- **Single-Chip Segmentation and Reassembly Device (SAR) for Full-Duplex ATM-Adaptation-Layer (AAL) Processing**
- **Integrated 64-Bit Peripheral-Component Interface for Transferring Data and Control Information for Packet Segmentation and Packet Reassembly**
- **Provides Complete Encapsulation and Termination of AAL5 Packets**
- **Supports Both Transmit and Receive Buffer Chaining (Buffer Scatter/Gather)**
- **Provides for Early Segmentation of a Transmit Packet by Beginning the Segmentation Process Once a Transmit Buffer Is Available Instead of Waiting for the Entire Packet to Be Buffered**
- **Provides Full VPI/VCI Support and Allows for the Simultaneous Segmentation of 1023 Packets and the Simultaneous Reassembly of 30720 Packets**
- **Byte-Wide Cell Interface Is Full Duplex and Compliant to the ATM-Forum UTOPIA Document**
- **Cell Interface Can Be Programmed to Operate as Either a Physical (PHY)-Layer Interface or as a ATM-Layer Interface**
- **Provides Reassembly Time-Out for an Incoming Packet**
- **Provides a High-Priority Mechanism for Transmitting Constant-Bit-Rate Traffic**
- **Provides Support for Transparent/Null AAL**

description

The TNETA1570 is an asynchronous transfer mode (ATM) segmentation and reassembly (SAR) device with a 64-bit peripheral component interconnect (PCI)-bus interface. This device incorporates ATM adaptation-layer (AAL) processing, ATM SAR processing for full-duplex operation up to STS-3c rate of 155.52 Mbit/s, and the controls for the register interface on the PHY layer. The device provides complete encapsulation and termination of AAL5 packets in hardware.

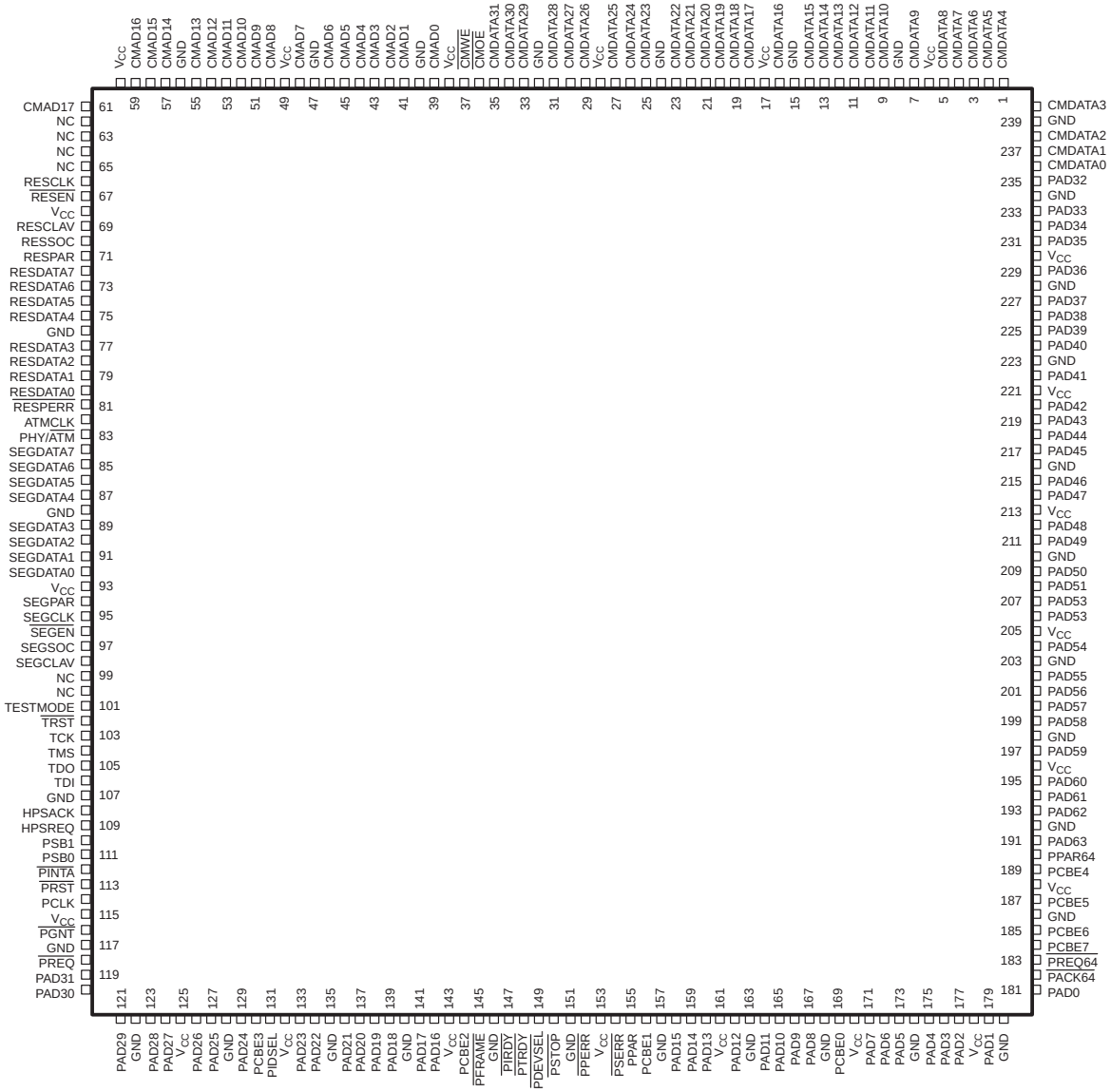
The TNETA1570 supports high-speed networking applications utilizing ATM protocols as either a backbone/backplane or desktop technology. Features include: high level of VPI/VCI support, high-priority segmentation option for constant-bit-rate traffic, early buffer segmentation, buffer scatter/gather capability, and 32-/64-bit PCI-bus support.

The device contains an integrated 32-/64-bit PCI interface for transferring data and control information. The segmentation and reassembly processes use host memory for storing packets that are transmitted or received. No local-packet memory is required. The device is capable of segmenting up to 1023 packets simultaneously and reassembling 30720 packets simultaneously. The device supports the full range of VPI/VCI values for both transmit and receive operations.

The TNETA1570 also supports two methods of transporting a transparent/null AAL used for transferring proprietary information. In addition, the device recognizes ATM-layer OAM cells and provides a mechanism for handling these cells. The device contains a full-duplex, byte-wide cell interface compliant to the ATM-Forum UTOPIA specification. The cell interface can be programmed to operate as either a PHY-layer interface or a ATM-layer interface.

The integrated PCI-host interface operates as either a 32-bit or 64-bit interface for DMA operations. The device operates as a 64-bit interface if the target device can accept 64-bit transfers; otherwise, it operates as a 32-bit interface. The PCI-host interface provides both master and slave capability and operates at a frequency up to 33 MHz. The PCI-host interface is functionally compliant to the PCI-local-bus specification revision 2.0. The TNETA1570 operation is explained in detail in the *Principles of Operation* section.

MFP PACKAGE
(TOP VIEW)



NC – No internal connection

Terminal Functions

PCI-bus interface

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION
$\overline{\text{PACK64}}$	182	I/O (3 state)	PCI acknowledge 64-bit transfer. $\overline{\text{PACK64}}$ is driven by the current target indicating the ability to support a 64-bit access. When acknowledging a 64-bit access, the assertion of $\overline{\text{PACK64}}$ is coincident with the assertion of $\overline{\text{PDEVSEL}}$. If $\overline{\text{PACK64}}$ is not asserted coincident with $\overline{\text{PDEVSEL}}$, the transaction defaults to 32 bit.
PAD31 – PAD0	119 – 121, 123 – 124, 126 – 127, 129, 133 – 134, 136 – 139, 141 – 142, 158 – 160, 162, 164 – 167, 171 – 173, 175 – 177, 179, 181	I/O (3 state)	PCI address bus and data bus. PAD31 – PAD0 are multiplexed on the same PCI terminals. During the first phase of the address phase of a transaction, PAD31 – PAD0 contain a 32-bit PHY address. This phase is the clock cycle when $\overline{\text{PFRAME}}$ is asserted. During the data phase, PAD7 – PAD0 contain the least significant byte and PAD31 – PAD24 contain the most significant byte. Write data is stable when $\overline{\text{PIRDY}}$ is asserted. Read data is stable when $\overline{\text{PTRDY}}$ is asserted. Data is transferred during those clock cycles when both $\overline{\text{PIRDY}}$ and $\overline{\text{PTRDY}}$ are asserted.
PAD63 – PAD32	191, 193 – 195, 197, 199 – 202, 204, 206 – 209, 211 – 212, 214 – 215, 217 – 220, 222, 224 – 227, 229, 231 – 233, 235	I/O (3 state)	PCI address bus and data bus. PAD63 – PAD32 are multiplexed on the same PCI terminals. During the first phase of the address phase of a transaction, PAD63 – PAD32 contain a 32-bit PHY address. This phase is the clock cycle when $\overline{\text{PFRAME}}$ is asserted. During the data phase, PAD39 – PAD32 contain the least significant byte and PAD63 – PAD56 contain the most significant byte. Write data is stable when $\overline{\text{PIRDY}}$ is asserted. Read data is stable when $\overline{\text{PTRDY}}$ is asserted. Data is transferred during those clock cycles when both $\overline{\text{PIRDY}}$ and $\overline{\text{PTRDY}}$ are asserted.
PCBE3 – PCBE0	130, 144, 156, 169	I/O (3 state)	PCI-bus command and byte enable. PCBE3 – PCBE0 lines are multiplexed on the same PCI terminals. During the address phase of a transaction, PCBE3 – PCBE0 lines define the bus command. During the data phase, PCBE3 – PCBE 0 lines define which bytes are valid.
PCBE7 – PCBE4	184 – 185, 187 , 189	I/O (3 state)	PCI-bus command and byte enable. PCBE7 – PCBE4 lines are multiplexed on the same PCI terminals. During the address phase of a transaction, PCBE7 – PCBE4 lines define the bus command. During the data phase, PCBE7 – PCBE4 lines define which bytes are valid.
PCLK	114	I	PCI clock. PCLK provides timing for all transactions on the PCI interface.
$\overline{\text{PDEVSEL}}$	149	I/O (3 state)	PCI device select. When actively driven, $\overline{\text{PDEVSEL}}$ indicates that the address of the driving device is decoded as the target of the current access. As an input, $\overline{\text{PDEVSEL}}$ indicates whether any device on the bus is selected.
$\overline{\text{PFRAME}}$	145	I/O (3 state)	PCI frame. $\overline{\text{PFRAME}}$ is driven by the current master to indicate the beginning and duration of an access. $\overline{\text{PFRAME}}$ is asserted at the beginning of the bus transaction and remains asserted during data transfer. When $\overline{\text{PFRAME}}$ is deasserted, the transaction is in the final data phase.
$\overline{\text{PGNT}}$	116	I	PCI bus grant. $\overline{\text{PGNT}}$ indicates to the agent that the arbiter has granted access to the bus. $\overline{\text{PGNT}}$ is a point-to-point signal and every master has its own.

Terminal Functions (Continued)

PCI-bus interface (continued)

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION
PIDSEL	131	I	PCI initialization and device select. PIDSEL is used as a chip select during configuration read and write transactions.
$\overline{\text{PINTA}}$	112	O (open drain)	PCI interrupt. $\overline{\text{PINTA}}$ is an interrupt request from PCI SAR.
$\overline{\text{PIRDY}}$	147	I/O (3 state)	PCI initiator ready. $\overline{\text{PIRDY}}$ indicates the initiating agent's (bus master) ability to complete the current data phase of the transaction. During a write, $\overline{\text{PIRDY}}$ indicates valid data on PAD31–PAD0. During a read, $\overline{\text{PIRDY}}$ indicates the master is prepared to accept the data. $\overline{\text{PIRDY}}$ is used with $\overline{\text{PTRDY}}$ when wait cycles are inserted until both are asserted.
PPAR	155	I/O (3 state)	PCI parity. PPAR is even parity across PAD31–PAD0 and PCBE3–PCBE0. For data phases, PPAR is valid one clock after either $\overline{\text{PIRDY}}$ is asserted on a write or $\overline{\text{PTRDY}}$ is asserted on a read. Once asserted, PPAR remains valid until one clock after the completion of the current data phase. The master drives the PPAR for address- and write-data phases, and the target drives PPAR for the read-data phase.
PPAR64	190	I/O (3 state)	PCI parity. PPAR64 is even parity across AD63–AD32 and CBE7–CBE4. For data phases, PPAR64 is valid one clock after either $\overline{\text{PIRDY}}$ is asserted on a write or $\overline{\text{PTRDY}}$ is asserted on a read. Once asserted, PPAR64 remains valid until one clock after the completion of the current data phase. The master drives the PPAR 64 for address and write-data phase, and the target drives the PPAR for read-data phases.
$\overline{\text{PPERR}}$	152	I/O (3 state)	PCI parity error. $\overline{\text{PPERR}}$ reports a data-parity error on all commands except special cycle. An agent cannot report a $\overline{\text{PPERR}}$ until it has claimed the access by $\overline{\text{PDEVSEL}}$ and completed a data phase.
$\overline{\text{PREQ}}$	118	O	PCI request. $\overline{\text{PREQ}}$ indicates to the arbiter that this agent desires use of the bus. Every master has its own $\overline{\text{PREQ}}$.
$\overline{\text{PREQ64}}$	183	I/O (3 state)	PCI request 64-bit transfer. $\overline{\text{PREQ64}}$ is driven by the current master indicating the desire for a 64-bit access. When initiating a 64-bit transaction, the assertion of $\overline{\text{PREQ64}}$ is coincident with the assertion of $\overline{\text{PFRAME}}$. If a 64-bit access is not acknowledged by $\overline{\text{PFRAME}}$, the access defaults to 32 bit. The transfer type (32 or 64 bit) is dynamically negotiated for each transaction and remains constant for the length of the transaction.
$\overline{\text{PRST}}$	113	I	PCI reset. $\overline{\text{PRST}}$ forces the PCI sequence of each device to a known state.
PSB1–PSB0	110–111	O	PCI sideband. PSB1–PSB0 lines define the size of the transfer when the TNETA1570 is the bus master. The definitions of these signals are as follows: 00: 4-byte transfer 01: 16-byte transfer (except for receive-completion ring) 10: Payload transfer 11: Transfer to receive-completion ring (16-byte transfer) PSB1–PSB0 are synchronous to the address phase of the bus-master operations by the TNETA1570. The sideband signals do not change upon a bus retry after disconnect.
$\overline{\text{PSERR}}$	154	I/O (open drain)	PCI system error. $\overline{\text{PSERR}}$ reports address-parity errors and data-parity errors on special-cycle commands.
$\overline{\text{PSTOP}}$	150	I/O (3 state)	PCI stop. $\overline{\text{PSTOP}}$ indicates that the current target is requesting the master to stop the current transaction.
$\overline{\text{PTRDY}}$	148	I/O (3 state)	PCI target ready. $\overline{\text{PTRDY}}$ indicates the target agent's (selected device) ability to complete the current data phase of the transaction. During a read, $\overline{\text{PTRDY}}$ indicates that valid data is present on PAD31–PAD0. During a write, $\overline{\text{PTRDY}}$ indicates that the target is prepared to accept data.

Terminal Functions (Continued)

cell-segmentation interface

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION	
SEGCLAV	98	O (CMOS)	Segmentation cell available	
			PHY mode	SEGCLAV ($\overline{\text{RXEMPTY}}$/RXCLAV) indicates that a complete cell is available.
			ATM mode	SEGCLAV ($\overline{\text{TXENB}}$) is active low when SEGDATA contains a valid byte.
SEGCLK	95	I (TTL)	Segmentation clock	
			PHY mode	SEGCLK (RXCLK) is used to synchronize transfers on SEGDATA. SEGCLK is sourced from the UTOPIA interface.
			ATM mode	SEGCLK (TXCLK) is used to synchronize transfers on SEGDATA. SEGCLK is sourced from the ATMCLK.
SEGDATA7–SEGDATA0	84–87, 89–92	O (CMOS)	Segmentation data	
			PHY mode	SEGDATA7–SEGDATA0 (RXDATA) is byte-wide true data that is sourced by the TNETA1570. SEGDATA7 is the MSB.
			ATM mode	SEGDATA7–SEGDATA0 (TXDATA) is byte-wide true data that is sourced by the TNETA1570. SEGDATA7 is the MSB.
$\overline{\text{SEGEN}}$	96	I (TTL)	Segmentation enable	
			PHY mode	$\overline{\text{SEGEN}}$ ($\overline{\text{RXENB}}$) indicates that a valid byte will be sent during the next clock cycle.
			ATM mode	$\overline{\text{SEGEN}}$ ($\overline{\text{TXFULL}}$/TXCLAV) indicates that at least 1-byte SEGDATA will be accepted.
SEGPAR	94	O (CMOS)	Segmentation parity	
			PHY mode	SEGPAR (RXPAR) is the odd-parity bit over SEGDATA7–SEGDATA0.
			ATM mode	SEGPAR (TXPAR) is the odd-parity bit over SEGDATA7–SEGDATA0.
SEGSOC	97	O (CMOS)	Segmentation start of cell	
			PHY mode	SEGSOC (RXSOC) is active high when SEGDATA contains the first valid byte of the cell.
			ATM mode	SEGSOC (TXSOC) is active high when SEGDATA contains the first valid byte of the cell.

high-priority segmentation (request and acknowledge)

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION
HPSREQ	109	I (TTL)	High-priority segmentation request. HPSREQ is sampled at each new segmentation opportunity and is synchronous to the PCI-bus clock. When HPSREQ is active, TNETA1570 initiates the procedure for transmitting a cell from TX DMA channel 1. To ensure that the high-priority segmentation request is processed, HPSREQ remains active until HPSACK is set low. HPSREQ is deasserted within two PCI-bus clock cycles of when HPSACK is asserted.
HPSACK	108	O (CMOS)	High-priority segmentation acknowledge. HPSACK is asserted for one PCI-bus clock cycle to acknowledge that HPSREQ is detected.

Terminal Functions (Continued)

cell-reassembly interface

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION	
RESCLAV	69	O (CMOS)	Reassembly cell available	
			PHY mode	RESCLAV ($\overline{\text{TXFULL}}$/TXCLAV) indicates that a transfer of a complete cell can be accepted.
			ATM mode	RESCLAV ($\overline{\text{RXENB}}$) indicates that a valid byte RESDATA will be sent during the next clock cycle.
RESCLK	66	I (TTL)	Reassembly clock	
			PHY mode	RESCLK (TXCLK) is used to synchronize transfers on RESDATA. ATMCLK is used as the source for RESCLK.
			ATM mode	RESCLK (RXCLK) is used to synchronize transfers on RESDATA. ATMCLK is used as the source for RESCLK.
RESDATA7–RESDATA0	72–75, 77–80	I (TTL)	Reassembly data	
			PHY mode	RESDATA7–RESDATA0 (TXDATA). RESDATA7 is the MSB.
			ATM mode	RESDATA7–RESDATA0 (RXDATA). RESDATA7 is the MSB.
$\overline{\text{RESEN}}$	67	I (TTL)	Reassembly enable	
			PHY mode	$\overline{\text{RESEN}}$ ($\overline{\text{TXENB}}$) goes active low when RESDATA contains a valid byte.
			ATM mode	$\overline{\text{RESEN}}$ ($\overline{\text{RXEMPTY}}$/RXCLAV) indicates that RESDATA will be sent during the next clock cycle.
RESPAR	71	I (TTL)	Reassembly parity	
			PHY mode	RESPAR (TXPAR) is the odd-parity bit over RESDATA7–RESDATA0.
			ATM mode	RESPAR (RXPAR) is the odd-parity bit over RESDATA7–RESDATA0.
$\overline{\text{RESPERR}}$	81	O (CMOS)	Reassembly parity error (not required by UTOPIA)	
			PHY mode	$\overline{\text{RESPERR}}$ indicates that parity error is present at the previous rising edge of RESCLK.
			ATM mode	
RESSOC	70	I (TTL)	Reassembly start of cell	
			PHY mode	RESSOC (TXSOC) is received when RESDATA contains the first valid byte of the cell.
			ATM mode	RESSOC (RXSOC) is received when RESDATA contains the first valid byte of the cell.

Terminal Functions (Continued)

control-memory interface

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION
CMAD17–CMAD0	61, 59–57, 55–50, 48, 46–41, 39	O (CMOS)	Control-memory address. CMAD17–CMAD0 contains an 18-bit physical address to the control memory.
CMDATA31–CMDATA0	35–33, 31–29, 27–25, 23–18, 16, 14–9, 7, 5–1, 240, 238–236	I (TTL) O (CMOS)	Control-memory data. CMDATA31–CMDATA0 contains 32-bit data to/from the control memory.
$\overline{\text{CMOE}}$	36	O (CMOS)	Control-memory output enable. When $\overline{\text{CMOE}}$ is active low, the address is valid and data is read into the TNETA1570 on the rising edge of $\overline{\text{CMOE}}$.
$\overline{\text{CMWE}}$	37	O (CMOS)	Control-memory write enable. When $\overline{\text{CMWE}}$ is active low, the address and data are valid.

boundary-scan interface

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION
TCK	103	I (TTL)	Test clock. TCK clocks the test-access-port (TAP) operation.
TDI	106	I (TTL)	Test data input. TDI shifts serial-test data and instructions into the device during (TAP) operation.
TDO	105	O (TTL)	Test data output. TDO shifts serial-test data and instructions out of the device during TAP operation.
TMS	104	I (TTL)	Test-mode select. TMS controls the state of the TAP controller.
$\overline{\text{TRST}}$	102	I (TTL)	Test reset. $\overline{\text{TRST}}$ asynchronously forces the TAP controller to a known state.

test signal

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION
TESTMODE	101	I (TTL)	Test mode. TESTMODE is used for device testing. Grounded for normal operation.

miscellaneous signals

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION	
PHY/$\overline{\text{ATM}}$	83	I (TTL)	PHY/$\overline{\text{ATM}}$ mode select. PHY/$\overline{\text{ATM}}$ selects ATM mode when low and the PHY mode when high.	
ATMCLK	82	O (CMOS)	ATM clock	
			PHY mode	In PHY mode, the ATMCLK is driven low. It can be left as a N/C.
			ATM mode	ATMCLK is used as the clock source to provide for data transfers/synchronization. ATMCLK is connected to SEGCLK and RESCLK in the ATM mode. The clock generated by the interface is 33 MHz (nominal) using the PCI clock.
NC	62–65, 99–100		No connection. Leave open.	

power and ground

TERMINAL NAME	TERMINAL NO.	DESCRIPTION
GND	8, 15, 24, 32, 40, 47, 56, 76, 88, 107, 117, 122, 128, 135, 140, 146, 151, 157, 163, 168, 174, 180, 186, 192, 198, 203, 210, 216, 223, 228, 234, 239	Ground
V_{CC}	6, 17, 28, 38, 49, 60, 68, 93, 115, 125, 132, 143, 153, 161, 170, 178, 188, 196, 205, 213, 221, 230	Supply voltage

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC} (see Note 1) . . . . . –0.5 V to 6 V
Input voltage range, V_I . . . . . –0.5 V to V_{CC} + 0.5 V
Output voltage range, V_O . . . . . –0.5 V to V_{CC} + 0.5 V
Input clamp current, I_{IK} (V_I < 0 or V_I > V_{CC}) (see Note 2) . . . . . ±20 mA
Output clamp current, I_{OK} (V_O < 0 or V_O > V_{CC}) (see Note 3) . . . . . ±20 mA
Operating free-air temperature range, T_A . . . . . 0°C to 70°C
Storage temperature range, T_{stg} . . . . . –65°C to 150°C

† Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. All voltage values are with respect to the GND terminals.
2. Applies for external input and bidirectional buffers
3. Applies for external output and bidirectional buffers

recommended operating conditions

				MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage			4.75	5	5.25	V
V_{IH}	High-level input voltage	CMOS	V_{CC} = 4.75 V	3.325			V
			V_{CC} = 5.25 V	3.675			
		TTL		2			
V_{IL}	Low-level input voltage	CMOS	V_{CC} = 4.75 V			0.95	V
			V_{CC} = 5.25 V			1.05	
		TTL				0.8	
T_A	Operating free-air temperature			0		70	°C

electrical characteristics over recommended operating conditions, T_A = 25°C (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage	I_{OH} = 8 mA	V_{CC}–0.8			V
		I_{OL} = 4 mA	V_{CC}–0.8			
V_{OL}	Low-level output voltage	I_{OH} = 8 mA			0.5	V
		I_{OL} = 4 mA			0.5	
I_{OZ}	High-impedance-state output current	V_I = V_{CC} or GND			±10	µA
I_{IL}	Low-level input current	V_I = GND			–1	µA
I_{IH}	High-level input current	V_I = V_{CC}			1	µA
I_{CC}	Supply current, dynamic	V_{CC} = 5.25 V, f_{clock} = 33 MHz		500		mA

timing requirements (see Note 4 and Figure 1)

NO.			MIN	MAX	UNIT
1	$t_{w(RESCLKH)}$	Pulse duration, RESCLK high	12		ns
2	$t_{w(RESCLKL)}$	Pulse duration, RESCLK low	12		ns
3	$t_{su(RESSOC)}$	Setup time, RESSOC high before RESCLK↑	10		ns
4	$t_{su(RESEN)}$	Setup time, $\overline{\text{RESEN}}$ low before RESCLK↑	10		ns
5	$t_{su(RESDATA)}$	Setup time, RESDATA7–RESDATA0 valid before RESCLK↑	10		ns
6	$t_{su(RESPAR)}$	Setup time, RESPAR valid before RESCLK↑	10		ns
7	$t_{h(RESSOC)}$	Hold time, RESSOC high after RESCLK↑	1		ns
8	$t_{h(RESEN)}$	Hold time, $\overline{\text{RESEN}}$ low after RESCLK↑	1		ns

NOTE 4: All output signals are generated on the rising edge of RESCLK.

operating characteristics (see Note 4 and Figure 1)

NO.			MIN	MAX	UNIT
9	$t_{d(RESCLAV)}$	Delay time, RESCLK ↑ to RESCLAV↑	1	20	ns
10	$t_{d(RESPERR)1}$	Delay time, RESCLK ↑ to $\overline{\text{RESPERR}}$↓	1	20	ns
11	$t_{d(RESPERR)2}$	Delay time, RESCLK ↑ to $\overline{\text{RESPERR}}$↑	1	20	ns

NOTE 4: All output signals are generated on the rising edge of RESCLK.

