Now a product of Brooktree Corporation.
Call 1-800-843-3642

R8071A

Rockwell

071A
ISDN Link Layer Controller

APPLICATIONS

- PRIMARY RATE INTERFACES
- BASIC RATE D-CHANNEL CONTROLLER

INTRODUCTION

The Rockwell R8071A ISDN Link Layer Controller device multiplexes/demultiplexes up to 32 high speed data channels to support ISDN implementations (Figure 1). The R8071A operates at layer 2 (data link protocol level) of the Open Systems Interconnection (OSI) reference model recommended by the International Organization for Standardization (ISO) and can reside between the R8070/RT9170 Transceiver and a buffer memory shared with one or more host processors.

The R8071A processes transmit and receive data on a T1 communications medium compatible with ANSI T1.403-1989 for 1.544 Mbps systems, or at 2.048 Mbps in the CEPT PCM 30 carrier format. The device provides HDLC formatting functions for synchronous data and manages buffer memory for each of the active data channels, including the common signaling channel, using simple linked-list structures.

The R8071A is compatible with the Integrated Services Digital Network (ISDN) specified by the International Telegraph and Telephone Consultative Committee (CCITT) and supports connections of computers to the ISDN at the primary rate. It also supports clear channel transmission of data at 64 kbps, 56 kbps synchronous, and supports 64 kbps virtual circuit protocol using LAPD, respectively.

The R8071A device provides additional functions which support X.30 and X.31 rate adaption as well as ISDN and fully Flexible Hyperchannels. The device is also compatible with HDLC, SNA SDLC, X.25, X.75, LAPB and LAPD protocols. These features allow the use of the R8071A in applications that go beyond the host-end computer-PBX interface.

The R8071A complements the Rockwell R8070 T1/CEPT PCM Transceiver and the RT9170 Intelligent T1 Controller, which operates at layer 1 (physical interface level) of the OSI reference model, and provides basic T1 framing and maintenance functions of the T1 link.

The R8071A finds applications in widely diverse areas of telecommunications (including TDM machines, Central Office switches, and PBX), as well as the basic host computer-PBX links. In ISDN switching applications, the R8071A can function as a multiplexed controller for as many as 32 ISDN basic access "D" channels, and can substantially off-load LAPD processing from the Switch Central Control.

FEATURES

- ISDN implementation, solidly based on T1 and CEPT primary rate carrier with designed-in compatibility with ISDN, is the key to world-wide networking
  - — T1/CEPT provides the link beyond the central office or PBX
  - — ISDN provides the phone/terminal to the central office or PBX basic access link
  - — ISDN provides the computer to computer link and computer to PBX link
- R8071A single chip CMOS monolithic device simplifies ISDN/DMI implementation
- Provides up to 32 full duplex channels with HDLC/SDLC protocol formatting
- Provides fully programmable hyperchannel configuration
- Supports all four DMI B channel data options:
  - — Mode 0 (clear channel 64 kbps synchronous)
  - — Mode 1 (56 kbps synchronous data without or with HDLC protocol)
  - — Mode 2 (up to 19.2 kbps synchronous or asynchronous)
  - — Mode 3 (64 kbps virtual circuit service)
- Supports both DMI D channel signaling options
  - — Bit-oriented signaling (BOS)
  - — Message-oriented signaling (MOS)
- Compatible with 1.544 Mbps T1 SF and ESF as well as 2.048 Mbps CEPT PCM 30 carrier format
- Supports both flag stuffing (I.462, DMI mode 2) and RA2 intermediate rate adaption (I.460, X.30, V.110 or ECMA-102)
- Compatible with HDLC, SNA SDLC, X.25, X.75, LAPB and LAPD protocols
- On-board buffer memory management function
- On-board CRC-16 generation and checking, automatic flag detection and transmission, and zero-bit insertion and deletion
- Simple interface to Rockwell R8070 T1/CEPT PCM Transceiver and RT9170 Intelligent T1 Controller
- Backward compatible with the R8071
- Available in 64-pin quad in-line package (QUIP), 68-pin leaded chip carrier, or 68-pin pin grid array
- Operates from a single +5 Vdc supply

ORDERING INFORMATION

Part Number:
R8071A_ _

Temperature
Blank = 0°C to +70°C

Package
P = 64-Pin Plastic QUIP
S = 64-Pin Cerpac QUIP
J = 68-Pin Plastic Leaded Chip Carrier (PLCC)
JC = 68-Pin Ceramic Leaded Chip Carrier (CLCC)
G = 68-Pin Pin Grid Array (PGA)

INTERFACE

The R8071A ISDN Link Layer Controller transmits data to, and receives data from, an R8070 T1/CEPT Transceiver or the RT9170 Intelligent T1 Controller as illustrated in Figure 1. It transfers the data in 8-bit parallel form to and from an external buffer memory shared by a host computer system. (The external buffer memory is referred to as shared memory and the host computer system is referred to as the host.) The R8071A interface signals are functionally grouped in Figure 2. The R8071A pin assignments are shown in Figure 3 and the R8071A interface signals are defined in Table 1.

