) at worst case power supply	
Power (watts)	2.56

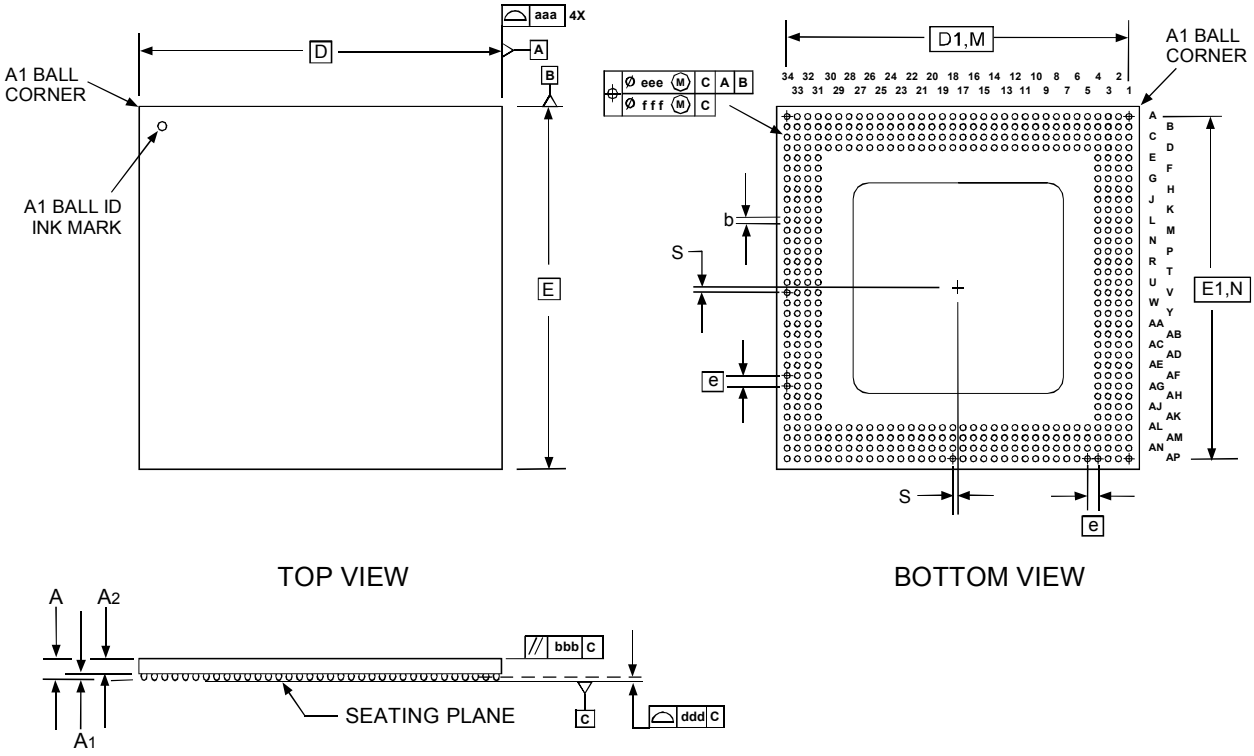
Notes

1. Short-term is understood as the definition stated in Telcordia Generic Requirements GR-63-Core
2. Θ_{JA} , the total junction to ambient thermal resistance as measured according to JEDEC Standard JESD51 (2S2P)
3. Θ_{JB}, the junction-to-board thermal resistance and $\Theta_{JT,}$ the residual junction to ambient thermal resistance are obtained by simulating conditions described in JEDEC Standard, JESD 15-8

19 Mechanical Information

The NSE-8G is packaged in a 480 UBGA package.

480 Pin UBGA –35 mm x 35 mm BODY - (B SUFFIX)



NOTES: 1) ALL DIMENSIONS IN MILLIMETER.
2) DIMENSION aaa DENOTES PACKAGE BODY PROFILE.
3) DIMENSION bbb DENOTES PARALLEL.
4) DIMENSION ccc DENOTES FLATNESS.
5) DIMENSION ddd DENOTES COPLANARITY.
6) DIAMETER OF SOLDER MASK OPENING IS 0.45 +/- 0.025 MM (SMD).

PACKAGE TYPE : 480 THERMALLY ENHANCED BALL GRID ARRAY - UBGA																
BODY SIZE : 35 x 35 x 1.47 MM																
Dim.	A	A_1	A_2	D	D_1	E	E_1	M,N	b	e	aaa	bbb	ddd	eee	f f f	S
Min.	1.32	0.40	0.92	-	-	-	-	-	0.50	-	-	-	-	-	-	-
Nom.	1.47	0.50	0.97	35.00 BSC	33.00 BSC	35.00 BSC	33.00 BSC	34x34	0.63	1.00 BSC	-	-	-	-	-	-
Max.	1.62	0.60	1.02	-	-	-	-	-	0.70	-	0.20	0.25	0.20	0.30	0.10	0.05

Notes