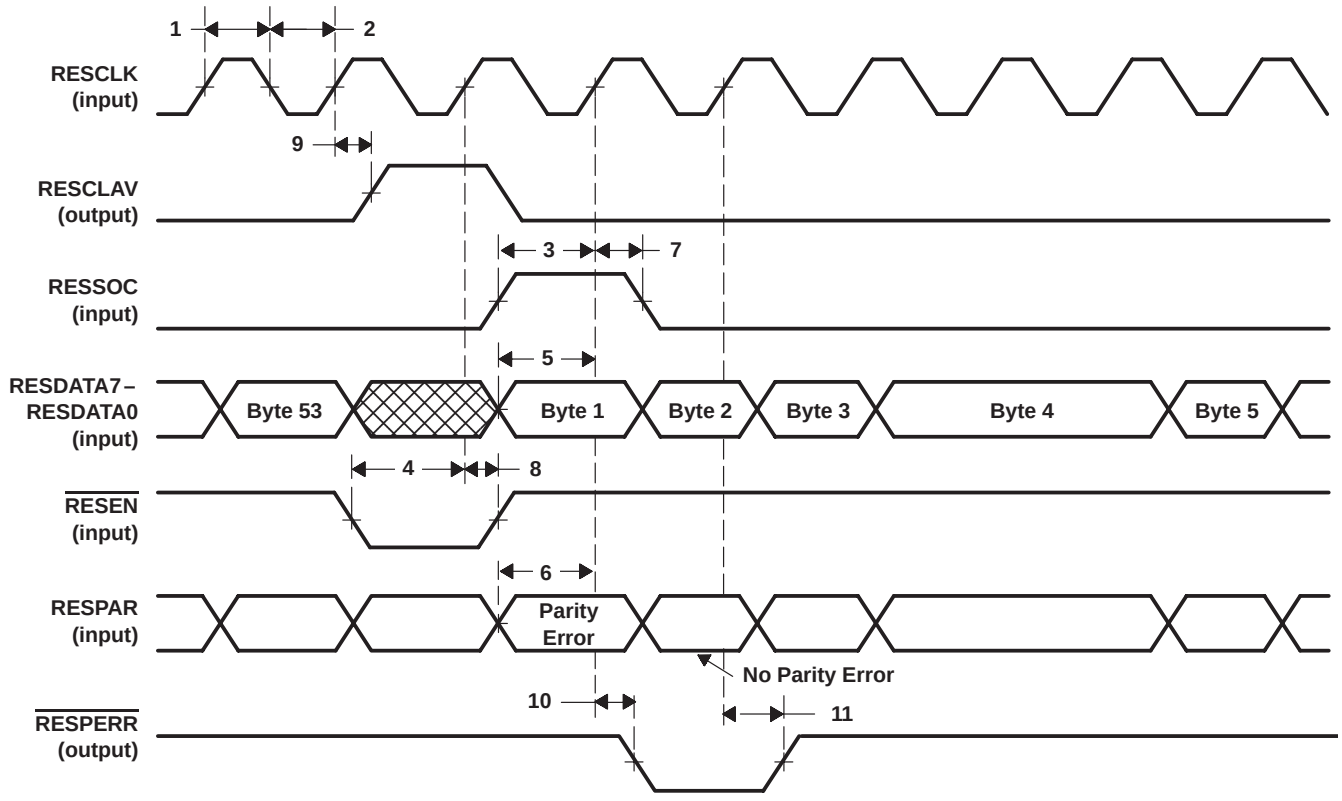


Figure 1. Reassembly-Cell Interface (PHY/$\overline{\text{ATM}}$ Low)

timing requirements (see Note 5 and Figure 2)

NO.			MIN	MAX	UNIT
1	$t_{w(SEGCLKH)}$	Pulse duration, SEGCLK high	12		ns
2	$t_{w(SEGCLKL)}$	Pulse duration, SEGCLK low	12		ns
3	$t_{su(SEGEN)}$	Setup time, $\overline{\text{SEGEN}}$ low before SEGCLK↑	10		ns
4	$t_{h(SEGEN)}$	Hold time, $\overline{\text{SEGEN}}$ low after SEGCLK↑	1		ns

NOTE 5: All output signals are generated on the rising edge of SEGCLK. All inputs are sampled on the rising edge of SEGCLK.

operating characteristics (see Note 5 and Figure 2)

NO.			MIN	MAX	UNIT
5	$t_{d(SEGCLAV)}$	Delay time, SEGCLK↑ to SEGCLAV↓	1	20	ns
6	$t_{d(SEGSOC)}$	Delay time, SEGCLK↑ to SEGSOC↑	1	20	ns
7	$t_{d(SEGDATA)}$	Delay time, SEGCLK↑ to SEGDATA 7–SEGDATA0 valid	1	20	ns
8	$t_{d(SEGPAR)}$	Delay time, SEGCLK↑ to SEGPAR valid	1	20	ns

NOTE 5: All output signals are generated on the rising edge of SEGCLK. All inputs are sampled on the rising edge of SEGCLK.

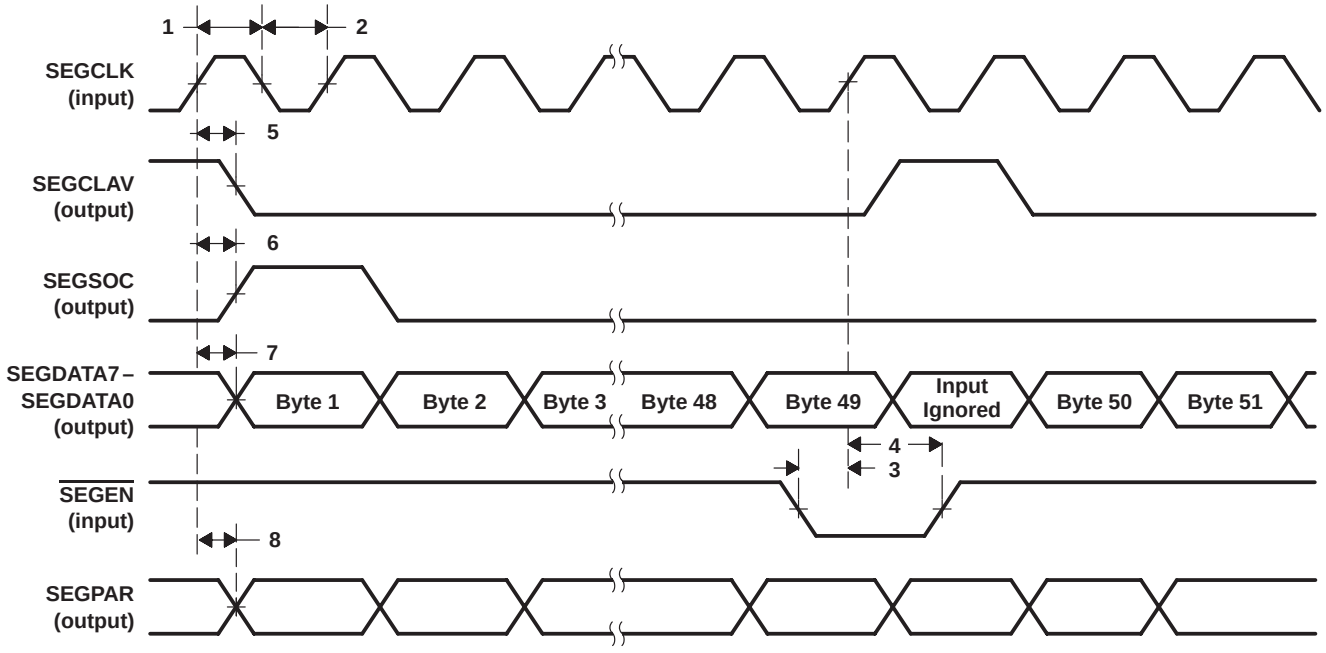


Figure 2. Segmentation-Cell Interface (PHY/$\overline{\text{ATM}}$ Low)

timing requirements (see Note 6 and Figure 3)

NO.			MIN	MAX	UNIT
1	$t_{w(RESCLKH)}$	Pulse duration, RESCLK high	10		ns
2	$t_{w(RESCLKL)}$	Pulse duration, RESCLK low	10		ns
3	$t_{su(RESEN)}$	Setup time, $\overline{\text{RESEN}}$ high before RESCLK↑	10		ns
4	$t_{su(RESSOC)}$	Setup time, RESSOC high before RESCLK↑	10		ns
5	$t_{su(RESDATA)}$	Setup time, RESDATA7–RESDATA0 valid before RESCLK↑	10		ns
6	$t_{su(RESPAR)}$	Setup time, RESPAR valid before RESCLK↑	10		ns
7	$t_{h(RESEN)}$	Hold time, $\overline{\text{RESEN}}$ high after RESCLK↑	1		ns
8	$t_{h(RESSOC)}$	Hold time, RESSOC high after RESCLK↑	1		ns
9	$t_{h(RESDATA)}$	Hold time, RESDATA7–RESDATA0 valid after RESCLK↑	1		ns
10	$t_{h(RESPAR)}$	Hold time, RESPAR valid after RESCLK↑	1		ns

NOTE 6: All output signals are generated on the rising edge of RESCLK. All input signals are sampled on the rising edge of RESCLK.

operating characteristics (see Note 6 and Figure 3)

NO.			MIN	MAX	UNIT
11	$t_{d(RESCLAV)}$	Delay time, RESCLK↑ to RESCLAV↓	1	20	ns
12	$t_{d(RESPERR)1}$	Delay time, RESCLK↑ to $\overline{\text{RESPERR}}$↓	1	20	ns
13	$t_{d(RESPERR)2}$	Delay time, RESCLK↑ to $\overline{\text{RESPERR}}$↑	1	20	ns

NOTE 6: All output signals are generated on the rising edge of RESCLK. All input signals are sampled on the rising edge of RESCLK.

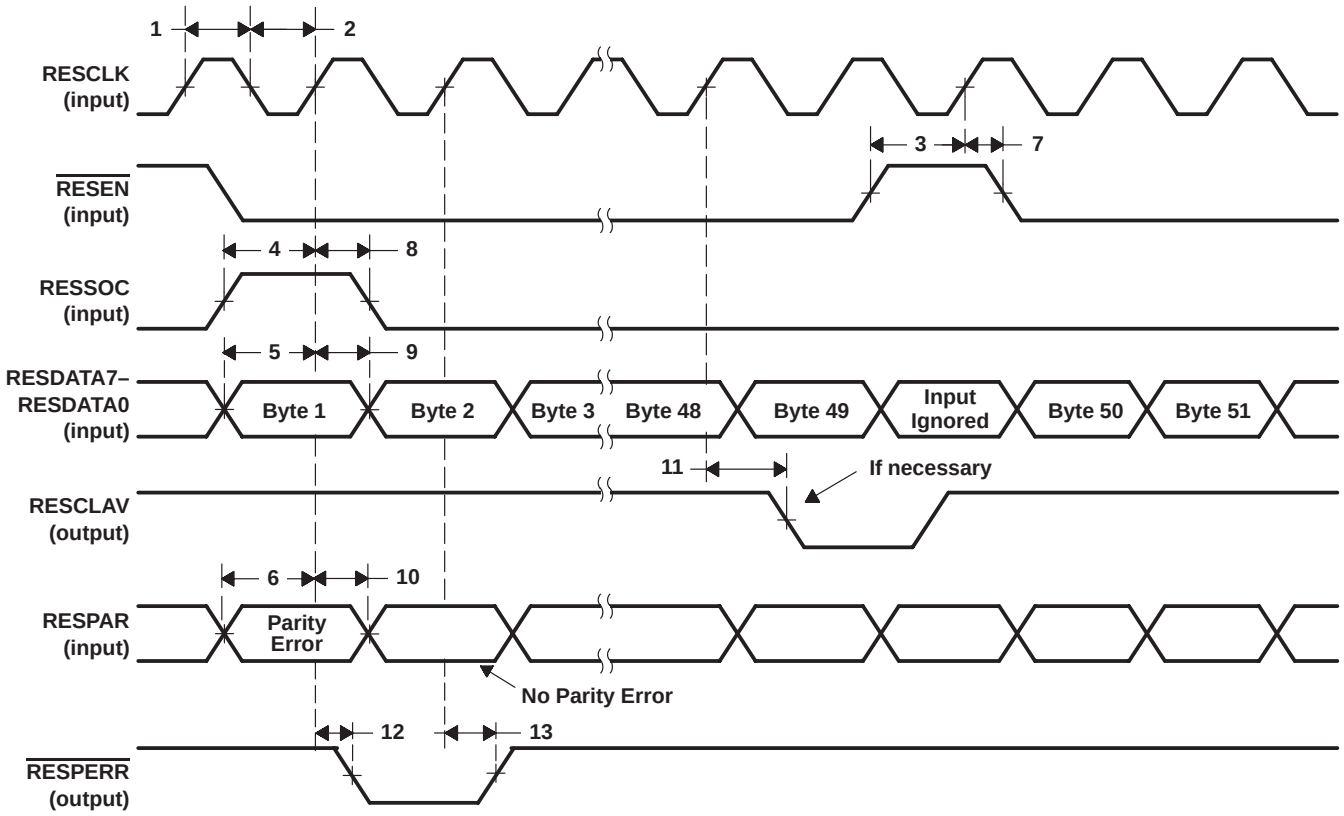


Figure 3. Reassembly-Cell Interface (PHY/$\overline{\text{ATM}}$ High)

timing requirements (see Note 5 and Figure 4)

NO.			MIN	MAX	UNIT
1	$t_{w(SEGCLKH)}$	Pulse duration, SEGCLK high	10		ns
2	$t_{w(SEGCLKL)}$	Pulse duration, SEGCLK low	10		ns
3	$t_{su(SEGEN)}$	Setup time, $\overline{\text{SEGEN}}$ high before SEGCLK↑	10		ns
4	$t_{h(SEGEN)}$	Hold time, $\overline{\text{SEGEN}}$ high after SEGCLK↑	1		ns

NOTE 5: All output signals are generated on the rising edge of SEGCLK. All input signals are sampled on the rising edge of SEGCLK.

operating characteristics (see Note 5 and Figure 4)

NO.			MIN	MAX	UNIT
5	$t_{d(SEGSOC)}$	Delay time, SEGCLK↑ to SEGSOC↑	1	20	ns
6	$t_{d(SEGDATA)}$	Delay time, SEGCLK↑ to SEGDATA7 – SEGDATA0 valid	1	20	ns
7	$t_{d(SEGPAR)}$	Delay time, SEGCLK↑ to SEGPAR↑	1	20	ns
8	$t_{d(SEGCLAV)1}$	Delay time, SEGCLK↑ to SEGCLAV↑	1	20	ns
9	$t_{d(SEGCLAV)2}$	Delay time, SEGCLK↑ to SEGCLAV↓	1	20	ns

NOTE 5: All output signals are generated on the rising edge of SEGCLK. All input signals are sampled on the rising edge of SEGCLK.

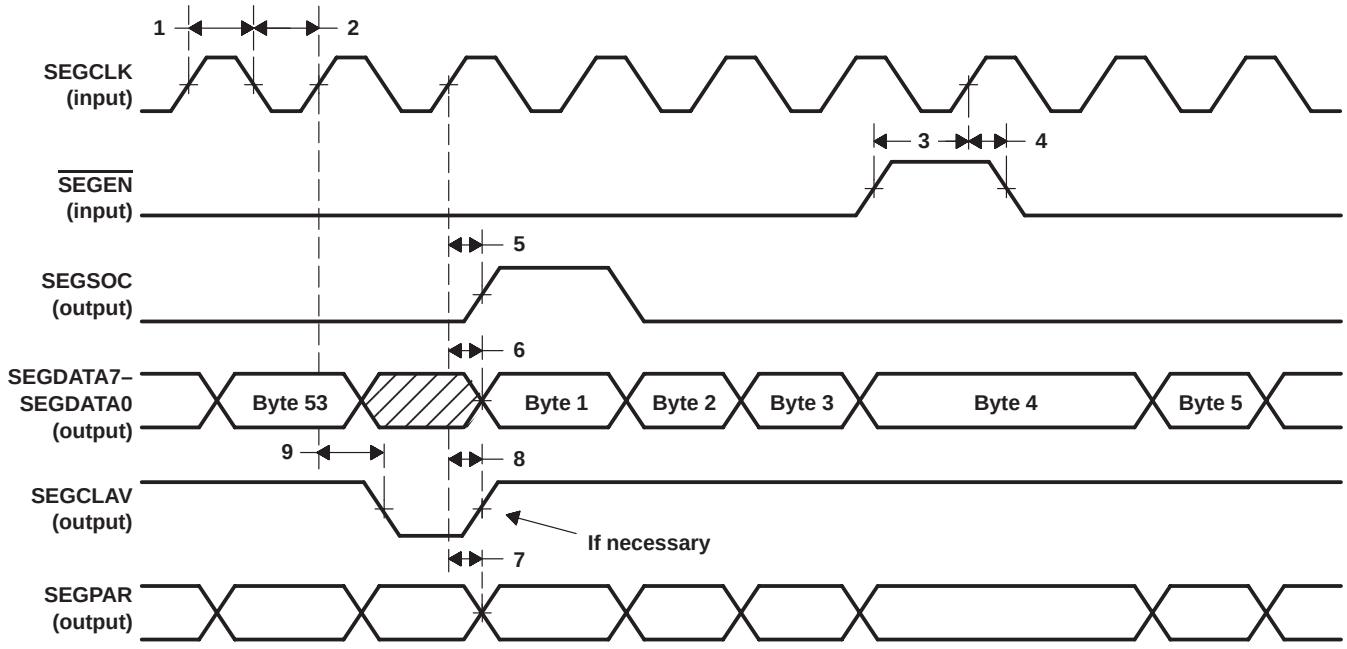


Figure 4. Segmentation-Cell Interface (PHY/$\overline{\text{ATM}}$ High)

timing requirements (see Figure 5)

NO.			MIN	MAX	UNIT
1	$t_{su(PGNT)}$	Setup time, $\overline{\text{PGNT}}$ low before PCLK↑	10		ns
2	$t_{su(PTRDY)}$	Setup time, $\overline{\text{PTRDY}}$ low before PCLK↑	7		ns
3	$t_{su(PDEVSEL)}$	Setup time, $\overline{\text{PDEVSEL}}$ low before PCLK↑	7		ns
4	$t_{h(PTRDY)}$	Hold time, $\overline{\text{PTRDY}}$ low after PCLK↑	2		ns
5	$t_{h(PDEVSEL)}$	Hold time, $\overline{\text{PDEVSEL}}$ low after PCLK↑	2		ns

operating characteristics (see Figure 5)

NO.			MIN	MAX	UNIT
6	$t_{d(PREQ)}$	Delay time, PCLK↑ to $\overline{\text{PREQ}}$↓	2	12	ns
7	$t_{d(PFRAME)}$	Delay time, PCLK↑ to $\overline{\text{PFRAME}}$↓	2	11	ns
8	$t_{d(PCBE)}$	Delay time, PCLK↑ to PCBE valid	2	11	ns
9	$t_{d(PIRDY)}$	Delay time, PCLK↑ to $\overline{\text{PIRDY}}$↓	2	11	ns
10	$t_{d(PAD)}$	Delay time, PCLK↑ to PAD31–PAD0 valid	2	11	ns
11	$t_{d(PSB)}$	Delay time, PCLK↑ to PSB1–PSB0 valid	2	11	ns

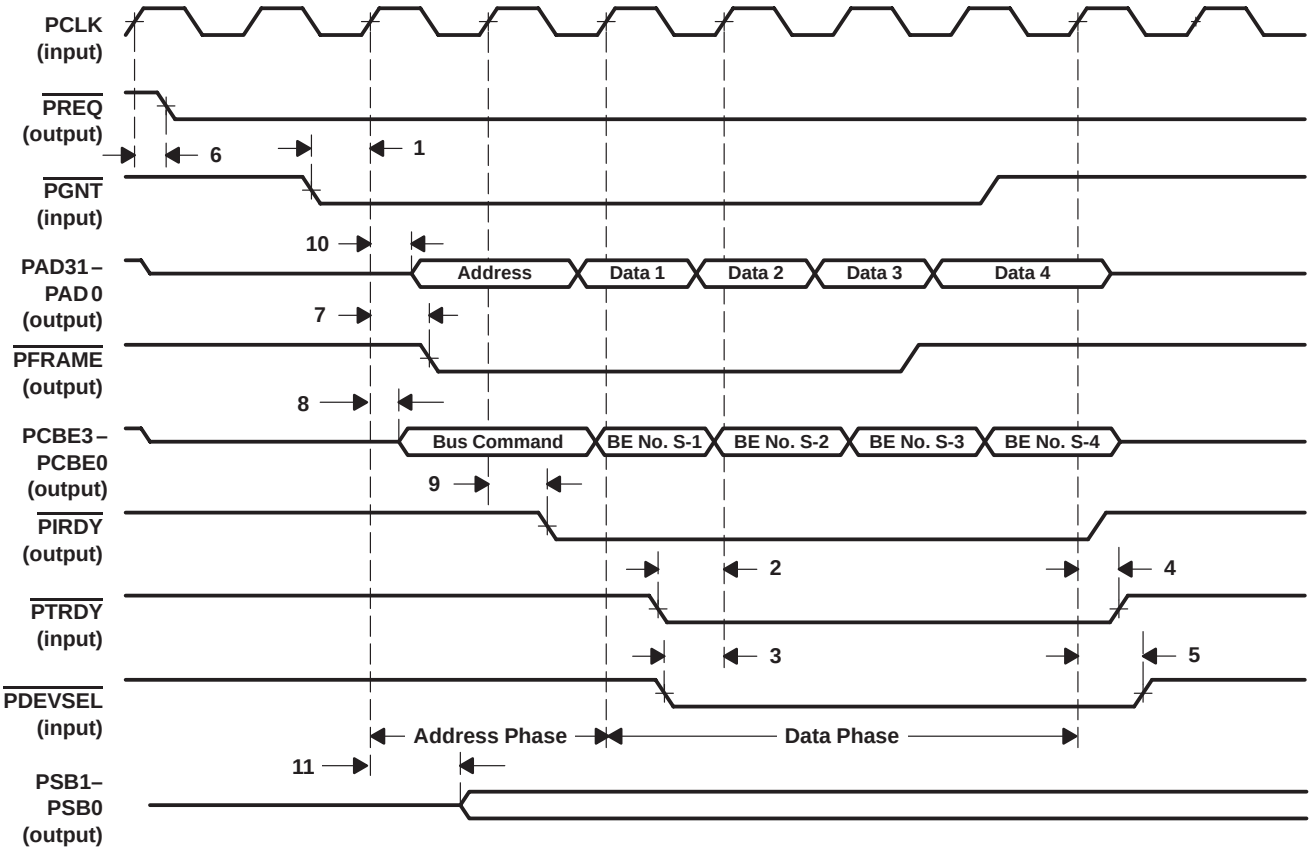


Figure 5. TNETA1570 Write Operation (PCI SAR as Master)

timing requirements (see Figure 6)

NO.			MIN	MAX	UNIT
1	$t_{w(PCLKH)}$	Pulse duration, PCLK high	12		ns
2	$t_{w(PCLKL)}$	Pulse duration, PCLK low	12		ns
3	$t_{su(PGNT)}$	Setup time, PGNT low before PCLK↑	10		ns
4	$t_{su(PAD)}$	Setup time, PAD31–PAD0 valid before PCLK↑	7		ns
5	$t_{su(PTRDY)}$	Setup time, $\overline{\text{PTRDY}}$ low before PCLK↑	7		ns
6	$t_{su(PDEVSEL)}$	Setup time, $\overline{\text{PDEVSEL}}$ low before PCLK↑	7		ns
7	$t_{h(PAD)}$	Hold time, PAD31 – PAD0 valid after PCLK↑	1		ns
8	$t_{h(PTRDY)}$	Hold time, $\overline{\text{PTRDY}}$ low after PCLK↑	2		ns
9	$t_{h(PDEVSEL)}$	Hold time, $\overline{\text{PDEVSEL}}$ low after PCLK↑	2		ns

operating characteristics (see Figure 6)

NO.			MIN	MAX	UNIT
10	$t_{d(PFRAME)}$	Delay time, PCLK↑ to $\overline{\text{PFRAME}}$↓	2	11	ns
11	$t_{d(PAD)}$	Delay time, PCLK↑ to PAD31–PAD0 valid	2	11	ns
12	$t_{d(PCBE)}$	Delay time, PCLK↑ to PCBE3–PCBE0 valid	2	11	ns
13	$t_{d(PIRDY)}$	Delay time, PCLK↑ to $\overline{\text{PIRDY}}$↓	2	11	ns
14	$t_{d(PREQ)}$	Delay time, PCLK↑ to $\overline{\text{PREQ}}$↓	2	12	ns

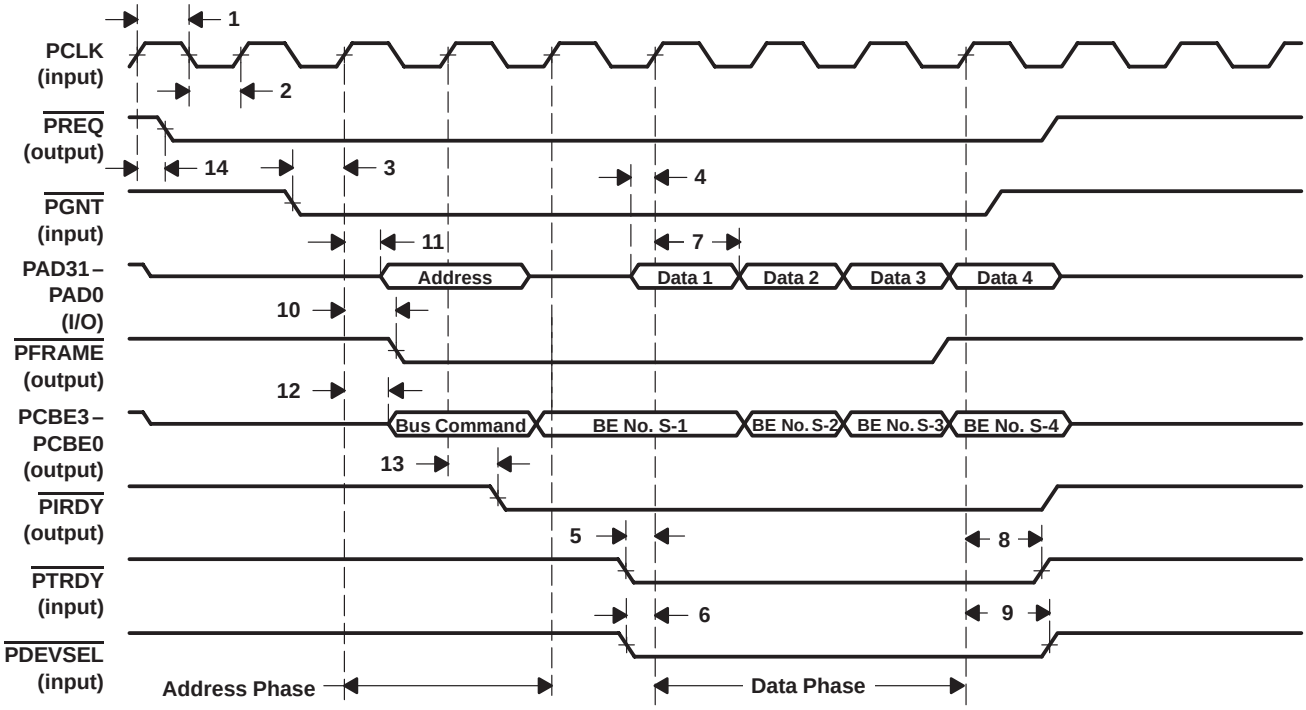


Figure 6. TNETA1570 Read Operation (PCI SAR as Master)

timing requirements (see Figure 7)

NO.			MIN	MAX	UNIT
1	$t_{su(PIDSEL)}$	Setup time, PIDSEL high before PCLK↑	7		ns
2	$t_{su(PAD)}$	Setup time, PAD31–PAD0 valid before PCLK↑	7		ns
3	$t_{su(PCBE)}$	Setup time, PCBE3–PCBE0 valid before PCLK↑	7		ns
4	$t_{su(PIRDY)}$	Setup time, $\overline{\text{PIRDY}}$ low before PCLK↑	7		ns
5	$t_{h(PIDSEL)}$	Hold time, PIDSEL high after PCLK↑	1		ns
6	$t_{h(PAD)}$	Hold time, PAD31–PAD0 valid after PCLK↑	2		ns
7	$t_{h(PCBE)}$	Hold time, PCBE3–PCBE0 valid after PCLK↑	1		ns
8	$t_{h(PIRDY)}$	Hold time, $\overline{\text{PIRDY}}$ low after PCLK↑	1		ns

operating characteristics (see Figure 7)

NO.			MIN	MAX	UNIT
9	$t_{d(PAD)}$	Delay time, PCLK↑ to PAD31–PAD0 valid	2	11	ns
10	$t_{d(PTRDY)}$	Delay time, PCLK↑ to $\overline{\text{PTRDY}}$↓	2	11	ns
11	$t_{d(PDEVSEL)}$	Delay time, PCLK↑ to $\overline{\text{PDEVSEL}}$↓	2	11	ns

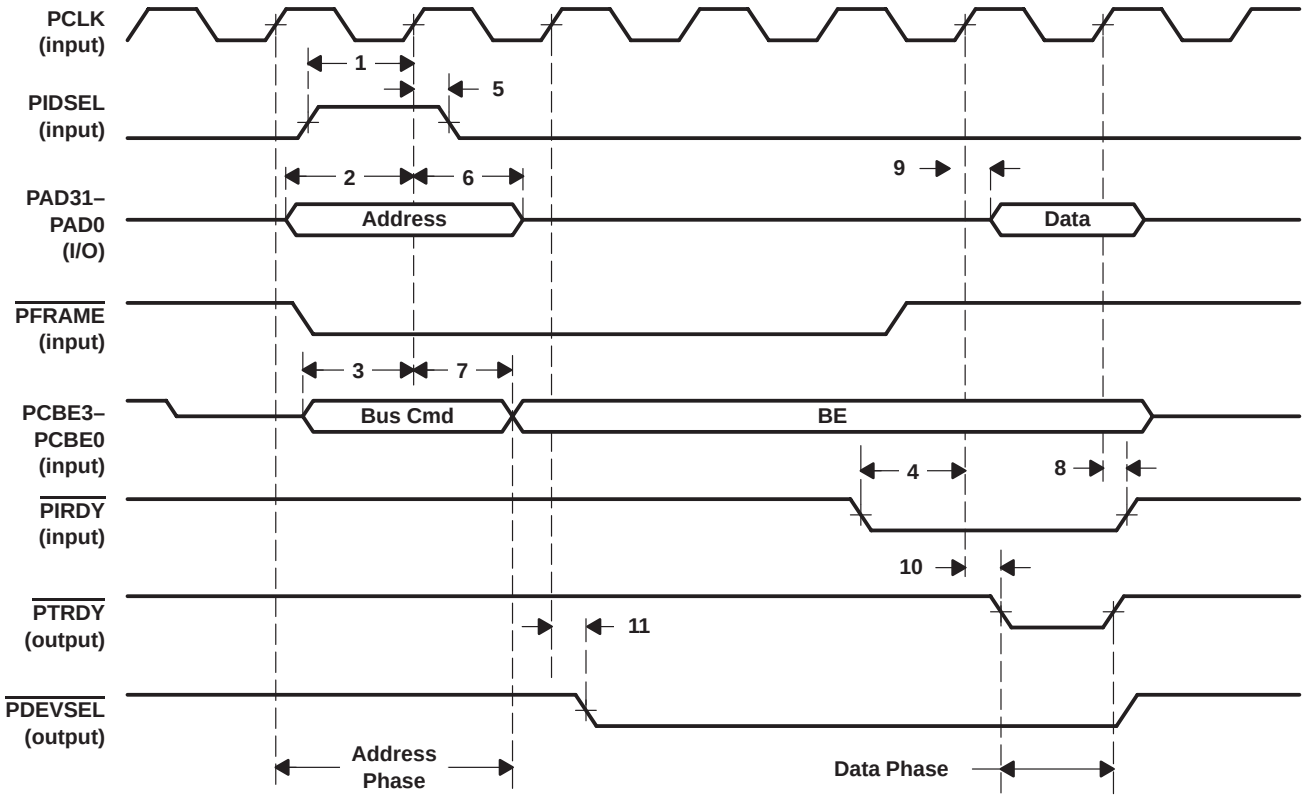


Figure 7. TNETA1570 Read Operation (PCI SAR as Slave)

timing requirements (see Figure 8)

NO.			MIN	MAX	UNIT
1	$t_{su(PIDSEL)}$	Setup time, PIDSEL high before PCLK↑	7		ns
2	$t_{su(PAD)}$	Setup time, PAD31–PAD0 valid before PCLK↑	7		ns
3	$t_{su(PFRAME)}$	Setup time, $\overline{\text{PFRAME}}$ low before PCLK↑	7		ns
4	$t_{su(PIRDY)}$	Setup time, $\overline{\text{PIRDY}}$ low before $\overline{\text{PCLK}}$↑	7		ns
5	$t_{h(PIDSEL)}$	Hold time, PIDSEL high after PCLK↑	1		ns
6	$t_{h(PIRDY)}$	Hold time, $\overline{\text{PIRDY}}$ low after PCLK↑	1		ns

operating characteristics (see Figure 8)

NO.			MIN	MAX	UNIT
7	$t_{d(PTRDY)}$	Delay time, PCLK↑ to $\overline{\text{PTRDY}}$↓	2	11	ns

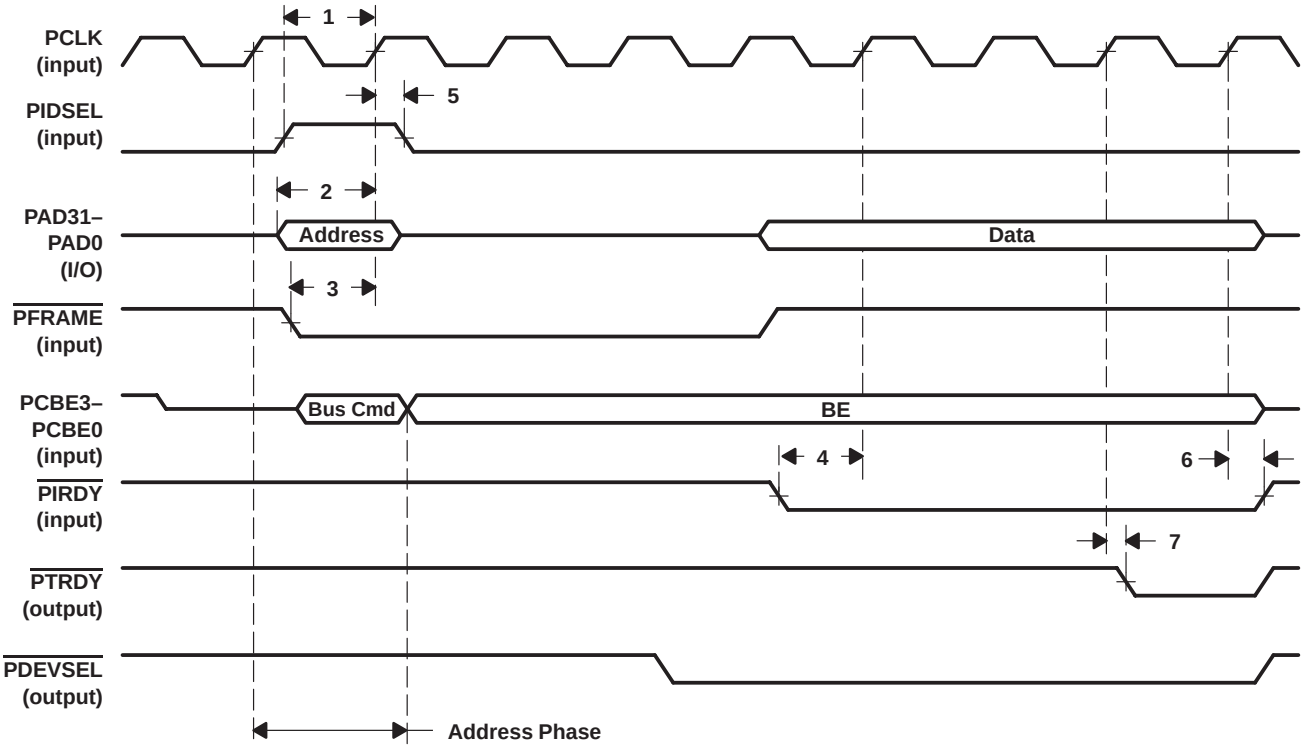


Figure 8. TNETA1570 Write Operation (PCI SAR as Slave)

operating characteristics (see Figure 9)

NO.			MIN†	MAX†	UNIT
1	$t_{w(CMWEL)}$	Pulse duration, $\overline{CMWE}$ low	28	31	ns
2	$t_{d(CMWE)1}$	Delay time, CMAD17 – CMAD0 valid to $\overline{CMWE}\downarrow$	12	15	ns
3	$t_{d(CMWE)2}$	Delay time, CMDATA31 – CMDATA0 valid to $\overline{CMWE}\uparrow$	10	14	ns
4	$t_{d(CMDATA)}$	Delay time, $\overline{CMWE}\uparrow$ to CMDATA31 – CMDATA0 invalid	15	19	ns

† These values are for full-duplex operation at 33 MHz.

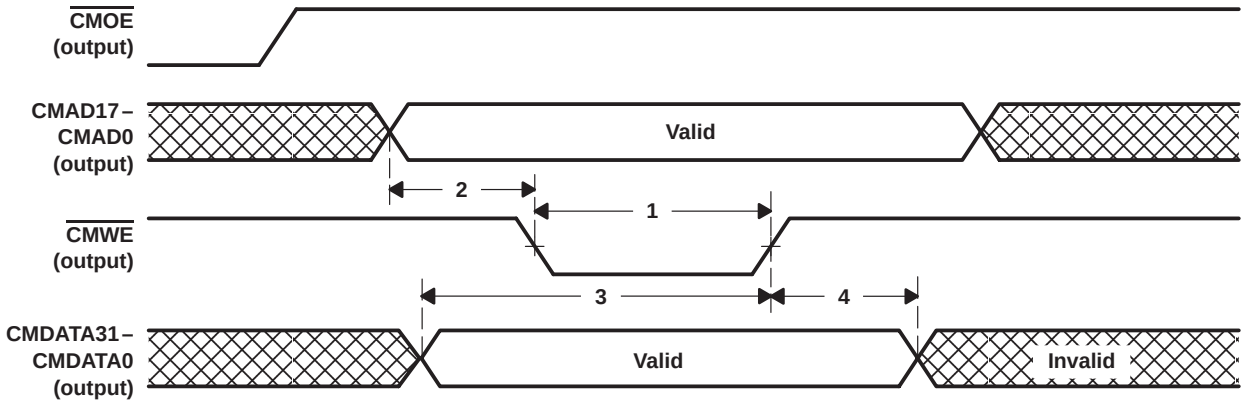


Figure 9. Control-Memory-Interface Write Operation

timing requirements (see Figure 10)

NO.			MIN†	MAX	UNIT
1	$t_{su(CMDATA)}$	Setup time, CMDATA31 – CMDATA0 valid before $\overline{CMOE}\uparrow$	10		ns
2	$t_{h(CMDATA)}$	Hold time, CMDATA31 – CMDATA0 valid after $\overline{CMOE}\uparrow$	0		ns

† These values are for full-duplex operation at 33 MHz.

operating characteristics (see Figure 10)

NO.			MIN	MAX	UNIT
3	$t_{d(CMOE)}$	Delay time, CMAD17 – CMAD0 valid to $\overline{CMOE}\downarrow$		3	ns

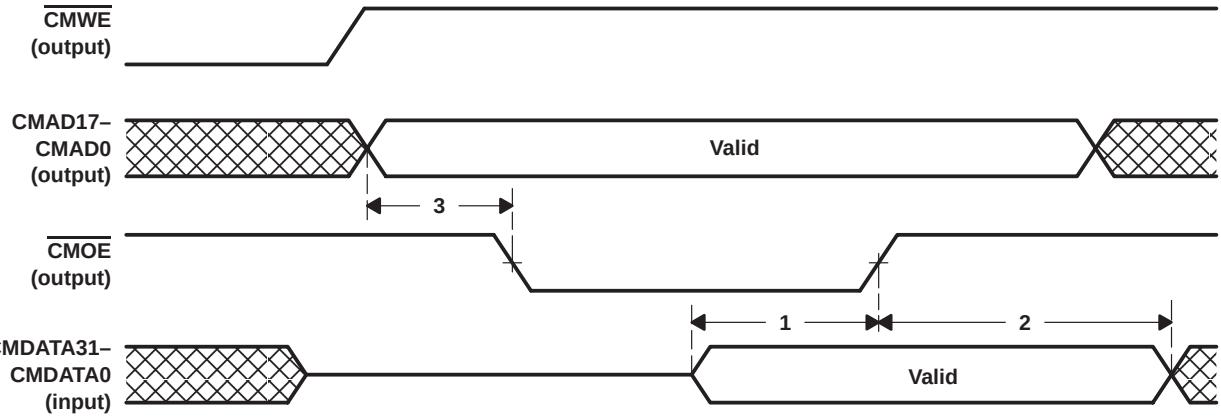


Figure 10. Control-Memory-Interface Read Operation

PRINCIPLES OF OPERATION

Table of Contents

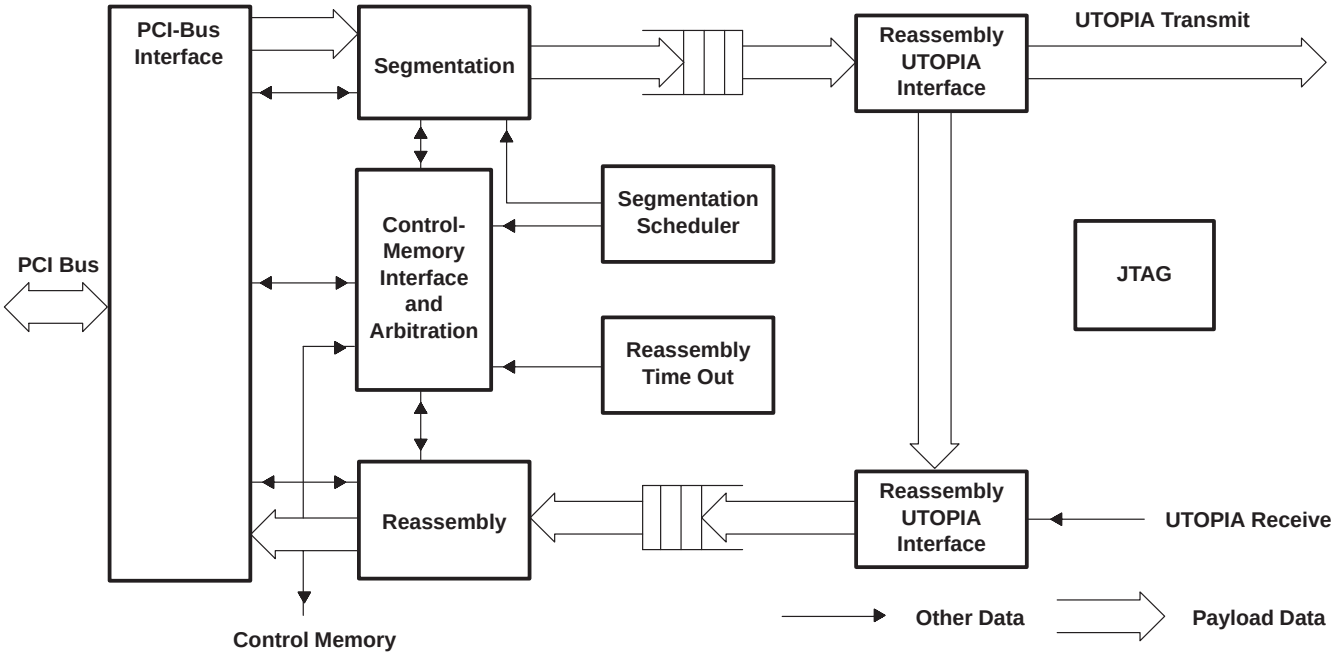
functional overview

The function of the TNETA1570 is centered around the entries in the TX DMA state table for segmentation and the RX DMA state table for reassembly. The entries in the tables hold information regarding what type of data to process, where to find it or store it, and the current status of the segmentation/reassembly process for the entry.

For segmentation, the scheduler table is used to select the next entry in the TX DMA state table for processing. For reassembly, the VPI and VCI fields of the incoming ATM cell are decoded using the VPI/VCI DMA pointer table to locate the entry in the RX DMA state table for processing.

The data interface to the host is the 32-bit/64-bit PCI-bus interface. The interface to the ATM layer is the UTOPIA interface.

functional block diagram



The TNETA1570 can be configured to transmit and receive either AAL5 or transparent-AAL packets. The segmentation and reassembly of AAL5 packets is performed in accordance with ITU-T specifications I.361 and I.363 (11/93 update). The protocol used to segment and reassemble AAL5 packets is well specified in the ITU-T documents. The protocols for segmenting and reassembling a transparent-AAL packet is not specified by any known standards body; therefore, there is no official sanction for a transparent AAL and no interoperability specification exists. However, a transparent AAL provides a convenient method for transporting proprietary control and data information or data traffic that does not fit well in a currently defined AAL (i.e., voice/video traffic).

NOTE: The terms transmit and segmentation are both used to describe the segmentation operation and the terms receive and reassemble are both used to describe the reassembly operation in this document.

PRINCIPLES OF OPERATION

segmentation overview

The segmentation operation of the TNETA1570 can operate on three types of packets: AAL5 packets, transparent (undefined)-AAL packets with end-of-packet (EOP) termination in the PTI field of the ATM header, and transparent AAL-packets without special ATM-header processing for the last cell of the packet. Up to 1023 separate virtual circuits (VCs) can be segmented concurrently, and each corresponds to its own host DMA channel maintained by the TNETA1570. Any VC (DMA channel) can be configured to segment any one of the three packet types. Each packet type is stored in host memory in either single buffers or buffer chains. The buffer-chaining mechanism allows segmentation to begin on a packet before all of the packet contents are established in buffers.

For each VC, the host establishes a VPI/VCI connection and assigns it to a transmit packet-segmentation ring that is assigned a free-segmentation DMA channel controlled by the TNETA1570. Memory structures initialized by the host for packet segmentation on a virtual channel include the scheduler table (control memory), the transmit completion-ring pointers (register), and word 3 of the DMA state entry for the DMA channel assigned to the VC (control memory). The ON bit (bit 31) of word 3 in the DMA state entry activates the DMA channel for packet segmentation. The lower 30 bits of word 3 form the packet-segmentation-ring pointer. The TNETA1570 registers that need to be initiated by the host for segmentation are the configuration register, interrupt-mask register, TX completion ring with and without interrupt registers, and TX packet-segmentation ring-size register.

When at least one buffer of a packet in an established DMA channel is prepared for processing, the RDY bit in the buffer header is set to a 1 indicating that the TNETA1570 has control of the buffer and the start-of-packet (SOP) bit is set in the header. The address of the next buffer, if used, must be known at this time and placed in the appropriate buffer-descriptor location. After this, a pointer to the first buffer of the packet is placed in the next entry of the DMA packet-segmentation ring, and the OWN bit of the entry is set to 1 indicating to the TNETA1570 that a packet is ready for processing. The host indicates the readiness of each subsequent buffer to the TNETA1570 upon completion of its contents preparation. The last buffer of a packet must have the EOP bit set in the buffer descriptor.

Upon a buffer-segmentation completion, the TNETA1570 posts an entry in the next position of a transmit-completion ring. The completion ring with and without interrupt selection is initialized by the host in the buffer descriptor. If a completion ring with interrupt entry is posted and the TNETA1570 interrupt-mask register permits, an interrupt is generated on the PCI bus and an interrupt indication is generated in the status register.

The memory structure for scheduling transmit-DMA-channel service is the scheduler table located in the control memory. The TNETA1570 continuously reads through each entry in the scheduler table from the top down. Two transmit-DMA-channel pointers are located in each 32-bit word entry in this table.

Each entry of the scheduler table is an opportunity for packet segmentation. Each entry indexes the transmit-DMA channel used to extract a single cell during packet segmentation. If the DMA channel is on, the segmentation procedure can continue. If the scheduler-table entry contains zero or the DMA channel is off, no packet segmentation is performed and the next scheduler-table entry is examined.

After determining that a transmit-DMA channel is on, packet segmentation for the channel is initiated. If the ACT bit in the TX DMA state table is set, a packet is begun and not ended. Also, if the current data-byte-count field for the entry is greater than zero, a buffer is in use and packet segmentation resumes from the DMA current buffer-address field. A current data-byte-count field of zero causes the TNETA1570 to use the next buffer-address field to acquire a new buffer from host memory before packet segmentation can begin.

When acquiring a new buffer, the RDY bit of the buffer is examined. If the RDY bit is set, the buffer descriptor is loaded into the DMA-state entry. Packet segmentation for the DMA channel is terminated for the cell opportunity if the buffer RDY bit is not set in the buffer.

PRINCIPLES OF OPERATION

segmentation overview (continued)

The last possibility for an active DMA channel is the start of a new packet. This is detected by the cleared ACT status of the DMA channel. In this case, the packet-segmentation ring-base offset pointer and its index fields of the DMA entry are used to acquire a new buffer address from the packet-segmentation ring for the virtual channel. If the packet-segmentation-ring entry is owned by the TNETA1570, the pointer contained in the entry is used to acquire a new segmentation buffer, the ACT bit of the DMA channel is set, and the packet-segmentation-ring index is incremented. Packet segmentation for the DMA channel is terminated for this cell opportunity if the packet-segmentation-ring entry is owned by the host. It is important that the host sets the RDY bit of the first buffer of a packet before setting the OWN bit of its packet-segmentation-ring entry.

The 32-bit, single and multiple data-phase PCI accesses are used by the TNETA1570 for buffer and DMA initialization information. After establishing a buffer address to acquire cell-payload data, the 4-byte ATM header contained in the DMA entry is loaded into the internal transmit FIFO. The EOP bit in the ATM header is set for the last cell of AAL5 packets and PTI-based transparent packets. If the EOP status in the DMA entry is cleared and the current-data-byte count is greater than or equal to 48, a 48-byte PCI-bus read is used to send the cell-payload contents to the transmit FIFO. If the EOP status in the DMA channel is cleared and the current-data-byte count is less than 48, the remaining bytes are acquired from the data buffer. The next data buffer is obtained using the next buffer-address field in the TX DMA state table. If segmentation is occurring on an EOP buffer, the TNETA1570 performs a PCI-bus read equal to the current-data-byte count acquiring cell payload and providing byte pad as required.

At the end of a segmentation cycle for a single buffer that is not the end of a packet, the TX DMA state table entries for current-data-byte count and current-buffer address are modified to show the number of bytes remaining in the buffer and the starting location for the next ATM-payload fill. The 64-bit, if allowed, or 32-bit multiple data-phase PCI accesses are used by the TNETA1570 for acquiring ATM payload.

When the AAL5 indicator is set in the TX DMA state table entry, the AAL5 PDU pad and trailer are added to the last cell(s) by the TNETA1570. As each 64-bit ATM-payload word is loaded in the transmit FIFO, a 32-bit CRC is calculated with the generator polynomial x^32 + x^26 + x^23 + x^22+ x^16 + x^12+ x^11 + x^ 10 + x^8 + x^ 7 + x^ 5 + x^4 + x^2 + x +^1. The AAL5-packet length is also calculated on the number of nonpad bytes in the AAL5 SDU for each ATM-payload fill. At the end of each AAL5-segmentation cycle for a single cell, the partial CRC and current AAL5-packet length are updated in the TX DMA state table entry. In many cases, pad bytes (all 0) must be added after the AAL5 SDU and before the AAL5 trailer to complete cells. The 32-bit CRC is also calculated using the pad bytes and the control and length fields of the trailer.

Upon processing an AAL5 packet where the current-data-byte count of the last cell in an EOP buffer is between 0 and 40 bytes, pad bytes (all 0)are added up to the 40th payload octet. The AAL5-control field, CPCS-UU and CPI octets, stored in the TX DMA state table entry are packed in payload octets 41 and 42, and the AAL5 length-field calculated across the SDU (not including the pad) is placed in octets 43 and 44. The 32-bit CRC for the PDU calculated across the packet data, pad, control, and length fields is inverted and placed in octets 44 through 48. For the last cell in an EOP buffer, which is filled with exactly 40 bytes from the segmentation buffer, no pad is added. After processing the last cell of the packet, the DMA ACT bit is cleared.

When processing a cell in the last buffer of an AAL5 packet with the current-data byte-count field in the TX DMA state table entry between 41 and 48 bytes, the pad is added up to the 48th octet. The segmentation unit creates an empty cell including the 8-byte AAL5 trailer. The current-data-byte count in the TX DMA state table entry is set to zero, but the ACT bit remains set.

Two methods are available for transmitting a transparent-AAL packet. The first method is basically the same as the procedure for transmitting an AAL5 packet, except that no 8-byte trailer (control, length, and 32-bit CRC) is generated and transmitted. The TNETA1570 adds any padding necessary to complete the 48-byte cell payload. The last cell in the transparent-AAL packet has the PTl bits modified to indicate that this is the last cell in the packet, (i.e., the PTI bits is set to 0×1). Abort processing is disabled for transparent-AAL packets.

PRINCIPLES OF OPERATION

segmentation overview (continued)

The second method for transmitting a transparent-AAL packet differs from the first method only in that the PTI bits in the last cell of the packet are not modified. The PTI bits are transmitted as they occur in the descriptor entry for all cells in the packet. This method can be used to transmit VC-level OAM cells, which are single cell packets that set the PTI bits to a value of l0x.

After segmenting the contents of a buffer, the current-data-byte count for the DMA entry is set to zero. The OWN bit of the next entry of the completion ring selected for the channel is examined. If the entry is owned by the host, the transmit-freeze bit in the status register is set and an interrupt is generated. Otherwise, a pointer to the start of the buffer is placed in the entry.

The size of the internal transmit FIFO is four cells. The UTOPIA interface for the segmentation unit can operate asynchronous to the PCI bus as cells are read from the transmit buffer in 8-bit bytes. The HEC byte for the ATM header is calculated byte wise and inserted in the fifth-header octet as a cell is read. When the HEC error bit in the configuration register is set, the HEC byte is inverted prior to being inserted into the outgoing byte stream for test purposes. The segmentation UTOPIA bus implements parity generation.

reassembly overview

For reassembly operation, the host must allocate memory for the following memory data structures: data buffers with descriptor entries, completion ring with/without interrupt pointers, the receive-free buffer rings/receive-free buffer FIFO, the RX VPI/VCI DMA pointer table, and a DMA state table entry for each VCI. The TNETA1570 registers, which need to be initialized by the host for reassembly, are the configuration register, the interrupt-mask register, the reassembly-global-timer register, the RX DMA-state-table register, the completion ring with/without interrupt-size register, and the FIFO free-buffer-size register.

Free buffers are fetched using either the receive free-buffer ring or the receive free-buffer FIFO. If the free-buffer rings are used, the lower eight bits of word 6 of the DMA state table entry are read to obtain the index for the free-buffer rings. The index points to one of the 256 entries in the free-buffer-ring-pointer table. Each entry contains the base address of each free-buffer ring, as well as the buffer size, the ring size, and the current index. If the free-buffer FIFO is used, a read on the address in word 6 of the DMA state table provides the starting address of a free buffer.

If the receive free-buffer-ring method is used, the host sets the OWN bit of a completion-ring entry to indicate that the buffer is available for use by the TNETA1570. If no free buffers are available, buffer starvation occurs and the reassembly of the packet is terminated.

The completion ring with/without interrupt is initialized by the host in the DMA state table entry. If a completion ring with interrupt entry is posted and the TNETA 1570 interrupt-mask register allows, an interrupt is generated on the PCI bus and an interrupt indication is placed in the status register. A bit in the configuration register can be set so that an interrupt is posted at the completion of the segmentation of either a packet or a buffer.

The VPI of an incoming cell is used as an index into the VPI/VCI DMA pointer-table entry, which resides in control memory. The information in the VPI/VCI DMA pointer table is used to determine if the VPI is enabled to receive packets and if the VCI is within the valid range for the corresponding VPI. The index value in the VPI/VCI pointer table together with the VCI value of the ATM cell are used to form the address of the DMA state-table entry for that particular VCI. If a VCI is found to be out of range, an interrupt is generated.

If the PTI value of the incoming cell is 100, the cell is an OAM F5-segment cell. If the PTI value of the incoming cell is 101, the cell is an OAM F5 end-to-end cell. Both OAM cells are routed to DMA channel 0. If the PTI value is 110 or 111, the cell is routed to DMA channel 1. If the VCI value of the incoming cell is three, the cell is an OAM F4 segment cell. If the VCI value of the incoming cell is four, the cell is an OAM F4 end-to-end cell and both F4 OAM cells are routed to channel 2.

PRINCIPLES OF OPERATION

reassembly overview (continued)

If the VPI is on, the DMA entry is enabled, the wait on EOP is cleared, the VCI is in range, and the TNETA1570 owns the free buffer, the TNETA1570 begins the reassembly process for the incoming cell.

Word 0 of the free-buffer descriptor is copied to word 2 of the DMA state table entry to get the start-buffer address. If the receive free-buffer rings are used, the lower 16 bits of word 3 in the state table, the current data-byte count, is copied from word 1 of the appropriate receive free-buffer ring-pointer entry. If the receive free-buffer FIFO is used, the lower 16 bits of word 3 in the state table are copied from the FIFO free-buffer-size register. If the cell is a SOP, word 0 of the descriptor entry is copied to word 4 of the state table to keep track of the first buffer in the packet. When this is completed, transfer of the cell to host memory occurs.

The current-data-byte count is decremented by one for each cell transmitted to the host. If the current-data-byte count reaches zero and an EOP has not yet arrived, a new buffer is fetched either from the free-buffer ring or the FIFO.

The TNETA1570 can be configured to reassemble either AAL5 or transparent-AAL packets. Whether a DMA state table entry performs AAL5, PTI-based-transparent AAL, or counter-based-transparent AAL, processing is set by the host at initialization. If the cell belongs to an AAL5 packet, the CRC value is calculated and the trailer is extracted and posted in the completion ring. Incoming VC-level OAM cells are reserved for DMA channels 0 and 1. VP-level OAM cells are routed to DMA channel 2. No other incoming packets can use these channels. AAL5 packets are terminated upon detection of EOP.

Two methods of terminating transparent-AAL packets are provided. The first method uses the PTI bits to indicate EOP. This method uses a PTI value of 0×1 to initiate EOP processing. The second method uses a counter to count the number of transparent-AAL cells that have been received. Once the counter equals a predetermined value, the device performs EOP processing. For both methods, the 32-bit CRC, packet abort, and packet-overflow checks are disabled. In addition, reassembly aging is not performed on receive channels configured to reassemble transparent-AAL packets using the counter-based method. Receive channels using the PTI field to determine EOP allows reassembly aging and can experience a time-out. For the PTI-based scheme, the reassembly time-out provides a method of detecting that the cell containing the EOP indication has been dropped or lost somewhere in the transmission path.

When the reassembly of a packet is complete, an entry is posted to either the completion rings with or without interrupt. The ATM header is written to word 0 of the RX completion ring. If the EOP processing started because of an error condition, that information is posted, along with the packet-byte count and the congestion-cell count, to word 1. Word 4 of the DMA state table, which holds the SOP pointer, is copied to word 2 of the completion-ring entry. If the reassembled packet is AAL5, the AAL5 trailer is copied to word 3. Word 6 of the DMA state table, which contains the free-buffer FIFO pointer/RX free-buffer-ring pointer, is copied to word 4 of the RX completion-ring entry.

The EOP bit, bit 30 of word 1 in the RX buffer-descriptor entry, is set in the last buffer in a packet during EOP processing. EOP processing occurs for normal termination of a packet, a RX packet-aging time-out, AAL5 SDU overflow, abort or buffer starvation. Packet aging is provided to age the packets that are in the reassembly process. This option is programmable both on global and local level. This feature is intended to be invoked if a packet fails to proceed to EOP processing because of an error condition in the network.

At initialization, the reassembly-global-aging timer is loaded and a time-out value is written to each DMA state table entry. When the global counter reaches zero, the TNETA1570 reads the address register to get the DMA state table to age. If the state table is active, word 7 is read to obtain the time-out value and the current time-out count. If a time-out has occurred, EOP processing begins immediately. If a time-out has not occurred, the current time-out count is incremented. If the state table is not active, the current time-out count is not incremented. The TNETA1570 performs HEC checking for the incoming cell. If a HEC error is detected, the cell is dropped.

PRINCIPLES OF OPERATION

segmentation data structures

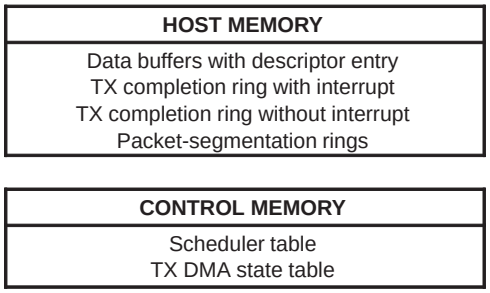
The transmit operation uses data structures that reside in both control memory, host memory, and internal registers. Control memory is used to store dynamic parameters associated with the segmentation of a particular packet, as well as configuration information for each individual transmit group that is initialized by the host at startup. Control memory contains both dynamic and static parameters. Host memory contains the data buffers that are to be segmented and the information required to begin segmentation of an individual data buffer into ATM cells. The information contained in host memory is specific to the data buffers currently being transmitted and is constantly being updated.

The transmit data structures that reside in host memory are the data buffers, which include the interrupt and the associated descriptor information, and the transmit completion rings with/without packet-segmentation rings. The transmit completion rings contain the addresses of the data buffers that have completed segmentation and are used by the host to reclaim buffers. The host must decide at initialization whether a transmit DMA channel uses the transmit completion ring with or without interrupt. The difference between the two rings is that when an entry is posted to the completion ring with interrupt, an interrupt is generated by the TNETA1570. When an entry is posted to the completion ring without interrupt, an interrupt is not generated. The size of the transmit completion rings is variable and is set by the host at initialization by writing to the transmit-completion-ring-size register located inside the TNETA1570.

The packet-segmentation rings are used to queue up packets for transmission. The location of the first buffer in a packet that is to be transmitted is written into the segmentation-ring entry. Each packet-segmentation ring has up to 256 entries. Since there is a separate packet-segmentation ring for each transmit DMA channel, there is a maximum of 1023 segmentation rings in host memory.

The transmit data structures that reside in control memory are the scheduler table and the transmit DMA state table. The scheduler table is used to schedule the different virtual connections and contains 3100 32-bit words with two 16-bit entries per word. This provides a resolution of approximately 32 Kbit/s. The transmit DMA state table contains the DMA-channel-state information. It has a maximum of 1023 entries.

Internal registers are used for the scheduler-table size, TX packet-segmentation-ring size, TX completion-ring pointers, and the TX completion-ring size.



reassembly data structures

The receive operation, as the transmit, uses data structures in both host memory and control memory to reassemble incoming packets. The actual reassembly of a packet occurs in host memory; the device contains only enough internal buffering to absorb the effects of bus transactions and availability. Control memory is used to maintain the status and configuration information necessary to complete the reassembly of a packet.

The receive data structures that reside in host memory are the data buffers (including the descriptor information), receive completion-rings with/without interrupt, and receive free-buffer rings (FIFOs). A receive completion-ring entry contains the address of the first data buffer of a packet that has completed reassembly and is used to notify the host that a packet has been received. The host must decide at initialization whether

PRINCIPLES OF OPERATION

a receive DMA channel uses the receive completion ring with or without interrupt. The difference between the two rings is that when an entry is posted to the completion ring with interrupt, an interrupt is generated by the TNETA1570. When an entry is posted to the completion ring without interrupt, an interrupt is not generated. An entry to either ring is posted when the reassembly process is completed. The sizes of the receive completion rings are variable and are set by the host at initialization by writing to the receive completion ring with/without interrupt size registers located inside the TNETA1570.

The TNETA1570 allows the use of multiple buffers to store an incoming packet (receive-buffer chaining). When a new packet arrives or a buffer used to store cells from an incoming packet is filled, the device obtains an empty buffer from either the receive free-buffer ring or the free-buffer FIFO, depending upon the configuration of the receive DMA channel that is reassembling the packet. The device allows up to 256 different receive free-buffer rings to be set up in host memory. If the user chooses to use free-buffer FIFOs, a separate receive free-buffer FIFO can be assigned to each receive DMA channel.

The receive data structures that reside in control memory are the receive DMA state table, the receive VPI/VCI DMA pointer table, and the receive free-buffer ring-pointer table. The receive DMA state table contains the state information associated with the reassembly of an incoming packet. This table contains a maximum of 30 720 entries. The receive VPI/VCI DMA pointer table provides the address to the receive DMA state table entry for a particular receive channel. The free-buffer ring-pointer block contains the addressing for the receive free-buffer rings, as well as the size of the buffers associated with each free-buffer ring.

Internal registers are used for the maximum RX DMA state-table size, RX unknown, RX FIFO free-buffer size, RX completion-ring pointers, RX completion-ring size, and reassembly-global timer.

HOST MEMORY
Data buffers with descriptor entries
RX completion ring with interrupt
RX completion ring without interrupt
RX free-buffer rings/receive free-buffer FIFOs

CONTROL MEMORY
RX DMA state table
RX VPI/VCI DMA pointer table
RX free-buffer ring pointers

internal registers

The TNETA1570 has several internal registers and counters for keeping statistics on selected parameters, configuring the operation of the device, device status and interrupt masking, and for aging packets currently undergoing reassembly. These registers are accessible by the host.

The internal registers and counters with some of their features are shown in the following table. A detailed description can be found in the internal registers and counters section.

PRINCIPLES OF OPERATION

internal registers (continued)

DESCRIPTION	SIZE	R/W
Configuration register	32 bits	R/W
Status register	32 bits	R
Interrupt-mask register	32 bits	R/W
Reassembly-global-timer register	16 bits	R/W
Maximum RX DMA state-table register	16 bits	R/W
RX unknown register	32 bits	R
TX completion ring with interrupt-size register	16 bits	R/W
TX completion ring without interrupt-size register	16 bits	R/W
RX completion ring with interrupt-size register	16 bits	R/W
RX completion ring without interrupt-size register	16 bits	R/W
TX packet-segmentation ring-size register	16 bits	R/W
FIFO free-buffer-size register	16 bits	R/W
HEC-error counter	16 bits	R
Unknown-protocols counter	32 bits	R
AAL5 CPCS PDU-discard counter	16 bits	R
ATM-cells-received counter	32 bits	R
ATM-cells-transmitted counter	32 bits	R
VCI-mask register	16 bits	R/W
FIFO maximum occupancy value	16 bits	R
Scheduler-table-size register	16 bits	R/W
Software reset	32 bits	W
TX completion ring without interrupt-pointer address	32 bits	R/W
TX completion ring with interrupt-pointer address	32 bits	R/W
RX completion ring without interrupt-pointer address	32 bits	R/W
RX completion ring with interrupt-pointer address	32 bits	R/W

host interface

The TNETA1570 incorporates a PCI revision 2.0 (April 30, 1993) compliant host interface. The TNETA1570 operates as a 32-bit PCI-slave device for configuration cycles, accesses to internal registers, and accesses to the onboard control memory. It also acts as a PCI-master device for accessing data structures that are contained in host memory. The TNETA1570 initiates 64-bit or 32-bit data transfers to and from data structures in host memory, but only initiates 32-bit data transfers to and from control structures in host memory.

segmentation operation

The transmit operation of the TNETA1570 utilizes data structures in both host memory and control memory. The packets being segmented are stored in buffers in host memory, along with the descriptor-entry information for that buffer. The transmit completion rings and packet-segmentation rings also are located in host memory. Control memory contains the scheduler table and the TX DMA state table. These data structures are described in the following paragraphs.

The segmentation process for the TNETA1570 is based on the segmentation of buffers instead of the segmentation of packets. EOP processing still is required to add the AAL5 trailer to the last cell in an AAL5 packet. However, segmentation can start once a buffer ending on an even 48-byte boundary is filled, instead of waiting until all of a packet is stored in memory. The segmentation process also uses link entries to queue up buffers associated with the same packet for segmentation. For the targeted application, link entries provide a more efficient use of host memory for storing descriptor and user data. The packet provides a more efficient use of host memory for storing descriptor and user data. The packet-segmentation rings are used to queue up different packets for segmentation.

PRINCIPLES OF OPERATION

segmentation operation (continued)

The host interface on the TNETA1570 is a PCI interface with a selectable 64-bit or 32-bit address/data bus. The boundaries of the data structures contained in host memory must be set to either 64 or 32 bits, depending upon the configuration selected. The control-memory interface is a 32-bit interface and all data structures in control memory contain 32-bit entries. The host interface on the TNETA1570 is responsible for ensuring the proper data formatting within the limitations set for transactions between host memory and control memory and for host accesses of internal registers.

scheduler table

Addressing	Control-memory address 00000–00CF7. An autonomous counter points to the entries sequentially, thereby initiating the segmentation engine.
Size	Max logical entries: 6200 Max 32-bit words: 3100 Min logical entries: 2 Min 32-bit words: 1

The scheduler table is located in control memory. It contains the order in which the transmit DMA channels are serviced. This structure consists of 3100 32-bit words with two 16-bit entries per word, providing for a maximum of 6200 entries. Only ten of the 16 bits in each entry are used, allowing for the simultaneous segmentation of 1023 packets. A zero entry means that no cell is transmitted during this cell opportunity. If a zero is detected, the entry is skipped and the next entry is examined immediately. The scheduler table is initialized by the host at startup.

ENTRY	DESCRIPTION			
Word 0	Reserved (bits 31–26)	Entry 1 (bits 25–16)	Reserved (bits 15–10)	Entry 0 (bits 9–0)
Word N	Reserved (bits 31–26)	Entry 2N + I (bits 25–16)	Reserved (bits 15–10)	Entry 2N (bits 9–0)
Word 3099	Reserved (bits 31–26)	Entry 6199 (bits 25–16)	Reserved (bits 15–10)	Entry 6198 (bits 9–0)

NOTE: This table may be modified dynamically. This means that during normal operation of the device, the host may modify the contents of the table while the transmit section of the device is operational. The ability to dynamically modify the contents of the scheduler table must be comprehended in the design.

The MSBs for the table entries are bits 25 and 9 , the LSBs for the table entries are bits 16 and 0. Allowed values in the entry fields are 0 to 1023. The address to the TX DMA state-table-entry word 0 can be found by using the following formula:

(index value) $\times$ 8 entries $\times$ 4 bytes per entry + 8000h

transmit DMA state table

Addressing	Control-memory address 0201h – 3FFFh (normal memory map). The entries are addressed using the scheduler-table-entry values.
Size	Max logical entries: 1023 Max 32-bit words: 8184 Min logical entries: 1 Min 32-bit words: 8

The transmit DMA state table resides in control memory and contains 1023 states, allowing for the simultaneous segmentation of 1023 packets. Each state entry contains eight 32-bit words. The transmit DMA state table has several entries that must be initialized, including whether a particular state is on or off and the location of the first data buffer for the first packet sent. Four of the eight words in the transmit DMA state table are copied from the descriptor entry located at the start of each data buffer queued for segmentation.

PRINCIPLES OF OPERATION

transmit DMA state table (continued)

ENTRY	DESCRIPTION
Word 0	Control field, buffer offset, buffer data-byte count
Word 1	Current-buffer address
Word 2	4-byte ATM header
Word 3	DMA state on/off
Word 4	Next-buffer address
Word 5	Start-of-buffer address
Word 6	Partial AAL5-transmit CRC
Word 7	AAL5-control field and length field

word 0

Control (bits 31–25)	Reserved (bit 24)	Buffer Offset (bits 23–16)	Buffer Data-Byte Count (bits 15–0)

The contents of word 0 are copied directly from word 0 of the descriptor entry located at the start of each data buffer.

act (ACTive) (bit 31)

This bit is set when the DMA channel is in the process of segmenting and transmitting a packet. This bit is cleared by the TNETA1570 after the entire packet is transmitted.

start of packet (SOP) (bit 30)

This bit is set by the host indicating that this is the first buffer of a packet that consists of one or more buffers. This bit is also set if a packet consists of only a single buffer. This bit is not checked by the TNETA1570.

end of packet (EOP) (bit 29)

This bit indicates that this is the last buffer of a packet; therefore, every packet has at least one buffer with the EOP bit set. For a packet that consists of a single buffer, both the SOP and EOP bits are set.

abort (bit 28)

This bit indicates that EOP processing is initiated immediately and no further processing of the current packet will occur. When both this bit and the AAL5-indicator bit are set, the AAL5 length field in the outgoing AAL5 packet is reset to zero.

transmit AAL-packet-type indicator (bits 27 and 26)

These bits are set by the host to indicate the AAL type of the packet being transmitted. Three options are available for processing a packet for transmission as shown in the following table:

BIT 27	BIT 26	INDICATION
0	0	Transparent-AAL packet with PTI bits set to indicate EOP
0	1	AAL5
1	0	Transparent-AAL packet without PTI bits set to indicate EOP
1	1	Invalid

AAL5 processing includes the calculation of the 32-bit CRC and the addition of the pad, control, and length fields. All buffers associated with an AAL5 packet must have the AAL5-indication bits set. When either of the other two processing methods is chosen, the TNETA1570 does not perform packet-level processing, except for adding the necessary pad to complete a 48-byte payload. If both bits 27 and 26 are set to zero, the device also sets the PTI field in the last cell of a packet to a value of $0 \times I$ to indicate EOP. If bits 27 and 26 are set to a value of 10, the PTI field in the last cell of the packet is not modified. This last method can be used to send VC-level OAM cells, which require that the PTI value of the cell being transmitted be set to a value of 10x. The TNETA1570 does not modify the PTI field for these cells.

PRINCIPLES OF OPERATION

transmit completion-ring select (bit 25)

This bit is set by the host to indicate whether the TNETA1570 uses the transmit completion ring with or without interrupt. It is also set when a buffer completes segmentation. If this bit is set to a one, the TNETA1570 uses the completion ring with interrupt. If this bit is cleared, the TNETA1570 uses the completion ring without interrupt.

buffer offset (bits 23–16)

The buffer offset (number of bytes offset from beginning of the payload area) is a pointer into the buffer that indicates to the segmentation unit where the real data begins. When a new buffer descriptor is imported by the TNETA1570, the TNETA1570 should store it in TX DMA state table entry memory. The segmentation unit must add the buffer offset to the start address to create the initial current-buffer-data address. An offset of 0 means that the data starts immediately.

current-buffer-data-byte count (bits 15–0)

This field indicates the current number of bytes in the buffer undergoing segmentation. EOP processing is triggered by this field reaching zero and the end-of-chain bit being set.

word 1

Current-Buffer-Data Address (bits 31–0)

The current-buffer pointer is set to a value equal to the start of the buffer plus four 32-bit words (two 64-bit words) when segmentation of a new buffer begins. After that, the pointer is adjusted to point to the current-data location after each transfer of payload data from the host.

word 2

GFC (bits 31–28)	VPI (bits 27–20)	VCI (bits 19–4)	PTI (bits 3–1)	CLP (bit 0)

Word 2 is copied directly from word 2 of the descriptor entry. This word is copied only from the first buffer in a packet and is ignored on the subsequent buffers.

word 3

ON (bit 31)	Reserved (bit 30)	Packet-Segmentation-Ring Base-Offset Pointer (bits 29–8)	Index 0–255 (bits 7–0)

This entry must be initialized by the host at startup and prior to the transmit operation being enabled.

ON (bit 31)

The DMA state on this bit is used to turn a particular DMA state on and off. This bit is initialized by the host.

packet-segmentation-ring base-offset pointer (bits 29–8)

This field contains the base address of the packet-segmentation ring assigned to this transmit DMA channel. Each segmentation ring contains up to 256 32-bit entries. The segmentation-ring queues are located on 1024-byte boundaries ($256 \times 4 = 1024$). This field must be initialized by the host at startup. This 22-bit field contains the most-significant bits of the packet-segmentation pointer.

index 0–255 (bits 7–0)

This field contains the index to the packet-segmentation-ring entry. This field is incremented by the TNETA1570 after each access to the segmentation ring and compared with the value in the packet-segmentation ring-size register. If the index field is greater than the value in the size register, the index field is reset to zero prior to being written back to control memory. The address to the segmentation ring is generated by setting the lower two bits of the address to zero (word accesses only are permitted) and appending the index and the segmentation-queue-base-offset pointer as shown below (x indicates concatenation):

Address = (packet-segmentation-ring base-offset pointer) x (index) x (00)

PRINCIPLES OF OPERATION

word 4

Start-of-Next-Buffer Address (bits 31–0)

This field contains the address of the next buffer in the transmit buffer chain to be segmented. The buffer is either 32 bit or 64 bit aligned. This word is copied from word 1 of the descriptor entry and shifted left two bits to form a complete 32-bit address.

word 5

Start-of-Next-Buffer Address (bits 31–0)

This field is copied from word 4 prior to the start of segmentation and provides a place holder for the entry written in the transmit completion ring. The transmit completion ring is updated when the segmentation of a buffer is complete.

word 6

Partial-AAL5 Transmit CRC (bits 31–0)

This field is used by the TNETA1570 to hold the interim AAL5-packet CRC calculation.

word 7

AAL5-Control Field (bits 31–16)	AAL5-Length Field (bits 15–0)

The AAL5-control field is copied directly from word 3 of the descriptor entry. The AAL5-length field must be immediately calculated by the TNETA1570 because segmentation can begin before all of the packet is in memory.

transmit packet-segmentation rings

Addressing	In host memory. The entries are addressed using the TX DMA state-table entry.
Number of rings supported	One ring per TX DMA channel (i.e., max = 1024)
Size: set in the TX packet-segmentation ring-size register	Max logical entries: 256 Max 32-bit words: 256 Min logical entries: 1 Min 32-bit words: 1

The transmit packet-segmentation rings are located in host memory and provide the address of the first buffer of the next packet to be segmented. Each transmit DMA channel has its own packet-segmentation ring. The size of the rings is programmable through the transmit packet-segmentation ring-size register, up to a maximum of 256 entries per ring. The structure of a packet-segmentation ring is shown below:

ENTRY	DESCRIPTION
Word 0	Control, buffer pointer

The definitions of the fields are as follows:

word 0

OWN (bit 31)	Reserved (bit 30)	Pointer (bits 29–0)

PRINCIPLES OF OPERATION

OWN (bit 31)

This bit is set by the host indicating that the first buffer of a packet is ready for segmentation. This bit and the rest of the 32-bit word is cleared by the TNETA1570 after segmentation of the packet is complete indicating that the entry is available for use by the host.

pointer (bits 29 – 0)

This field is set by the host and contains the upper 30 bits of the 32-bit address of the first buffer of a packet that is ready for segmentation. The lower two address bits are always set to zero because the data buffers are aligned to a 32-bit boundary. These bits and the OWN bit are cleared by the TNETA1570 after segmentation of the packet is complete indicating that the entry is available for use by the host.

transmit data buffers

Addressing	First buffer in new packets: address stored in transmit packet-segmentation ring New buffer in on-going packet: address stored in previous buffer descriptor On-going buffer: address stored in TX DMA state table
Size	Min size: buffer descriptor + 1-byte payload data Max size: buffer descriptor + 65535-byte payload data
Packets (buffer chains) supported	Max number of packets simultaneously on-going segmentation: 1023 Max number of queued new packets on one transmit packet-segmentation ring: 256 Max total number of queued new packets: 256 x 256 = 65536

The descriptor entry and the user data are both placed into the data buffer (located in host memory) for segmentation. The descriptor entry consists of four 32-bit words followed by the user data. The buffer size of the user data is user programmable from 1 byte to 64K bytes using the data-byte-count field in the buffer descriptor. The size is programmable on a per-DMA-channel basis. The total transmit data buffer must be at least five (4 plus 1) words and contain at least one byte of data. The maximum total size is 16388 (4 plus 16384) words and contain 65535 bytes of data.

Descriptor Entry (four 32-bit words)
User Information

The descriptor entry contains four 32-bit words as described below:

ENTRY	DESCRIPTION
Word 0	Control field, buffer offset, buffer data-byte count
Word 1	Start-of-next-buffer pointer
Word 2	4-byte ATM header
Word 3	PCS-UU/CPI field (AAL5 control field)

NOTE: A possible race condition can occur when updating the descriptor field, unless the OWN bit is in the last field to be updated in the descriptor field. The race occurs if the OWN bit gets updated and then the write operation is discontinued by the PCI bus before the complete descriptor has been updated.

The definitions of the fields inside each word follow:

word 0

Control (bits 31 – 25)	Reserved (bit 24)	Buffer Offset (bits 23 – 16)	Buffer-Data-Byte Count (bits 15 – 0)

PRINCIPLES OF OPERATION

RDY (bit 31)

This bit indicates whether the buffer is ready for segmentation. The host sets this bit to a one to indicate that the buffer is ready to be transmitted. The TNETA1570 does not write to this bit.

The RDY bits of the buffers within a packet, which has a cell divided across multiple buffers, is set in a zipper-up fashion, (i.e. the RDY bit of the last buffer in the chain is set first). If the last cell in the buffer has its beginning in one buffer and the ending in another, the RDY bit for the first buffer cannot be set before the RDY bit for the second buffer is set. This propagates down the buffer chain until a cell is completed on an even buffer boundary.

start of packet (SOP) (bit 30)

This bit is set by the host indicating that this is the first buffer of a packet that consists of one or more buffers. This bit is also set if a packet consists of only a single buffer.

end of packet (EOP) (bit 29)

This bit is set by the host to indicate that this is the last buffer of a packet. If a packet consists of only one buffer, both the SOP and EOP bits are set. If a packet consists of multiple buffers, the SOP bit is set on the first buffer and the EOP bit is set on the last buffer.

abort (bit 28)

When this bit is set by the host, the TNETA1570 stops segmentation of the packet and performs EOP processing immediately. An entry is written into the transmit completion ring indicating that an abort has occurred. When an abort occurs, the AAL5-length field in the AAL5 trailer of the last cell transmitted is set to zero to indicate that an abort has occurred.

transmit AAL-packet-type indicator (bits 27 and 26)

These bits are set by the host to indicate the AAL type of the packet being transmitted. Three options are available for processing a packet for transmission as shown in the following table.

BIT 27	BIT 26	INDICATION
0	0	Transparent-AAL packet with PTI bits set to indicate EOP
0	1	AAL5
1	0	Transparent-AAL packet without PTI bits set to indicate EOP
1	1	Invalid

AAL5 processing includes the calculation of the 32-bit CRC and the addition of the pad, control, and length fields. All buffers associated with an AAL5 packet must have the AAL5-indication bits set. When either of the other two processing methods is chosen, the TNETA1570 does not perform packet-level processing except for adding the necessary pad to complete a 48-byte payload. If both bits 27 and 26 are set to zero, the device also sets the PTI field in the last cell of a packet to a value of 0 x 1 to indicate EOP. If bits 27 and 26 are set to a value of 10, the PTI field in the last cell of the packet is not modified. This last method can be used to send VC-level OAM cells, which requires that the PTI value of the cell being transmitted be set to a value of 10x. The TNETA1570 does not modify the PTI field for these cells.

transmit completion-ring select (bit 25)

This bit is set by the host to indicate whether the TNETA1570 uses the transmit completion ring with or without interrupt when a buffer completes segmentation. If this bit is set to a one, the TNETA1570 uses the completion ring with interrupt. If this bit is cleared, the TNETA1570 uses the completion ring without interrupt.

PRINCIPLES OF OPERATION

buffer offset (bits 23–16)

The buffer offset (number of bytes offset from beginning of the payload area) is a pointer into the buffer that indicates to the segmentation unit where the real data begins. When a new buffer descriptor is imported by the TNETA1570, the TNETA1570 stores it in TX DMA state table-entry memory. The segmentation unit must add the buffer offset to the start address to create the initial current-buffer-data address. An offset of 0 means that the data starts immediately.

buffer-data-byte count (bits 15–0)

This field is set by the host to indicate the number of bytes in the buffer that are to be segmented. The maximum buffer size is 64K bytes, allowing an entire packet to be placed in a single buffer. Minimum size is one byte.

word 1

00	Start-of-Next-Buffer Pointer (bits 29–0)

This word provides the upper 30 bits of the 32-bit address of the next data buffer in the buffer chain; the lower two bits of the address are always set to zero. If the buffer being segmented is the last buffer in the chain, this entry is not used by the TNETA1570. The data buffer (with descriptor entry) is aligned to either a 32-bit or 64-bit boundary, depending on whether the 64-bit or 32-bit PCI-host interface is selected.

word 2

GFC (bits 31–28)	VPI (bits 27–20)	VCI (bits 19–4)	PTI (bits 3–1)	CLP (bit 0)

This word provides the 4-byte ATM header. The PTI field is changed in the last cell of the packet to indicate EOP (i.e., PTI = 0×1). This word is only read into the TX DMA state table for the first buffer in a packet. For subsequent buffers, this word is not copied. The AAL5 ANSI and ITU-T specifications require that the last cell in an AAL5 packet set the PTI field to xx1 to indicate EOP.

word 3

CPCS-UU (bits 31–24)	CPI (bits 23–16)	User defined (bits 15–0)

CPCS-UU (bits 31–24), CPI (bits 23–16)

This field is set by the host and copies directly into the control field of the outgoing AAL5 trailer. This field is only copied over for the first buffer in a packet. For subsequent buffers, this field is ignored.

user defined (bits 15–0)

This field is available for use by the user and is not used or modified by the TNETA1570.

transmit completion rings with/without interrupt

Addressing	In host memory. The entries are addressed using the TX completion-ring pointers.
Size: set in the TX completion-ring-size registers	Max logical entries: 8192 Max 32-bit words: 8192 Min logical entries: 1 Min 32-bit words: 1

The structure of the transmit completion rings with/without interrupt located in host memory follows. An entry is posted to the completion ring when a buffer completes segmentation. The starting address of the buffer completing segmentation is written into the transmit completion ring. When segmentation of a buffer is completed and the transmit completion ring with interrupt has been selected and updated, an interrupt is generated on either a buffer or packet basis, depending upon the state of the bit in the configuration register. If the transmit completion ring without interrupt has been selected and updated, an interrupt is not generated.

PRINCIPLES OF OPERATION

transmit completion rings with/without interrupt (continued)

ENTRY	DESCRIPTION
Word 0	Control, buffer pointer

The definitions of the fields are as follows:

word 0

OWN (bit 31)	ABORT (bit 30)	Pointer (bits 29 – 0)

OWN (bit 31)

This bit is set by the host indicating that the entry is available for use by the TNETA1570. This bit is cleared by the TNETA1570 indicating that the entry is owned by the host. If the OWN bit is not set when the TNETA1570 attempts to post an entry, TX freeze occurs.

ABORT (bit 30)

This bit is set by the TNETA1570 indicating that segmentation of the packet was aborted due to the ABORT bit being set in the transmit-descriptor entry.

address (bits 29 – 0)

This field contains the upper 30 bits of the buffer address that has completed segmentation. The lower two bits are always zero because the buffer is aligned to a 32-bit boundary.

high-priority-segmentation service

The TNETA1570 provides a high-priority-segmentation service that bypasses the scheduler table and the normal segmentation flow. This method can be used to insert constant bit-rate traffic into the transmit data stream by using a method other than the scheduler table. The operation of the high-priority-segmentation service requires the use of two external pins, the high-priority-segmentation request (HPSREQ) input and the high-priority-segmentation acknowledge (HPSACK) output.

HPSREQ is sampled by the TNETA1570 at each new segmentation opportunity, synchronous to the PCI-bus clock. When the input is active, the TNETA1570 initiates the procedure for transmitting a cell from TX DMA channel 1.

NOTE: This method requires that transmit DMA channel 1 not be used for any other data traffic and that the number 1 does not appear in the scheduler table.

Normal scheduler table look-up and pointer-increment processing is suspended until the next segmentation opportunity for a new cell. HPSACK is asserted for one PCI-bus-clock cycle. HPSREQ must be deasserted within two PCI-bus-clock cycles of HPSACK being asserted to ensure that another cell is not segmented for TX DMA channel 1. If HPSREQ is not deasserted within two PCI-bus clock cycles of HPSACK being asserted, another cell is sent from TX DMA channel 1. To ensure that the high-priority-segmentation request is processed, HPSREQ must remain active until HPSACK is set low.

ATM header-error-check (HEC) byte generation

The TNETA1570 generates the HEC byte in the 5-byte ATM header on all outgoing ATM cells in accordance with the appropriate specifications (ANSI T1.624, CCIFT 1.432, and ATM Forum UNI specification version 3.0). To facilitate testing and functional verification, a mechanism to error the HEC byte in the outgoing cells is made available in the configuration register. When the HEC bit in the configuration register is set, the HEC byte is inverted prior to insertion into the outgoing cell.

PRINCIPLES OF OPERATION

transmit loopback

The ability to loopback the segmentation operation to the reassembly operation is helpful in system debug and verification. This feature is enabled through a bit in the configuration register. It is important to follow the procedure described below to ensure error-free operation.

The transmit outputs are disabled during loopback to prevent the transfer of data to the downstream element during testing. The receiver inputs are ignored when loopback is enabled. To place the TNETA1570 in loopback mode, a device-software reset should be executed to place the SAR in a known state. After reset is executed (ten clock cycles), the LOOP_BACK bit is set by the host in the configuration register. In a following write instruction by the host to the configuration register, the EN_TX and EN_RX bits can be set. It is important to set the EN_TX and EN_RX bits in separate write instructions from the LOOP_BACK bit to allow a clock exchange for the reassembly UTOPIA bus.

After loopback-mode testing is complete, the device should be reset again before resuming normal operation. In loopback mode, the UTOPIA bus clock for the segmentation side of the device is also used to internally control the reassembly-side UTOPIA clock. This allows loopback operation when the two external clocks operate asynchronously.

reassembly operation

The receive circuitry provides support for the simultaneously reassembly of 30720 packets, hardware timers for packet aging, and receive-buffer chaining.

receive VPI/VCI DMA pointer table

Addressing	Control-memory address 01000 – 01FFF. The VPI field of the incoming ATM cell is used as address to the entry, thereby initiating the reassembly engine.
Size	Logical entries: 4096 32-bit words: 4096

The receive VPI/VCI DMA pointer table resides in control memory and is used to assign a receive DMA state-table entry to an incoming virtual connection. The VPI index from the header of an incoming cell is used to form the address for the VPI/VCI DMA pointer-table entry. The VPI/VCI DMA pointer-table entry contains the range of VCIs that are currently active for that particular VPI index, as well as a base pointer to the receive DMA state table. If the VCI index of the incoming cell is outside the range of VCIs currently active for the corresponding VPI, the cell is dropped, the VPI/VCI index is captured in the receive-unknown register, and an interrupt is generated (the interrupt may be masked through the interrupt-mask register). If the VCI index of the incoming cell is within the range of VCIs currently active for the corresponding VPI, the VCI index is shifted left three bits and added to the base pointer to obtain the address of the receive DMA state-table entry assigned to that particular virtual connection. Since there are 12 VPI bits in an ATM header, this table contains 4096 entries. The entries in the RX VPI/VCI DMA pointer table must be initialized by the host at startup and are shown below:

ENTRY	DESCRIPTION
Word 0	Control, base pointer, VCI range

The definitions of the fields inside each word are as follows:

word 0

Enable (bit 31)	Base Pointer (bits 30 – 16)	Valid VCI Range (bits 15 – 0)

PRINCIPLES OF OPERATION

enable (bit 31)

This bit is set by the host indicating that packets can be received on this VPI channel. When this bit is cleared, packets received on this VPI channel are dropped.

base pointer (bits 30–16)

This field is initialized by the host indicating the base address of the area in the RX DMA state table in control memory assigned to this VPI channel. The value in this field is used together with the valid VCI range (next entry in this register) to form the address in the RX DMA state table. Valid values are 800–7FFF(h).

valid VCI range (bits 15–0)

This field contains the upper value of the range of the VCI channels that can be received on this VPI channel. The lower value of the range is always zero. The VCI value of the incoming cell is compared with this field. If this value is less then or equal to the valid VCI range, the cells for this VCI are accepted. The VCI is masked before it is compared. Valid values are 0 – F00(h).

The address to the RX DMA state table is calculated as follows:

[[(base pointer + VCI value of incoming cell + value at base register 0) $\times$ 8] $\times$ 4] (i.e., shift left five times).

This value never exceeds the value of the maximum address in the control-memory map.

VCIs causing values greater than the value of the maximum address in the control-memory map can be transferred to the data buffers if a VCI mask is used, which reduces the value used by the reassembly engine to a value less than the maximum control-memory address.

receive DMA state table

Addressing	Control-memory address 04000–3FFFF. The VCI value of the incoming ATM cell combined with the value of the entry in the RX VPI/VCI pointer table is used to form the address.
Size	Max logical entries: 30720 Max 32-bit words: 245760 Min logical entries: 1 Min 32-bit words: 8

The receive DMA state table resides in control memory and contains 30720 entries, allowing for the simultaneous reassembly of 30720 packets. Each state entry contains eight 32-bit words. The receive DMA state table has several entries that must be initialized, including whether that particular state is on or off, etc. The entries in the receive DMA state table are shown below.

ENTRY	DESCRIPTION
Word 0	Control field, EFCN cell counter, packet length
Word 1	Current buffer address
Word 2	Start-of-buffer address
Word 3	Current transparent-AAL packet counter, current buffer length
Word 4	Start-of-packet pointer
Word 5	Partial AAL5 receive CRC
Word 6	On filter bit, FIFO/RX free-buffer ring-pointer table entry
Word 7	Time-out value, current time-out count

PRINCIPLES OF OPERATION

entries of receive DMA state table

The definitions of the fields within each word entry are given below:

word 0

Config (bits 31–25)	Reserved (bits 24–22)	EFCN Cell Counter (bits 21–11)	Current-Packet Length (bits 10–0)

ACT (ACTive) (bit 31)

This bit is set by the TNETA1570 to indicate that this DMA channel is active and a packet reassembly is currently underway. This bit is cleared by the TNETA1570 when the EOP bit is detected and the reassembly of a packet is complete.

reserved (bit 30)

receive complete-interrupt posting (bit 29)

When this bit is set, the receive completion ring with interrupt is used for posting a completion notice when reassembly of a packet is complete. When this bit is cleared, the receive completion ring without interrupt is used. This bit must be initialized by the host.

wait for EOP (bit 28)

This bit is set by the TNETA1570 when either a packet time-out or packet overflow condition is detected. All packet reassembly is then disabled until an EOP cell is detected. This bit also is set if a free buffer is not available for reassembling an incoming packet. The TNETA1570 clears this bit when the cell containing the EOP indication is received.

packet-type indicator (bits 27 and 26)

These bits are set by the host to indicate whether AAL5, PTI-based transparent-AAL, or counter-based transparent-AAL processing is performed on the incoming packet. The following values are defined for these bits:

BIT 27	BIT 26	INDICATION
0	0	PTI-based transparent-AAL packet
0	1	AAL5 packet
1	0	Counter-based transparent-AAL packet
1	1	Invalid

FIFO/free-buffer ring indicator (bit 25)

This bit is set by the host to indicate that the TNETA1570 uses the FIFO structure to fetch free buffers for this receive channel. When this bit is cleared by the host, the TNETA1570 uses the free-buffer-ring structure to fetch free buffers for this receive channel.

EFCN-cell counter (bits 21–11)

This field counts the number of cells that are received with the explicit forward congestion notification indicator (EFCN) in the PTI bits set (i.e., the PTI field is set to 01x). This field is cleared after reassembly of a packet is complete. This field is updated by the TNETA1570. The counter value is one greater than the actual value (actual + l), if the TNETA1570 experiences buffer starvation and the last received cell has the EFCN flag set.

current-packet length (bits 10–0)

This field counts the number of cells received on this DMA channel. If the value in this field exceeds 1366 cells before an EOP cell is detected, a packet-overflow error occurs, bit 28 is set, and EOP processing is initiated. This field is only valid for AAL5 packets.

PRINCIPLES OF OPERATION

word 1

Current-Buffer Pointer (bits 31–0)

The current-buffer pointer is set by the TNETA1570 to a value equal to the start-of-buffer pointer plus four 32-bit words when reassembly of a new buffer begins. After that, the pointer is adjusted to point to the current-data location after each payload-data transfer to the host.

word 2

Reserved	Start-of-Buffer Pointer (bits 29–0)

Word 2 is copied from word 0 of the RX free-buffer-ring entry or RX free-buffer FIFO entry.

word 3

Current Transparent-AAL Packet Counter (bits 31–16)	Current-Buffer Length (bits 15–0)

current transparent-AAL packet counter (bits 31–16)

This field is copied from word 5 bit (bits15–0) (see word 5 definition) when the first cell of a transparent-AAL packet arrives. This field is decremented every time a cell arrives until its value is equal to zero. Once this field's value reaches zero, EOP processing for a transparent-AAL packet is initiated. This field is used only if the receive DMA channel is configured to receive counter-based transparent-AAL packets (i.e., bit 27 of receive DMA state table entry word 0 has been set to a one and bit 26 has been set to a zero).

current-buffer length (bits 15–0)

This field is copied either from word 1 of the RX free-buffer ring-pointer table entry or FIFO free-buffer-size register, dependent upon whether the RX free-buffer descriptor ring or FIFO is utilized. This field is decremented every time a cell is received until it reaches zero. If the EOP cell has not arrived at that time, a new buffer is fetched from the free-buffer ring or FIFO.

word 4

Reserved	Start-of-Buffer Pointer (bits 29–0)

Word 4 is copied from word 0 of the receive free-buffer ring-pointer-table entry for only the first buffer in each new packet. When the received packet is complete, this value is written to the receive complete-ring-descriptor entry,

word 5 (AAL5 packets)

Partial-AAL5-Receive CRC (bits 31–0)

This field is used by the TNETA1570 to save the interim results of the 32-bit CRC for AAL5 packets when the receive DMA channel is configured to receive AAL5 packets.

transparent-AAL packets using counter-based reassembly

Reserved (bits 31–16)	Transparent-AAL Packet Length (bits 15–0) (counter base only)

transparent-AAL packet length (bits 15–0)

This field contains the number of cells that constitute a transparent-AAL packet when the receive DMA channel has been configured to receive transparent-AAL packets using counter-based reassembly. This field must be initialized by the host at startup.

PRINCIPLES OF OPERATION

word 6

ON (bit 31)	FILTER (bit 30)	Free-Buffer FIFO Pointer/RX Free-Buffer Pointer-Table Entry Index (bits 29 – 0)

ON (bit 31)

When this bit is set, packet reassembly is enabled for this receive DMA channel. OAM cells can only be received on DMA channels with the ON bit set (=1). This bit must be set by the host at initialization and not be turned off during ongoing reassembly of a packet, as data buffers might be lost.

FILTER (bit 30)

This bit is only valid when the ON bit = 0 for the DMA channel. When the FILTER bit is set (= I), the following happens: The cell is dropped, no interrupts are generated, and the unknown protocol counter is not incremented. When the FILTER bit is not set (= 0), the following happens: The cell is dropped, the ATM header is posted in the RX unknown register, interrupts are generated as they are done if the VCI is out of range, and the unknown-protocol counter is incremented.

free-buffer FIFO pointer/RX free-buffer pointer-table index (bits 29 – 0)

This entry contains either the address of the FIFO free-buffer entry or an index to the RX free-buffer ring-pointer table contained in control memory. Only the lower eight bits are used if the RX free-buffer rings are selected. The upper 22 bits remain unchanged and retain the initialized value. The free-buffer FIFO pointer/descriptor-index field must be initialized by the host only at startup.

word 7

Reserved (bit 31)	Reserved (bits 30 – 24)	Time-Out Value (bits 23 – 12)	Time-Out Count (bits 11 – 0)

time-out value (bits 23 – 12)

This field contains the value at which a reassembly time-out occurs. This field must be set at initialization.

time-out count (bits 11 – 0)

This field is incremented if the reassembly-aging mechanism has reached this DMA channel and the RX DMA channel is ACTive. If the value in this field is equal to the time-out value, a reassembly time-out is declared and EOP processing is initiated. This field is reset at SOP.

receive free-buffer ring-pointer table

Addressing	In control memory. The entries are addressed using the RX DMA-state-table entry.
Size	Logical entries: 256 32-bit words: 512

The receive free-buffer ring-pointer table resides in control memory and provides the addresses of the next entries in the receive free-buffer rings. There are a maximum of 256 free-buffer rings. The size of the receive free-buffer rings are programmable. The entries for the receive free-buffer ring-pointer table are shown below:

ENTRY	DESCRIPTION
Word 0	RX free-buffer ring pointer
Word 1	Ring size, buffer size

The definitions of the fields associated with each word entry are given below:

word 0

RX Free-Buffer Ring Pointer (bits 31 – 2)	Reserved (bits 1 – 0)

PRINCIPLES OF OPERATION

RX-free buffer-ring pointer (bits 31 – 2)

This field holds the 30 most significant bits of the base address of the RX free-buffer rings in host memory. The two least significant bits are always assumed to be zero. This field is initialized by the host.

word 1

Buffer Size (bits 31 – 16)	Ring Size (bits 15 – 10)	Current Index (bits 9 – 0)

buffer size (bits 31–16)

This field gives the size of the buffers associated with this free-buffer ring in number of ATM cells. This field is set by the host at initialization.

ring size (bits 15 – 10)

This field is set by the host at initialization and gives the number of entries in the free-buffer ring in multiples of 16. The maximum number of entries in a free-buffer ring is 1024. The actual value for the ring size can be found by shifting the value in the ring-size field left four places and filling the lower four bits with 1s. Shifting 000 00 111 left four places gives a value of 000 00 111111. For example, a value of 000 0011 indicates that the ring contains 64 entries (3 x 16 +15 = 63 +1 = 64) (the last 1 is needed to account for the zero entry).

The value for the ring-size field is found by: [(ring_size/16) – 1]. For a ring size of 16, the value of the ring-size field in word 1 is 0.

current index (bits 9 – 0)

This field is set to all zeroes by the host at initialization and contains the entry in the currently active free-buffer ring. This field is updated after it is read by the TNETA1570 and the new value written back to control memory. When the value in the current index exceeds the ring-size value, indicating that the end of the ring has been reached, the current index is reset to all zeroes.

receive free-buffer ring/receive free-buffer FIFO

receive free-buffer ring

Addressing	The value in the free-buffer ring-pointer table is used to form the address.
Size	Max logical entries: 1024 Min logical entries: 16 32-bit words: 1024 Max number of rings supported: 256 Min number of rings supported: 0 (only use free-buffer FIFO)

receive free-buffer FIFO

Addressing	The value in the RX DMA state table is used to form the address.
Size	Logical entries: 1

The TNETA1570 is able to fetch receive free buffers by using one of two possible methods. The first method uses multiple free-buffer rings, up to a maximum of 256. The free-buffer ring pointers are maintained in control memory. The number of entries in the free-buffer rings is programmable. Each entry in the RX DMA state-table contains a pointer to the free-buffer ring associated with that particular RX DMA state-table entry. When the TNETA1570 accesses the free-buffer ring entry, it reads the address of an available receive free buffer. The size of the receive free buffer is contained in the receive free-buffer ring-pointer-table entry in control memory.

The second method uses a FIFO-based mechanism. Each entry in the RX DMA state table contains an address pointer to a FIFO located in the host-memory space. When the TNETA1570 reads the contents of the FIFO located at the given address, the starting address of an available buffer is returned. For the FIFO-based mechanism, the size of the receive free buffer is programmed through an internal register. All buffers used with this mechanism must be the same size.

PRINCIPLES OF OPERATION

receive free-buffer FIFO (continued)

The FIFO/free-buffer ring entry consists of one 32-bit word as shown below:

ENTRY	DESCRIPTION
Word 0	Control, free-buffer pointer

The definitions of the fields contained in each word entry are given below:

word 0

Control (bits 31–30)	Start-of-Next-Data-Buffer Pointer (bits 29–0)

OWN (bit 31)

This bit is set by the host to indicate that the buffer is available for use by the TNETA1570. The TNETA1570 uses the free-ring entry if the OWN bit is set. If the OWN bit is not set, buffer starvation occurs. The TNETA1570 clears the OWN bit after acquiring the buffer, releasing the ring location to the host. For the FIFO-based mechanism, the TNETA1570 does not write back to the entry. The FIFO implementation must ensure that OWN is set properly to indicate whether the entry in the free-buffer FIFO is valid or not.

reserved (bit 30)

start-of-next-data-buffer pointer (bits 29–0)

This field contains the address of an available buffer, aligned on either a 32- or 64-bit boundary, depending upon whether a 32- or 64-bit PCI interface is selected. Only the upper 30 bits of the address are written to this entry. The lower two address bits are always zero.

receive data buffers

Addressing	New buffer using free-buffer rings. The address is stored in the RX free-buffer ring. New buffer using free-buffer FIFO. The address is stored in the RX free-buffer FIFO. Ongoing buffer: Address stored in RX DMA state table
Size	Min size: buffer descriptor + 48-byte payload data Max size: buffer descriptor + 3145728-byte payload data
Buffers/packets (buffer chains) supported	Max number of packets simultaneously being reassembled: 30720 Max number of queued free buffers on one RX free-buffer ring: 1024 Max number of queued free buffers on all RX free-buffer rings: 1024 × 256 = 262144 Max total number of queued free buffers on one RX free-buffer FIFO: limited by the FIFO size

The data-buffer-descriptor entry and the received user data are both placed in the data buffer as shown. The data-buffer-descriptor entry consists of four 32-bit words and the user data contains (N x 48) bytes of information, except for the last buffer in the chain; therefore, the total size of the data buffer, except for the last buffer in the buffer chain, is (N x 48) + 16 bytes. The value of N is greater than zero. This requirement ensures that all buffers except the last buffer in the chain contains an integral number of ATM-cell payloads. The buffer size is user configurable from 48 byte to 3.1M byte of user data. The total receive-data buffer must be at least 64 bytes and contain at least 48 byte of data. The maximum total size is 3145726 bytes.

Data-buffer descriptor entry (four 32-bit words)
User information [(N x 48) bytes]

PRINCIPLES OF OPERATION

receive data buffers (continued)

The descriptor entry contains four 32-bit words:

ENTRY	DESCRIPTION
Word 0	Reserved
Word 1	Control field, start-of-next-buffer pointer
Word 2	4-byte ATM header
Word 3	User defined

The definitions of the fields inside each word are:

word 0

Reserved (bits 31 – 0)

This word is not used by the TNETA1570 and reserved for future revisions of the TNETA1570.

word 1

Control (bits 31–30)	Start of Next RX Data-Buffer Pointer (bits 29–0)

start of packet (bit 31)

This bit is set by the TNETA1570 to indicate that this is the first buffer in the received packet.

end of packet (bit 30)

This bit is set by the TNETA1570 to indicate that this is the last buffer in the received packet.

start of next RX data-buffer pointer (bits 29 – 0)

This word provides the location of the next data-buffer in the data-buffer chain. If this is the last data-buffer in the chain, this word is zero (0). This pointer is set by the TNETA1570.

word 2

ATM Header Byte 1	ATM Header Byte 2	ATM Header Byte 3	ATM Header Byte 4

This word contains the first four header bytes from the ATM packet undergoing reassembly.

word 3

User Defined (bits 31 – 0)

user defined (bits 31 – 0)

This field is not overwritten by the TNETA1570 and is available for use by the user.

receive completion rings

Addressing	In host memory. The entries are addressed using the RX completion-ring pointers.
Size: set in the RX completion-ring-size registers	Max logical entries: 1024 Max 32-bit words: 8192 Min logical entries: 1 Min 32-bit words: 8

The TNETA1570 provides support for two RX completion rings that reside in host memory: the receive completion ring with interrupt and the receive completion ring without interrupt. The difference between the rings is that when an entry is posted to the receive completion ring with interrupt an interrupt is generated, and when an entry is

PRINCIPLES OF OPERATION

posted to the receive completion ring without interrupt, an interrupt is not generated. The receive completion ring without interrupt must be polled by the host to determine if a packet has been posted to this descriptor ring. Both rings use the same format. Each descriptor entry contains eight 32-bit words as described below.

The definitions of the fields in the word entries are given below:

word 0

ATM Header Byte 1	ATM Header Byte 2	ATM Header Byte 3	ATM Header Byte 4

This word provides the 4-byte ATM header from the last cell in the reassembled packet (unless there has been a packet-aging time-out when the VPI/VCI value is provided in bits (31–4), and bits (3–0) are set to zero).

word 1

Error Indicator (bits 31–26)	Reserved (bits 25–10)	Congestion-Cell Count (bits 9–0)

packet overflow (bit 31)

This entry is set by the TNETA1570 indicating that no EOP indicator was found in the packet being reassembled before the packet-length entry in control memory exceeded its maximum value of 1366 (number of cells received). This bit is valid only for AAL5 packets.

CRC-error condition (bit 30)

This bit is set by the TNETA1570 when the CRC value calculated on the incoming packet does not match the CRC value contained in the AAL5 trailer. This bit is valid only for AAL5 packets.

buffer starvation (bit 29)

This bit is set by the TNETA1570 when reassembly of a packet is terminated because no receive free buffers are available.

reassembly time-out (bit 28)

This bit is set by the TNETA1570 when the reassembly-aging timer reaches its maximum value before an EOP indication occurs on an incoming packet.

abort condition (bit 27)

This bit is set by the TNETA1570 when the AAL5-length field of the incoming packet is set to zero. This bit is valid only for AAL5 packets.

AAL5 indicator (bit 26)

This bit is set by the TNETA1570 to indicate that the received packet is an AAL5 packet. The packet-type-indicator bits in word 0 of the receive DMA state table entry determine whether a packet is an AAL5 packet.

congestion-cell count (bits 9–0)

This field is copied directly from the upper ten bits of word 0 of the RX DMA state table and contains the number of cells received with the EFCN indicator set. The EFCN indication is given by a value of 01x in the PTI field of the ATM header.

word 2

VALID (bit 31)	Reserved (bit 30)	Start-of-Packet Pointer (bits 29–0)

PRINCIPLES OF OPERATION

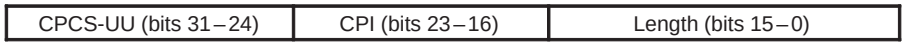
VALID (bit 31)

The VALID bit is set to 1, except when the start-of-packet pointer is not valid. This condition occurs if a cell is received on a DMA channel that is in buffer starvation, (i.e., when reassembly has started and the first buffer in a packet is not available.

start-of-packet pointer (bits 29 – 0)

This word provides the location of the first data-buffer in the receive data-buffer chain. This word is copied directly from word 4 of the RX DMA state table. Only the upper 30 bits of the 32-bit address are copied to this entry.

word 3 (AAL5 trailer)



AAL5 trailer (bits 31 – 0)

The CPCS-UU field, the CPI field, and the length field from the AAL5 trailer of an incoming packet are copied to this entry. This field is valid only in AAL5 mode.

word 4

OWN (bit 31)

This bit is set to a one by the host to indicate that this entry is available for use by the TNETA1570. If the OWN bit is not set when the TNETA1570 tries to access an entry, RX freeze occurs. The TNETA1570 clears this bit to indicate that it has completed the entry and that the entry is ready for the host to evaluate.

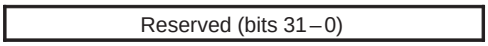
error entry (bit 30)

This bit is set by the TNETA1570 when one or more of the error conditions indicated by bits 27–31 of word 1 occur.

FIFO-pointer/ring-descriptor index (bits 29 – 0)

This field is copied from word 6 of the receive DMA state table. The value of this field is used by the host to determine which ring/FIFO the ring is to be returned to when the buffers has been read.

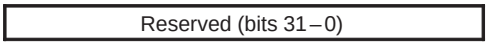
word 5



reserved (bits 31 – 0)

This word is not used by the TNETA1570 and reserved for future revisions of the TNETA1570.

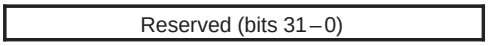
word 6



reserved (bits 31 – 0)

This word is not used by the TNETA1570 and reserved for future revisions of the TNETA1570.

word 7



reserved (bits 31 – 0)

This word is not used by the TNETA1570 and reserved for future revisions of the TNETA1570.

PRINCIPLES OF OPERATION

receive OAM-cell handling

The receive operation handles VP- and VC-level OAM cells. VC level OAM cells, which have a PTI value of 100, 101, 110, or 111, are routed to either receive DMA channel 0 or 1. VP-level OAM cells, which have a VCI value of 3 or 4, are routed to receive DMA channel 2. These channels must be configured by software to handle a counter-based transparent-AAL packet with a packet length of one cell. The assignment of the VP- and VC-level OAM cells is shown below:

OAM CELL	RX DMA CHANNEL ASSIGNMENT
VC level with PTI = 10x	Channel 0
VC level with PTI = 11x	Channel 1
VP level with VCI = 3 or 4	Channel 2

In addition, the RX VPI/VCI DMA pointer table must not assign receive DMA channels 0–2 for use in reassembling incoming packets. OAM cells do not cause the RX unknown register and unknown-protocol counter to be updated unless the VPI is not enabled or the RX DMA state table has the ON bit = 0 and filter = 0. F4 OAMs (VP termination) are identified by VCI 3,4. As long as the VPI is enabled, the range is ignored when identifying VCI 3,4 (the cells are always accepted). F5 OAMs (VC termination) are identified by PTI = 4 or 5. As long as the VPI is enabled, the cells are accepted regardless of the VCI range. The ON bits of the three dedicated OAM DMA channels should always be = 1. However, the TNETA1570 does not prohibit the user from setting the ON bit = 0 to disable OAM reception.

reassembly-aging timer

The TNETA1570 provides a reassembly-aging-timer (RAT) mechanism for determining the time elapsed since the start of AAL5 and PTI-based transparent packets and terminating their reassembly based on time-out values programmed per VC.

control structures

The RAT function is enabled globally by the EN_RAT bit in the device-configuration register. The reassembly-global-timer register (16 bits) and RAT-cycle-count register (15 bits) provide global control of this function. The RAT control functions on a per-VC basis are the time-out value (12 bits) and the time-out count (12 bits) in word 7 of the receive DMA state table entries.

There are four internal counters used for the RAT:

- The global-RAT timer. It decrements from the reassembly-global-timer-register value every PCI-clock cycle. When it reaches zero, a global time-out is declared. A global time-out is an opportunity to age a single VC.
- The RAT-cycle-count timer. It decrements from the RAT-cycle-count-register-value every global time-out. When it reaches zero, the reassembly-aging timer is reset and a new global-timer cycle is started.
- The VCI increments. It starts at zero when the RAT-cycle-count timer is reset and increments every global time-out until it reaches the max valid VCI value for the current VPI value.
- VPI increments. It starts at zero when the RAT-cycle-count timer is reset and increments every time the VCI incrementer reaches the max VCI value for the VPI, or if the VPI value points to a VPI that is not enabled.

global timer

An internal global counter decrements its count starting at the reassembly-global-timer-register value on each PCI-bus clock when the EN_RAT bit in the configuration register is set. For a 33-MHz PCI-bus clock, the 16-bit counter counts to zero in 1.966 ms or less, depending on the reassembly-global-timer-register value. When the timer reaches zero, a global time-out is declared.

PRINCIPLES OF OPERATION

reassembly time-out processing per VC

When the reassembly engine reaches the state to begin processing, the next received cell in the receive-cell queue , it first checks the aging timer to see if a global time-out has occurred. If a global time-out has occurred, a single-VC aging operation is performed before the next cell in queue is processed.

The reassembly-aging timer uses the receive VPI/VCI DMA pointer table to select a DMA channel for a single-VC aging operation. The VPIs for aging are selected one at a time in consecutive order starting at zero, providing one VPI/VCI combination the opportunity to be processed for each global time-out.

The single-VC operation starts with checking the enable bit for the next entry in the RX VPI/VCI DMA pointer table. If the VPI is disabled, the aging opportunity is terminated. If the VPI is enabled, the base pointer and VCI-incrementer value are used to form the base address of the receive DMA state entry to be processed.

For the aging opportunity to continue, the selected DMA entry must be active, not in the wait-on-EOP state, and be in use for either AAL5- or PTI-based transparent packets. If this is true, the time-out-count value of the entry is incremented and compared to the time-out value. If the reassembly time-out-count value is equal to the time-out value for the VC, the TNETA1570 device implements EOP processing on the packet associated with this RX DMA state table entry, including a write to the RX completion ring denoting that time-out has occurred.

terminating aging opportunity

At the termination of each aging opportunity, several control elements of the reassembly-aging timer are updated.

- The VCI value is incremented by one. If this value exceeds the valid VCI range for the VPI which points to the current RX VPI/VCI DMA pointer-table entry, or if the current RX VPI/VCI-DMA-pointer entry is not enabled, the VPI index is incremented on the next aging opportunity.
- The RAT cycle-count timer is also incremented by one at the end of each aging opportunity. This counter is compared to the RAT cycle-count register in the next aging opportunity. The VPI index is reset to zero beginning a new timer pass through the RX VPI/VCI DMA pointer table when these values are equal.

If the VPI index reaches 4095, the VCI index exceeds the valid VCI range for the VPI, and the internal RAT cycle-count-timer value is less than the RAT cycle-count register, aging opportunities are terminated immediately and the reassembly-aging timer waits for the RAT reset to occur.

formula for time-out time period

The time to time-out a packet on a per VC basis is determined by the following formula:

Time-out time period = (1/PCI clock) × (global RAT timer-register value) × (RAT cycle-count register value) × (time-out value in word 7 of the RX DMA state table)

receive idle cells handling

If the received cell is an idle cell, (i.e., a cell with the ATM header set to a binary value of xxxx 0000 0000 0000 0000 0000 0000 xxx 1), the ATM header, including the VPI/VCI value, is written to the RX unknown register, an interrupt is generated if enabled, and the unknown-protocols counter is increased.

cell interface

The TNETA1570 connects to the transmission logic through the cell interface. The cell interface is configured as a PHY interface with the option to be configured as an ATM interface. The operation of this dual interface requires the use of two external pins, ATMCLK and PHY/$\overline{\text{ATM}}$.

cell interface (continued)

When PHY/$\overline{\text{ATM}}$ is high (PHY mode), the cell-interface port configuration is defined as a PHY-RX port for the segmentation unit and the cell-interface port configuration is a PHY-TX port for the reassembly unit. The clock speed supported by the interface is up to 33 MHz nominal. The ATMCLK is connected in this mode and is configured as an output that is always low.

When PHY/$\overline{\text{ATM}}$ is low (ATM mode), the cell-interface port configuration is defined as an ATM-TX port for the segmentation unit and the cell-interface port configuration is an ATM-RX port for the reassembly unit. The ATMCLK is used as the clock source to provide for data transfers/synchronization. The ATMCLK has is connected to the SEGCLK and RESCLK in the ATM mode.

The interface is designed to work synchronously for an 8-bit data path. The interface uses either an octet-level handshake or a cell-level handshake. The clock generated by the interface is 33 MHz nominal, using the PCI clock.

segmentation-unit interface, PHY mode

The segmentation-unit interface on the TNETA1570 outputs RXDATA on the UTOPIA interface in PHY mode. This cell interface works on the low-to-high transition of SEGCLK to sample and generate signals. All signals are active high, unless denoted via a bar over the signal name.

SEGCLK

Clock (RXCLK) input. Data transfer/synchronization clock for synchronizing transfers on SEGDATA. When operating in PHY mode, the SEGCLK is sourced from the UTOPIA interface.

SEGDATA (bits 7–0)

Data (RXDATA) output. Byte-wide true data sourced by the TNETA1570. SEGDATA7 is the MSB.

SEGPAR

Parity (RXPRTY) output. SPAR is the odd-parity bit over SEGDATA.

SEGSOC

Start of cell (RXSOC) output. Active-high signal sourced by the TNETA1570 when SEGDATA contains the first valid byte of the cell.

$\overline{\text{SEGEN}}$

Enable ($\overline{\text{RXENB}}$) input. This active-low signal is received by the TNETA1570 when a valid byte SEGDATA is sent during the next clock cycle.

SEGCLAV

Empty/cell available ($\overline{\text{RXEMPTY}}$/RXCLAV) output. An indication that a complete cell is available.

reassembly-unit interface, PHY mode

The reassembly unit interface on the TNETA1570 inputs TXDATA from the UTOPIA interface in PHY mode. This receive cell interface works on the low-to-high transition of RESCLK to sample and generate signals. All signals are active high unless denoted via a bar over the signal name.

RESCLK

Clock (TXCLK) input. Data transfer/synchronization clock for synchronizing transfers on RESDATA. When operating in PHY mode, the RESCLK is sourced from the UTOPIA interface.

PRINCIPLES OF OPERATION

RESDATA(bits 7–0)

Data (TXDATA) input**.** RESDATA7 is the MSB.

RESPAR

Parity (TXPAR) input. RESPAR is the odd-parity bit over RESDATA.

RESSOC

Start of cell (TXSOC) input. This active-high signal is received by the TNETA1570 when RESDATA contains the first valid byte of the cell.

$\overline{\text{RESEN}}$

Enable ($\overline{\text{TXENB}}$) input. This active-low signal is received by the TNETA1570 when RESDATA contains a valid byte**.**

RESCLAV

Full/cell available ($\overline{\text{TXFULL}}$/TXCLAV) output. An indication that a transfer of a complete cell can be accepted.

$\overline{\text{RESPERR}}$

Parity error output (not required by the UTOPIA specification). This active-low signal asserted by the reassembly unit to indicate that parity error was present at the previous rising RESCLK edge.

segmentation-unit interface, ATM mode

The segmentation-unit interface on the TNETA1570 outputs TXDATA on the UTOPIA interface in ATM mode. This cell interface works on the low-to-high transition of SEGCLK to sample and generate signals. All signals are active high unless denoted via a bar over the signal name.

SEGCLK

Clock (TXCLK) input. Data transfer/synchronization clock for synchronizing transfers on SEGDATA. When operating in ATM mode, the SEGCLK is sourced from the ATMCLK.

SEGDATA (bits 7–0)

Data (TXDATA) output. Byte-wide true data sourced by the TNETA1570 SEGDATA7 is the MSB.

SEGPAR

Parity (TXPRTY) output. SEGPAR is the odd-parity bit over SEGDATA.

SEGSOC

Start of cell (TXSOC) output. Active-high signal sourced by the TNETA1570 when SEGDATA contains the first valid byte of the cell.

$\overline{\text{SEGEN}}$

Full/cell available ($\overline{\text{TXFULL}}$/RXCLAV) input. (Functional description does not match pin name for ATM mode) An indication that at least one byte of SEGDATA is accepted.

SEGCLAV

Enable ($\overline{\text{TXENB}}$) output. (Functional description does not match pin name for ATM mode). This active-low signal is sent by the TNETA1570 when SEGDATA contains a valid byte.

reassembly-unit interface, ATM mode

The reassembly-unit interface on the TNETA1570 inputs RXDATA from the UTOPIA interface in ATM mode. This receive cell interface works on the low-to-high transition of RESCLK to sample and generate signals. All signals are active high, unless denoted via a bar over the signal name.

PRINCIPLES OF OPERATION

RESCLK

Clock (RXCLK) input. Data transfer/synchronization clock for synchronizing transfers on RESDATA. When operating in ATM mode, the ATMCLK is used as the source for the RESCLK.

RESDATA

Data (RXDATA) input**.** RESDATA7 is the MSB.

RESPAR

Parity (RXPAR) input. RESPAR is the odd-parity bit over RESDATA.

RESSOC

Start of cell (RXSOC) input. This active-high signal is received by the TNETA1570 when RESDATA contains the first valid byte of the cell.

$\overline{\text{RESEN}}$

Empty/cell available ($\overline{\text{RXEMPTY}}$/RXCLAV) input. An indication that there is no valid data to be transferred in the current cycle. (Functional description does not match pin name for ATM mode).

RESCLAV

Enable ($\overline{\text{RXENB}}$) output. This active-low signal is sent by the TNETA1570 when a valid byte of RESDATA is sent during the next clock cycle. (Functional description does not match pin name for ATM mode).

$\overline{\text{RESPERR}}$

Parity error output. (Not required by the UTOPIA specification) Active-low signal asserted by the reassembly unit to indicate parity error was present at the previous rising RESCLK edge.

internal FIFOs

The TNETA1570 has a transmit FIFO for internal interfacing between the segmentation engine and the cell-transmit block. The TX FIFO size is 32 (double words) x 64 (bits), which hold 4 cells. Likewise, the TNETA1570 has a receive FIFO for internal interfacing between the reassembly engine and the cell-receive block. The RX FIFO size is 64 (double words) x 64 (bits), which hold 8 cells.

host interface

The TNETA1570 incorporates a PCI local-bus specification revision 2.0 (April 30, 1993) compliant-host-interface. The following sections describe the features and operation of the PCI-host interface.

host-interface overview

The TNETA1570 operates as a PCI-slave device for configuration cycles, accesses to internal registers, and accesses to the onboard control memory. It also acts as a PCI-master device for accessing data structures that are contained in host memory.

As a master, the TNETA1570 incorporates the following features:

- Directly supports the memory-read, memory-write, configuration-read and configuration-write PCI commands and aliases the memory-read multiple, memory-read line, and memory write and invalidate to the basic memory commands.
- Supports single data-cycle transfers and disconnects with retry after the first data cycle
- Responds to accesses as a 32-bit agent with medium $\overline{\text{PDEVSEL}}$ timing (single-wait state)
- Utilizes a 1-M block of addresses, which is mapped into the host-memory space using a single base-address register
- Does not support resource locking

PRINCIPLES OF OPERATION

host-interface overview (continued)

As a master, the TNETA1570 incorporates the following features:

- Utilizes the memory-read and memory-write PCI-bus commands
- Initiates transactions as a 32-bit or 64-bit agent (for data only)
- Initiates transactions as a 32-bit agent (for control structures)
- Performs multiple data-cycle transfers when possible up to a maximum of 12 data cycles
- Allows the user to specify the maximum number of times a transaction is retried before flagging a system-error condition

signal descriptions

The following is a summary of the PCI-bus signals with a brief description of each signal. An overbar is used to denote an active-low signal. The pin names are found by adding the prefix PCI_ to the signal names. The definitions of the electrical types used in the signal descriptions are as follows:

TYPE	DESCRIPTION
Standard input	Input is a standard input-only signal.
Standard output	Output is a standard totem-pole active driver.
3 state	Three-state is a bidirectional input/output signal.
Sustained 3 state	Sustained 3-state is an active-low 3-state signal owned and driven by only one agent at a given time.
Open drain	Open drain allows multiple devices to share signals as a wired-OR.

In addition to the standard PCI-interface signals, two additional sideband signals are also defined. These signals are SB(bit 0) and SB(bit 1). They give the size of the transfer when the TNETA1570 is the bus master. These signals are defined as follows:

SB (bit 1)	SB (bit 0)	DESCRIPTION
0	0	4-byte transfer
0	1	16-byte transfer except for receive completion ring
1	0	Payload transfer
1	1	Transfer to receive completion ring 20-byte transfers

Sideband signals are synchronous to the address phase of bus-mastering operations by the TNETA1570. Sideband signals are not changed upon a bus retry after disconnect. The sideband signals are always active (i.e., they are always driven by the TNETA1570). They comply with point-to-point PCI-bus timing.

configuration space definition

As described in the PCI-local bus specification revision 2.0, the TNETA1570 provides a 64-byte configuration space that can be accessed by system software for configuration, initialization, and error handling. The TNETA1570 responds to type 0 configuration accesses. The following map describes the 64-byte configuration space.

PRINCIPLES OF OPERATION

configuration space definition (continued)

ADDRESS	BYTE 3	BYTE 2	BYTE 1	BYTE 0	READ/WRITE
0x00	Device ID		Vendor ID		R
0x04	Status		Command		R/W
0x08	Class code			Revision ID	R
0x0C	BIST	Header type	Latency timer	Cache-line size†	R/W
0x10	Base address 0				R/W
0x14	Base address 1†				R/W
0x18	Base address 2†				R/W
0x1C	Base address 3†				R/W
0x20	Base address 4†				R/W
0x24	Base address 5†				R/W
0x28	Reserved (returns 0 when read)				—
0x2C	Reserved (returns 0 when read)				—
0x30	Expansion-ROM base address				R/W
0x34	Reserved (returns 0 when read)				—
0x38	Reserved (returns 0 when read)				—
0x3C	Maximum latency	Minimum grant	Interrupt pin	Interrupt line	R/W
0x40	Reserved	Reserved	Reserved	Reserved	R/W
0x44 – 0xFF	Reserved (returns 0 when read)				R

† Optional registers not implemented that return 0 when read

device-identification registers

The system software identifies installed PCI peripherals using the vendor-ID, device-ID, revision-ID, and class-code registers. The registers used for device identification are described below.

REGISTER	DESCRIPTION	HARDWIRED VALUE
Vendor ID	Uniquely identifies the manufacturer of the device. Vendor ID is specified by the PCI-special-interest group to ensure uniqueness.	104Ch
Device ID	Identifies a specific device from the manufacturer. The device ID is specified by the manufacturer.	A001h
Revision ID	Identifies a revision of the specific device. The revision ID is specified by the manufacturer.	00h
Class code	Identifies the generic function of the device	02h
Sub-class code		80h
Header type	Identifies the layout of bytes 10h through 3F and whether the device is single or multifunction	00h

PRINCIPLES OF OPERATION

device command and status registers

The system software accesses the status and command registers for error recovery, diagnostic, and control purposes. The command register is provided to enable coarse control over a device's ability to generate and respond to PCI cycles. The command register bits are defined in the following table.

BIT	NAME	I/O FUNCTION	RESET VALUE
0	I/O Space	This bit enables the device to respond to I/O accesses within its address space. The TNETA1570 does not support I/O mapped accesses; therefore, this bit is hardwired to a 0.	0
1	Memory space	This bit enables the device to respond to memory accesses within its address space. If this bit is cleared, the TNETA1570 does not respond to memory-mapped accesses.	0
2	Bus master	This bit enables the device to act as a PCI-bus master. If this bit is cleared, the device does not act as a master on the PCI bus.	0
3	Special cycles	This bit controls the device's response to special-cycle commands. If this bit Is cleared, the device ignores all special-cycle commands. If this bit is set to 1, the device can monitor special-cycle commands.	0
4	Memory write and invalidate enable	This bit enables the device to use the memory-write and invalidate command. The TNETA1570 does not support the memory-write and invalidate command as a master; therefore, this bit is hardwired to a 0.	0
5	VGA-palette snoop	This bit is not applicable for the TNETA1570 and is hardwired to a 0.	0
6	Parity-error response	This bit controls whether the device responds to detected parity errors. If this bit is set, the TNETA1570 responds normally to parity errors. If this bit is cleared, the TNETA1570 ignores detected parity errors.	0
7	Wait-cycle control	This bit indicates whether the device performs address stepping. The TNETA1570 does not require address stepping; therefore, this bit is hardwired to 0.	0
8	$\overline{\text{SERR}}$ enable	This bit is an enable for the output driver on $\overline{\text{SERR}}$. If this bit is cleared and a system-error condition is set inside the TNETA1570, the error signal does not appear on the external $\overline{\text{SERR}}$ signal.	0
9	Fast back-to-back enable	This bit controls whether the device is allowed to perform back-to-back writes to different targets. The TNETA1570 does not perform fast back-to-back transactions; therefore, this bit is hardwired to a 0.	0
10–15	Reserved	Always return 0	All 0s

PRINCIPLES OF OPERATION

device command and status registers (continued)

The status register allows the host to monitor the condition of the peripheral device. The individual bits within the status register are defined in the following table.

BIT	NAME	FUNCTION	RESET VALUE
0–6	Reserved	Always return 0	0
7	Fast back-to-back enable	This bit indicates that the device is capable of performing fast back-to-back transactions. The TNETA1570 does not support fast back-to-back transactions; therefore, this bit is hardwired to 0.	0
8	Data parity reported	This bit is set by the PCI-master unit in the TNETA1570 when all of the following conditions are met: ● The TNETA1570 asserted $\overline{\text{PERR}}$ or observed $\overline{\text{PERR}}$ asserted. ● The TNETA1570 PCI-master unit is the bus master during the observed $\overline{\text{PERR}}$. ● The parity-error-response bit (command-register bit 6) is set.	0
9,10	DEVSEL timing	This bit indicates the decode-response-time capability of the device. The TNETA1570 decode logic supports medium $\overline{\text{DEVSEL}}$ timing; therefore, these bits are hardwired to 01.	0
11	Signaled target abort	This bit is set by the PCI-slave unit in the TNETA1570 to indicate that it has initiated a target abort.	0
12	Received-target abort	This bit is set by the PCI-master unit in the TNETA1570 to indicate that it has received a target abort when acting as a bus master.	0
13	Initiated-master abort	This bit is set by the PCI-master unit in the TNETA1570 to indicate that It has initiated a master abort.	0
14	Signaled system error	This bit is set by the TNETA1570 to indicate that it signaled a system error on $\overline{\text{SERR}}$.	0
15	Detected parity error	This bit is set by the TNETA1570 to indicate that it detected a parity error, which was not necessarily reported on $\overline{\text{PPERR}}$ if parity reporting is disabled.	0

cache-line-size register

This register enables the host to inform the device of the cache-line size for the usage of the write and invalidate PCI command. The TNETA1570 does not support the write and Invalidate command as a master so this register is not implemented and returns all zeroes when read.

latency-timer register

The latency-timer register is provided so that the host can restrict the continued usage of the PCI bus by a master involved in a multiple data-cycle transaction after its $\overline{\text{PGNT}}$ has been removed. The host is required to write a value into this register indicating the maximum number of PCI cycles that the master can hold the bus (beginning from the assertion of $\overline{\text{PFRAME}}$). If $\overline{\text{PGNT}}$ is never removed during the transaction, the value in the latency-timer value is not used. Since the TNETA1570 supports transactions with multiple data cycles, the latency-timer register is implemented. The latency-timer register is initialized with all zeroes at reset. This register is not cleared on software reset.

built-in self-test (BIST) register

BIST is not supported by the TNETA1570. Reading this register returns all zeroes.

PRINCIPLES OF OPERATION

base-address registers

The base-address registers enable the host to map the device's address space into the host-memory or I/O-address space. The TNETA1570 only uses the base-address 0 register from the configuration block. This register is not cleared on software reset. A description of the individual bits in the base-address 0 register follows:

BIT	NAME	FUNCTION	RESET VALUE
0	Memory-space indicator	This bit indicates whether the base address maps into the host's memory or I/O space. This bit is hardwired to 0 in the TNETA1570 indicating that this base address is valid only for memory accesses. Due to the large amount amount of space required by the device, mapping into the I/O space is not supported.	0
1–2	Type	These bits indicate the size of the base address and how it can be mapped into the host memory. These bits are hardwired to 00 in the TNETA1570 indicating that a 32-bit base-address register is used, which can be located anywhere in memory.	00
3	Prefetchable	This bit is hardwired to 0 in the TNETA1570.	
4–19	Internally used address bits	These bits are hardwired to all zeroes to define the size of the address space needed by the TNETA1570, which contains 1-M locations. The system-setup software can determine the address space needed by a PCI device by writing all ones to the base-address register, reading the written value back, and noting which bits were cleared.	All 0s
20–31	Available address bits	These bits are writable by the host to allow initialization of the base address at startup.	All 0s

latency, grant, and interrupt registers

REGISTER	DESCRIPTION	HARDWIRED VALUE
MAX_LAT	This specifies how often the device needs to gain access to the PCI bus in 0.25-μs units. The typical value of .5 μs (02h) is hardwired for the TNETA1570.	02h
MIN_GNT	This specifies the length of the burst period the device needs in 0.25-μs units. The typical value of 0.25-μs (01h) is hardwired for the TNETA1570.	01h
Interrupt line	The value in this 8-bit register is written by the host and indicates to which input of the system-interrupt controller the TNETA1570 interrupt signal is connected. This register is not cleared on software reset.	00h
Interrupt signal	This register indicates which interrupt signal the device uses. This register is hardwired to a 01h in the TNETA1570 to indicate that interrupt A is used.	01h

error and interrupt signals

parity-error ($\overline{\text{PPERR}}$) protocol

If the TNETA1570 is mastering the bus, the data-parity-reported bit (bit 15) in the PCI configuration-space status register is set under either of the following conditions:

- If it detects a parity error during the data phase of a read transaction
- If it detects that $\overline{\text{PPERR}}$ has been asserted by the target during the data phase of a write transaction

The data-parity-detected bit (bit 8) in the PCI-configuration-space status register is set under any of the following conditions:

- If is is acting as the PCI-bus master and it detects a data-parity error during a read transaction
- If it is acting as a PCI-bus target and it detects a data-parity error during a write transaction
- If it detects an address-parity error

PRINCIPLES OF OPERATION

parity error ($\overline{PERR}$) protocol (continued)

The TNETA1570 asserts $\overline{PERR}$ if the parity-error-response bit (bit 6) in the PCI-configuration-space command register is set and the data-parity-detected bit is set. The assertion of $\overline{PERR}$ remains valid until the second clock after the cycle in which the error occurred. If a parity error is detected during a transfer involving the TNETA1570, the transaction is allowed to complete unless the TNETA1570 is the master and a target disconnect is detected (i.e. the TNETA1570 will not master abort due to a parity error).

system error ($\overline{SERR}$) protocol

The TNETA1570 sets an internal $\overline{SERR}$ flag under any of the following conditions:

- If an address-parity error is detected on the PCI bus (even if the TDC1570 is not the target of the transaction) and the parity-error-response bit is set in the PCI-configuration-space command register.
- If the TNETA1570 detects $\overline{PERR}$ asserted while mastering the bus
- If the PCI-master-retry counter expires
- If the TNETA1570 receives a target abort (disconnect without retry) while mastering the bus
- If the TNETA1570 self selects

The TNETA1570 asserts $\overline{SERR}$ if the SERR enable bit (bit 8) in the PCI-configuration-space command register is set and the internal system-error flag is set. The TNETA1570 halts and waits for software or hardware reset after $\overline{SERR}$ is asserted.

The TNETA1570 sets the signaled-system error bit (bit 14) in the PCI-configuration-space status register when $\overline{SERR}$ is asserted.

host-interrupt ($\overline{INTA}$) protocol

The TDC1570 generates an interrupt to the host by asserting $\overline{INTA}$ if any of the bits in the internal status register are set and the corresponding status-mask-register bit is also set.

PCI-to-UTOPIA data flow for segmentation unit

little-endian configuration

There are two scenarios for TNETA1570 cell-payload PCI reads from the host memory when the TNETA1570 is configured as a little-endian device. These are 32-bit PCI cell-payload reads and 64-bit PCI cell-payload reads.

32-bit PCI cell-payload reads

The top-down order of the following chart represents the time order of 32-bit word reads by the TNETA1570.

PCI ADDRESS	PCI ADDRESS BITS			
	31 24	23 16	15 8	7 0
0×10	3	2	1	0
0×14	7	6	5	4
0×18	11	10	9	8

The cell data above is transmitted from the segmentation unit's UTOPIA bus in the byte order 0, 1, 2, 3, 4, 5, . . .

PRINCIPLES OF OPERATION

64-bit PCI cell-payload reads

The top-down order of the following chart represents the time order of 64-bit word reads by the TNETA1570.

PCI ADDRESS	PCI ADDRESS BITS							
	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
0×10	7	6	6	4	3	2	1	0
0×18	15	14	14	12	11	10	9	8
0×20	23	22	22	20	19	18	17	16

The cell data above is transmitted from the segmentation unit's UTOPIA bus in the byte order 0, 2, 3, 4, 5, 6, 7, 8, 9 . . .

big-endian configuration

There are two scenarios for TNETA1570 cell-payload PCI reads from the host memory when the TNETA1570 is configured as a big-endian device. These are 32-bit PCI cell-payload reads and 64-bit PCI cell-payload reads.

32-bit PCI cell-payload reads

The top-down order of the following chart represents the time order of 32-bit word reads by the TNETA1570.

PCI ADDRESS	PCI ADDRESS BITS			
	31 24	23 16	15 8	7 0
0×10	0	1	2	3
0×14	4	5	6	7
0×18	8	9	10	11

The cell data above is transmitted from the segmentation unit's UTOPIA bus in the byte order 0, 1, 2, 3, 4, 5, . . .

64-bit PCI cell-payload reads

The following table shows the PCI bus byte/address ordering when cell-payload bytes are input to the reassembly UTOPIA port in order 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 . . .

The top-down order of the following chart represents the time order of 64-bit word reads by the TNETA1570.

PCI ADDRESS	PCI ADDRESS BITS							
	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
0×10	4	5	6	7	0	1	2	3
0×18	12	13	14	15	8	9	10	11
0×20	20	21	22	23	16	17	18	19

The cell data above is transmitted from the segmentation unit's UTOPIA bus in the byte order 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 . . .

PRINCIPLES OF OPERATION

UTOPIA-to-PCI data flow for reassembly unit

little-endian configuration (PCI compliant)

There are two scenarios for TNETA1570 cell-payload PCI writes to host memory when the TNETA1570 is configured as a little-endian device. These are 32-bit PCI cell-payload writes and 64-bit PCI cell-payload writes.

32-bit PCI cell-payload writes

The following table shows the PCI-bus byte/address ordering when cell-payload bytes are input to the reassembly UTOPIA port in the order 0, 1, 2, 3, 4, 5, . . .

The top-down order of the following chart represents the time order of 32-bit word writes by the TNETA1570.

PCI ADDRESS	PCI ADDRESS BITS			
	31 24	23 16	15 8	7 0
0×10	3	2	1	0
0×14	7	6	5	4
0×18	11	10	9	8

64-bit PCI cell-payload writes

The following table shows the PCI-bus byte/address ordering when cell-payload bytes are input to the reassembly UTOPIA port in the order 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, . . .

The top-down order of the following chart represents the time order of 64-bit word writes by the TNETA1570.

PCI ADDRESS	PCI ADDRESS BITS							
	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
0×10	7	6	5	4	3	2	1	0
0×18	15	14	13	12	11	10	9	8
0×20	23	22	21	20	19	18	17	16

big-endian configuration

There are two scenarios are for TNETA1570 cell-payload PCI writes to host memory when the TNETA1570 is configured as a big-endian device. These are 32-bit PCI cell-payload writes and 64-bit PCI cell-payload writes.

32-bit PCI cell-payload writes

The following table shows the PCI-bus byte/address ordering when cell-payload bytes are input to the reassembly UTOPIA port in the order 0, 1, 2, 3, 4, 5, . . .

The top-down order of the following chart represents the time order of 32-bit word writes by the TNETA1570.

PCI ADDRESS	PCI ADDRESS BITS			
	31 24	23 16	15 8	7 0
0×10	0	1	2	3
0×14	4	5	6	7
0×18	8	9	10	11

PRINCIPLES OF OPERATION

64-bit PCI cell-payload writes

The following table shows the PCI-bus byte/address ordering when cell-payload bytes are input to the reassembly UTOPIA port in the order 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, . . .

The top-down order of the following chart represents the time order of 64-bit word writes by the TNETA1570.

PCI ADDRESS	PCI ADDRESS BITS							
	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
0×10	4	5	6	7	0	1	2	3
0×18	12	13	14	15	8	9	10	11
0×20	20	21	22	23	16	17	18	19

TNETA1570 accesses to control structures in host memory

The PCI accesses in the big-endian configuration section apply to cell-payload transfers when segmenting or reassembling packets. All other host-memory accesses by the TNETA1570 are to entities which are defined as control structures. These control structures are the transmit-buffer descriptors, packet-segmentation rings, transmit-completion rings, receive-buffer descriptors, receive free-buffer rings/FIFO**,** and receive-completion rings.

Accesses to control structures by the TNETA1570 are 32 bit only. Big- or little-endian configuration does not affect control structures since the information contained in each 32-bit word of a control structure is bit defined in this document.

access to control memory via the PCI interface

The control memory is accessed via the PCI interface in full 32-bit words. Byte enable is not supported. All four bytes in a 32-bit-word in the control memory are read or written to in a single instruction. If none of the byte enables are asserted for a write instruction, none of the data in the 32-bit word is altered. If any byte enable is asserted, the entire 32-bit word is overwritten.

internal registers and counters

The TNETA1570 has several internal registers for storing configuration and status information, as well as keeping track of certain statistics. The following internal registers have been previously defined for the TNETA1570 device:

DESCRIPTION		
Bits 31–16	Bits 15–0	Offset Address From PCI-Base Address for TNETA1570 Reserved Area (= 3200 hex)
Configuration Register		0000h
Status Register		0004h
Interrupt-Mask Register		0008h
Reassembly-Global-Timer Register	RAT Cycle-Counter Register	000C
RX Unknown Register		0010h
TX Completion Ring Without Interrupt-Size Register	TX Completion Ring With Interrupt-Size Register	0014h
RX Completion Ring Without Interrupt-Size Register	RX Completion Ring With Interrupt-Size Register	0018h

PRINCIPLES OF OPERATION

internal registers and counters (continued)

DESCRIPTION			
TX Packet-Segmentation Ring-Size Register	FIFO Free-Buffer-Size Register		001Ch
HEC-Error Counter	AAL5 CPCS PDU Discard Register		0020h
Unknown-Protocols Counter			0024h
ATM-Cells-Received Counter			0028h
ATM-Cells-Transmitted Counter			002ch
VCI-Mask Register	TX FIFO Maximum Occupancy Value	RX FIFO Maximum Occupancy Value	0030h
Reserved	Scheduler-Table-Size Register		0034h
Software Reset			0038h
TX Completion Ring Without Interrupt-Pointer Address			0200h
TX Completion Ring With Interrupt-Pointer Address			0204h
RX Completion Ring Without Interrupt-Pointer Address			0208h
RX Completion Ring With Interrupt-Pointer Address			020Ch

configuration register

This register is used to program the operation of the device. This register is accessed as a 32-bit word, although not all 32 bits are used. The register is read/write and is cleared when a hardware or software reset occurs.

RESERVED	NNI/ UNI	MAX_RETRY _COUNT	LOOP_ BACK	TX_HEC _ERR	SMALL_MAP	EN_TX	EN_RX	ENDN_SEL	BP_SEL	EN_RAT
bits 31–14	bit 13	bits 12–9	bit 8	bit 7	bit 6	bit 5	bit 4	bits 3–2	bit 1	bit 0

NNI/UNI (bit 13) – select NNI or UNI ATM-cell-header interpretation

When this bit is not set (=0) the ATM cell header is interpreted as NNI ATM cell header, (i.e., the first four bits in the header are used as part of the VPI value). When this bit is set (=1), the cell header is interpreted as UNI cell header, (i.e., the first four bits are considered GFC bits and masked before the VPI field is used as an address to the RX VPI/VCI DMA pointer table). The GFC bits are not used in the TNETA1570, but they are passed on to the host as part of the ATM header in the RX data buffers. The NNI/UNI bit is cleared (=0) after reset.

MAX_RETRY_COUNT (bits 12–9)

These bits specify the maximum number of retries that the TNETA1570 master attempts for any transaction before asserting $\overline{\text{PSERR}}$. If this field is set to 0, the master retries forever. Reset value is zero.

LOOP_BACK (bit 8)

If the LOOP_BACK bit is set, the segmented data is looped back into the reassembly engine**.** External data available for reassembly are ignored. No loopback is performed if the bit is not set.

TX_HEC_ERR (bit 7)

The HEC field of a segmented package is inverted if the TX_HE_ ER bit is set high. The HEC field is transmitted correctly if the bit is not set.

PRINCIPLES OF OPERATION

SMALL_MAP (bit 6)

If the SMALL_MAP bit is set in the configuration register, address line CM_AD(14) is ORed in the TNETA1570 with address line CM_AD (13) and CM_AD (12) The SRAM needed is 16K x 32. If this option is chosen, a maximum of 512 TX DMA states and 512 RX DMA states are used. Alternative control-memory map is used to support small SRAM configuration.

EN_TX (bit 5)

The EN_TX bit allows the host to disable packet segmentation and any payload-data transfer from the host to the link. It is set high to enable normal transmit processing and set to zero to disable such processing. It is set to zero on reset, disabling transmit operation until various configuration registers, the scheduler table, and the DMA blocks are configured by the host. The transfer of the new cells from PCI bus to TNETA1570 is inhibited when the EN_TX bit is disabled; however, cells already in the output buffer are forwarded to the PHY layer. This bit is set to zero by the TNETA1570 upon TX_FRZ condition and set high by the host allowing resumption of normal operation.

EN_RX (bit 4)

The EN_RX bit allows the host to disable packet reassembly. All cells from the PHY layer are dropped when the EN_RX bit is zero. The EN RX bit is set high to enable normal processing. It is set to zero on reset, disabling receive operation until various configuration registers and the DMA blocks are reconfigured by the host. The transfer of new cells from the ATM link to the receive buffer is inhibited when the EN_RX bit is disabled. This bit is set to zero by the TNETA1570 upon RX_FRZ condition and set high by the host allowing resumption of normal operation.

ENDN_EL (bits 3, 2)

The endian-select bits allows the host to configure the endian mode for TNETA1570 as shown below. The TNETA1570 is configured for little-endian mode immediately after reset.

00 = little endian

01 = big endian

1X = reserved

BP_SEL (bit 1)

If interrupt is enabled, this bit selects the criteria used for the transmit-completion ring. When the buffer-/packet-interrupt-select bit is set to high, an interrupt is generated by the TNETA1570 upon posting to the transmit-completion ring on a per-packet basis. An interrupt is generated on a per-buffer basis when the bit is set to low.

EN_RAT (bit 0)

When this bit is set high (=1), the reassembly-aging timer is enabled. The global-timer register and the maximum RX DMA-state table-size register must be initialized for timer operation. When EN_RAT is set low (= 0), no reassembly aging on any receive DMA channel occurs.

status register

This register contains information on the current operating state of the device. When an interrupt occurs, the host can read this register to determine the cause of the interrupt. The lower ten bits (9–0) are configured as clear-on-read, (i.e., the contents of the lower ten bits of this register are cleared when the register is read). The higher 22 bits (bits 31–0) remain unchanged on read. This register is read only and is accessed as a 32-bit word, although not all 32 bits may be used. This register is cleared upon a hardware or software reset.

PRINCIPLES OF OPERATION

PCI_ STATUS	RESERVED	PCI_MODE	RX_ FRZ	TX_FRZ	CP_RX	RX_IRR	HEC_R	UP_R	APD_R	ACR_R	ACT_R	CP_TX
bits 31–16	bits 15–11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bits 4	bit 3	bit 2	bit 1	bit 0

PCI_STATUS (bits 31–16)

These sixteen bits of the status register contain the PCI-interface status bits. They are located here to allow for a single 32-bit word access to acquire the complete status of the TNETA1570.

PCI_MODE (bit 10)

This bit is used to indicate either 32-bit or 64-bit PCI interface. When this bit is high, the 64-bit PCI operation is selected at reset.

RX_FRZ (receive freeze) (bit 9)

The TNETA1570 sets the RX_FRZ bit when a receive-completion-ring overflow is detected. It is cleared on read.

TX_FRZ (transmit freeze) (bit 8)

The TNETA1570 sets the TX_FRZ bit when a transmit-completion-ring overflow is detected. It is cleared on read.

CP_RX (receive completion ring) (bit 7)

This bit is set when the reassembly of a packet is completed and the receive completion ring with interrupt is updated.

RX_IRR (receive unknown register set) (bit 6)

This bit of the status register is set when a value is written to the RX-unknown register.

HEC_R (bit 5)

This bit is set when the HEC-error counter rolls over.

UP_R (unknown protocols rollover) (bit 4)

This bit is set when the unknown-protocols counter rolls over.

APD_R (AAL5 CPCS PDU discard counter rollover) (bit 3)

This bit is set when the AAL5 CPCS PDU discard counter rolls over.

ACR_R (ATM cells received counter rollover) (bit 2)

This bit **is** set when the ATM-cells-received counter rolls over.

ACT_R (ATM cells transmitted counter rollover) (bit 1)

This bit is set when the ATM-cells-transmitted counter rolls over.

CP_TX (transmit completion ring) (bit 0)

This bit is set when the segmentation of a buffer or packet is completed and the transmit-completion ring with interrupt is updated.

PRINCIPLES OF OPERATION

interrupt-mask register

PCI Mask Bits (bits 31–16)	Reserved (bits 15–10)	Mask Bits (bits 9–0)

This register is used to mask interrupts caused by the conditions given in the status register. This register has a bit-to-bit correspondence with the status register. This register is read/write and is accessed as a 32-bit word, although not all 32 bits may be used. This register is cleared upon a hardware or software reset. When a bit in the interrupt-mask register is set, an interrupt is generated if the corresponding bit in the status register is also set.

reassembly-global-timer register

Reassembly-Global Timer (bits 15–0)

This register contains the value that is loaded into the reassembly-aging timer. When the reassembly-aging counter expires, a reassembly time-out count in one of the RX DMA state table entries is incremented, if that channel is currently receiving a packet. The entries in the RX DMA state table are serviced on a round-robin basis. This register is a 16-bit register and is read/write. This register is cleared upon a hardware or software reset.

RAT-cycle-count register

Reserved (bit 15)	RAT_Cycle-Count_Register (bits 14–0)

This register is used in the reassembly-aging process. The value must be equal to or greater than the aging opportunities in one trip through the VPI/VCI DMA pointer table. The minimum value entered in this field is 4096 + [the sum of all (valid VCI range) in the RX VPI/VCI DMA pointer table].

Example(decimal values):

Entry 0 has the valid VCI range = 255.

Entry 8 has the valid VCI range = 0.

Entry 1234 has the valid VCI range = 1234.

Entry 4095 has the valid VCI range = 5432.

All other entries are disabled.

The minimum value is 4096 + 255 + 0 + 1234 + 5432 = 11017. The maximum value is limited by the number of bits available (15) for this field. This is a 16-bit register with read/write capability and is cleared upon a hardware or software reset.

RX unknown register

Reserved (bits 31–29)	Control (bit 28)	VPI (bits 27–16)	VCI (bits 15–10)

This register contains the ATM header, including the VPI/VCI value of a received cell under the following conditions:

- The cell VPI address entry in the RX VPI/VCI DMA pointer table is not enabled.
- The cell VCI address does not fall within the range of active values in the RX VPI/VCI DMA pointer table.
- The cell's on bit and filter bit in the RX DMA state table is not set.
- The cell is an idle cell, i.e., a cell with the ATM header set to a binary value of xxxx 0000 0000 0000 0000 0000 0000 xxx1 (also known as an invalid cell).

PRINCIPLES OF OPERATION

RX unknown register (continued)

This register holds the value first written into it until it has been read by the host. This is a 32-bit register, read only, and is cleared when a hardware or software reset occurs. The register is also cleared when it is read by the host. This register also contains a control bit that is set when a value is written into the register by the TNETA1570 and cleared when the register is read by the host. When a value is written to this register, a bit in the status register is set and an interrupt may or may not be generated, depending upon the state of the interrupt-mask bit in the interrupt-mask register.

TX completion ring without interrupt (WOI)-size register

Reserved (bits 15–13)	TX Completion Ring WOI Size (bits 12–0)

This register contains the size of the TX completion ring without interrupt. The maximum size of this ring is 8192 entries. The size is restricted to a power of 2, I.e., 2, 4, 8, 1 6, 32, 64, 1 28, etc. This field is coded (ring size –1) to account for entry 0; therefore, for a ring with 16 entries, the value of 00 0000 1111 is entered in this field. This is a 16-bit register, read/write, and is cleared when a hardware or software reset occurs.

The value in the register is compared to an internal pointer counter that is incremented when a completion-ring entry is processed. When the value of the pointer counter equals the value in the register, the pointer value is reset and the next pointer value is zero.

TX completion ring with interrupt (WI)-size register

Reserved (bits 15–13)	TX Completion Ring WI Size (bits 12–0)

This register contains the size of the TX completion ring with interrupt. The maximum size of this ring is 8192 entries. The size is restricted to a power of 2, i.e., 2, 4, 8, 16, 32, 64, 128, etc. This field is coded (ring size –1) to account for entry 0; therefore, for a ring with 16 entries, the value of 00 0000 1111 is entered in this field. This is a 16-bit register, read/write, and is cleared when a hardware or software reset occurs.

The value in the registers is compared to an internal pointer counter that is incremented when a completion-ring entry is processed. When the value of the pointer counter equals the value in the register, the pointer value is reset and the next pointer value is zero.

RX completion ring without interrupt (WOI)-size register

Reserved (bits 15–10)	RX Completion Ring WOI Size (bits 9–0)

This register contains the size of the RX completion ring without interrupt. The maximum size of this ring is 1024 entries. The size is restricted to be a power of 2, i.e., 2, 4, 8, 16, 32, 64, 128, etc. This field is coded (ring size –1) to account for entry 0**;** therefore, for a ring with 16 entries, the value of 00 0000 1111 is entered in this field. This is a 16-bit register, read/write, and is cleared when a hardware or software reset occurs.

The value in the registers is compared to an internal pointer counter that gets incremented when a completion-ring entry is processed. When the value of the pointer counter equals the value in the register, the pointer value is reset and the next pointer value is zero.

RX completion ring with interrupt (WI)-size register

Reserved (bits 15–10)	RX Completion Ring WI Size (bits 9–0)

This register contains the size of the RX completion ring with interrupt. The maximum size of this ring is 1024 entries. The size is restricted to a power of 2, i.e., 2, 4, 8, 16, 32, 64, etc. This field is coded (ring size –1) to account for entry 0; therefore, for a ring with 16 entries, the value of 00 0000 1111 is entered in this field. This is a 16-bit register, read/write, and is cleared when a hardware or software reset occurs.

PRINCIPLES OF OPERATION

RX completion ring with interrupt (WI)-size register (continued)

The value in the registers is compared to an internal pointer counter that gets incremented when a completion-ring entry is processed. When the value of the pointer counter equals the value in the register, the pointer value is reset and the next pointer value is zero.

TX packet-segmentation ring-size register

Reserved (bits 15–8)	TX Packet-Segmentation-Ring Size (bits 7–0)

This register contains the size of the TX packet-segmentation ring. The maximum size of this ring is 256 entries. The size is restricted to a power of 2, i.e., 2, 4, 8, 16, 32, 64, 128, etc. This field is coded (ring size –1) to account for entry 0; therefore, for a ring with 16 entries, the value of 0000 1111 is entered in this field. This is a 16-bit register, read/write, and is cleared when a hardware or software reset occurs.

FIFO free-buffer-size register

FIFO Free-Buffer Size (bits 15–0)

This register contains the size of the buffers used with the FIFO free-buffer structure. The size of the free buffer is given in the number of ATM cells that the buffer can hold. If an AAL5 packet is to be placed in a single buffer, the buffer must be able to hold 1366 cells. To program a buffer size of 64K byte, the value in the register should be 64K / 48 rounded up = 1366(dec), (65536/48 = 1 365.3 = > 1366). This is a 16-bit register, read/write, and is cleared when a hardware or software reset occurs.

VCI-mask register

VCI Mask (bits 15–0)

The 16-bit VCI mask field is uniformly imposed on all received cells, except OAM cells. If the bit value in the mask=1, the VCI bit in the incoming ATM cell is replaced by 0 before it is processed. If the bit value in the mask=0, the VCI bit is not affected.

The VCI mask works as follows:

- Cell arrives
- The cell is checked if it is a OAM cell; if so, it is processed as an OAM cell.
- VCI has mask imposed on it (invert and perform logical AND function with the bits in the register).This results in a new 16-bit value.
- This new 16-bit value is compared to the valid VCI range value in the RX VPI/VCI pointer table to check if it is out of range.
- If not out of range, the new 16-bit value is used to get to the appropriate RX DMA state-table entry.

The RAT timer is not affected. It uses the value programmed in the VPI/VCI DMA pointer table.

The received VPI/VCI appears in the RX completion ring and the RX data-buffer descriptor.

FIFO maximum occupancy value

Reserved (bits 15–12)	TX FIFO maximum occupancy value (bits 11–8)	Reserved (bits 7–5)	RX FIFO maximum occupancy value (bits 4–0)

PRINCIPLES OF OPERATION

TX FIFO maximum occupancy value (bits 11–8), and RX FIFO maximum occupancy value (bits 4–0)

These registers contain the maximum occupancy value for the TX and RX FIFOs experienced during operation. An internal register keeps the current occupancy value of the FIFOs. If the value in the internal register is greater than the value in the register for more than one consecutive cycle, the value is written into the register. The values in the registers are used to measure the usage of the internal FIFOs to optimize a future revision of the TNETA1570. The reset value is zero.

scheduler-table-size register

Reserved (bits 31–16)	Reserved (bits 15–12)	Scheduler-Table-Size Register (bits 11–0)

This register contains the size (in 32-bit words) of the scheduler table. It is initialized by the host. Maximum value is 3100(dec), Minimum value is one. The reset value is zero. This register is programmed before the EN_TX is set.

software reset

Reserved (bits 31–0)

A write to this address location forces the TNETA1570 into reset. The address is decoded and an internal reset pulse with sufficient length is generated. The duration of reset is max ten clock cycles.

completion-ring address registers

TX completion ring WOI address (bits 31–0)
TX completion ring WI address (bits 31–0)
RX completion ring WOI address (bits 31–0)
RX completion ring WI address (bits 31–0)

The completion-ring address registers hold the addresses to the completion rings in host memory. The value is set by the host. The address is aligned to the size of the ring to ensure that the first entry is always on an even 2^n address boundary.

counters

HEC-error counter

This 16-bit counter counts the number of cells dropped due to HEC errors. This rollover counter causes a bit in the status register to be set on rollover. This counter is read only and is cleared when a hardware or software reset occurs. This counter does not clear on a read by the host.

unknown-protocols counter

This 32-bit counter counts the number of cells whose header is posted in the RX unknown register. This rollover counter causes a bit in the status register to be set on rollover. This counter is read only and is cleared when a hardware or software reset occurs. This counter does not clear on a read by the host.

AAL5 CPCS PDU-discard counter

This 16-bit counter counts the number of receive-AAL5 packets that were discarded because no receive free buffers are available. This rollover counter causes a bit in the status register to be set on rollover. This counter is read only and is cleared when a hardware or software reset occurs. This counter does not clear on a read by the host.

PRINCIPLES OF OPERATION

ATM-cells-received counter

This 32-bit counter counts the number of ATM cells received and not dropped. This rollover counter causes a bit in the status register to be set on rollover. This counter is read only and is cleared when a hardware or software reset occurs. This counter does not clear on a read by the host.

ATM-cells-transmitted counter

This 32-bit counter counts the number of ATM cells actually transmitted. This rollover counter causes a bit in the status register to be set on rollover. This counter is read only and is cleared when a hardware or software reset occurs. This counter does not clear on a read by the host.

control-memory maps

normal control-memory map

Control memory is broken into three sections: pointer block, transmit section and receive section. The pointer block contains the pointers to the receive free-buffer rings and the transmit and receive completion rings. The transmit section contains the scheduler table and the transmit DMA state table. The receive section contains the receive VPI/VCI DMA pointer table and the receive DMA state table. The mapping of these blocks into control memory is shown in the following table.

MEMORY BLOCK	SIZE (in 32-bit words)	CONTROL-MEMORY ADDRESS (hex)	PCI OFFSET ADDRESS (hex)
Scheduler Table	3100	00000–00C1B	00000–0306C
Reserved	228	00C1C–00CFF	03070–033FC
Reserved	256	00D00–00DFF	03400–037FC
Free-Buffer Pointers	512	00E00–00FFF	03800–03FFC
RX VPI/VCI DMA Pointer Table	4,096	01000–0IFFF	04000–07FFC
Reserved	8	02000–02007	08000–0801C
TX DMA State Table	8,184	02008–03FFF	08020–0FFFC
RX DMA State Table	245,760	04000–3FFFF	10000–FFFFC

The memory mapping is designed to make it possible for users to design the application boards to support the number of TX DMA states and RX DMA states used with a minimum number of SRAMs. This is achieved by choosing the start address of the TX DMA state table and the RX DMA state table on even boundaries.

small SRAM control-memory map

If the SMALL_MAP bit is set in the configuration register (if a maximum of 512 TX DMA states and 512 RX DMA states are to be used), address line CM AD (14) is ORed in the TNETA1570 with address line CM AD (13) and CM AD (12). The SRAM needed is 16K $\times$ 32.The hardware interprets address 04000–04FFF as address 03000–03FFF creating the following control-memory map:

PRINCIPLES OF OPERATION

small SRAM control-memory map (continued)

MEMORY BLOCK	SIZE (in 32-bit words)	CONTROL-MEMORY PHYSICAL ADDRESS (hex)	CONTROL-MEMORY LOGICAL ADDRESS (hex)	PCI OFFSET ADDRESS (hex)
Scheduler Table	3100	00000–00C1B	00000–00C1B	00000–0306c
Reserved	228	00C1C–00CFF	00C1C–00CFF	03070–33FC
Reserved	256	00D00–00DFF	00D00–00DFF	03400–37FC
Free-Buffer Pointers	512	00E00–00FFF	00E00–00FFF	03800–3FFC
RX VPI/VCI DMA Pointer Table	4,096	01000–01FFF	01000–01FFF	04000–7FFC
Reserved	8	02000–02007	02000–02007	08000–0801C
TX DMA State Table	4,088	02008–02FFF	02008–02FFF	08020–BFFC
Not Used	4,096	Not Used	03000–03FFF	0C000–FFFC
RX DMA State Table	4,096	03000–03FFF	04000–04FFF	10000–13FFC

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.