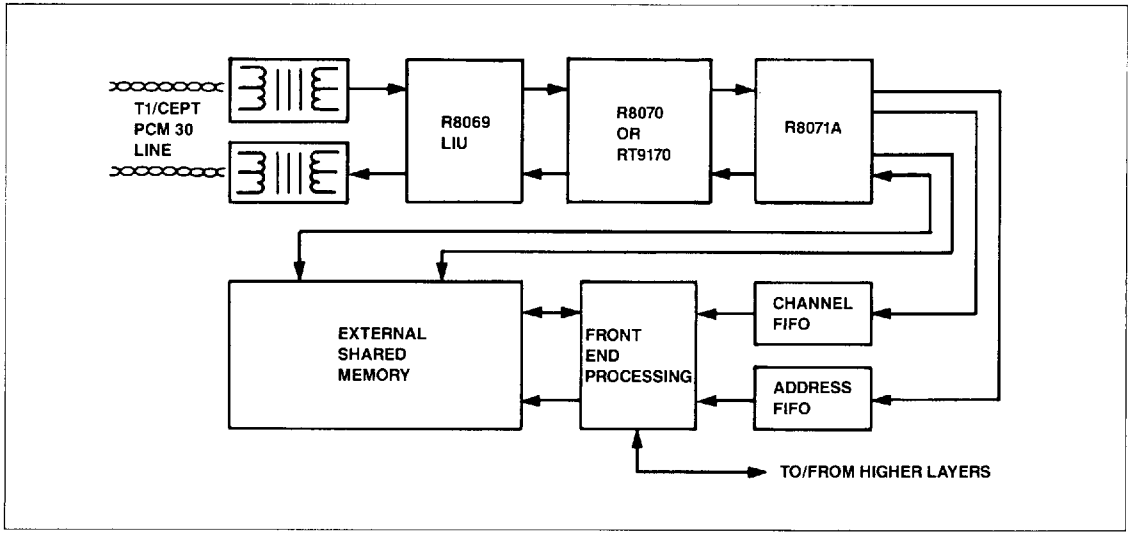


Figure 1. Functional Architecture for Host ISDN Primary Rate Access, PBX ISDN Trunkside Adapter

R8071A ENHANCEMENTS

The R8071A incorporates two major enhancements to the R8071—Partial Command Buffers and software programmable Flexible Hyperchannel. The R8071A is hardware and software compatible with the R8071.

PARTIAL COMMAND BUFFERS

This new command is incorporated to significantly lower the overhead needed to initiate the transmission of a new chain of data buffer after the transmitter of a chain has entered the idle flag (HDLC mode) or all ones (non-HDLC mode) transmission state.

The COMPLETE/PARTIAL bit of the STATUS byte of the Command Buffers is not accessed on the R8071—it is a "Don't Care" bit. The R8071A uses this bit in conjunction with the COMMAND/DATA bit to determine if the Partial Command Buffer feature is to be used. If this bit was always set to one in an R8071 system then the R8071A will work as a direct replacement for the R8071.

Please refer to the Transmit Command Buffer sector for detail description.

FLEXIBLE HYPERCHANNEL

Flexible Hyperchannel allows software to dynamically group multiple individual 64 kbps channels into hyperchannels, thus allowing users to allocate bandwidth according to channel requirement. Flexible Hyperchannel supports grouping of any number of contiguous and non-contiguous individual channels for both the T1 and the PCM 30 environments. This enhancement also supports independent transmit and receive hyperchannel assignments.

DATA LENGTH in the Command Buffers is used by the R8071A to determine if Flexible Hyperchannels are to be activated or not. If DATA LENGTH ≤ 002 then no Flexible Hyperchannel is used for this channel and the R8071A functions the same as the R8071. If DATA LENGTH > 002 then the R8071A will enter the Flexible Hyperchannel mode for this channel. The R8071 does not access DATA LENGTH in the Command Buffers. If DATA LENGTH in the Command Buffers is always ≤ 002 in an R8071 system then the R8071A will work as a direct replacement for the R8071. Please refer to the Flexible Hyperchannel sections for detail description.

The CH0–CH4 pins of the R8071 and R8071A indicate which channel is being processed at any given time. On the R8071 the channel numbers are always sequential for both the Receiver and Transmitter: the channel number increments by one until it wraps around to zero, then counts by one again. If the R8071A is not running Flexible Hyperchannels then the channel numbers operate the same as the R8071. If Flexible Hyperchannels are used then the channel numbers may not be sequential but instead display the channels as they are arranged in the Flexible Hyperchannel scheme. The channel numbers still reflect the actual channel being processed, and so may be used as part of the Dual Port Memory addressing circuit, but the R8071A system design may not depend on sequential channel numbers.

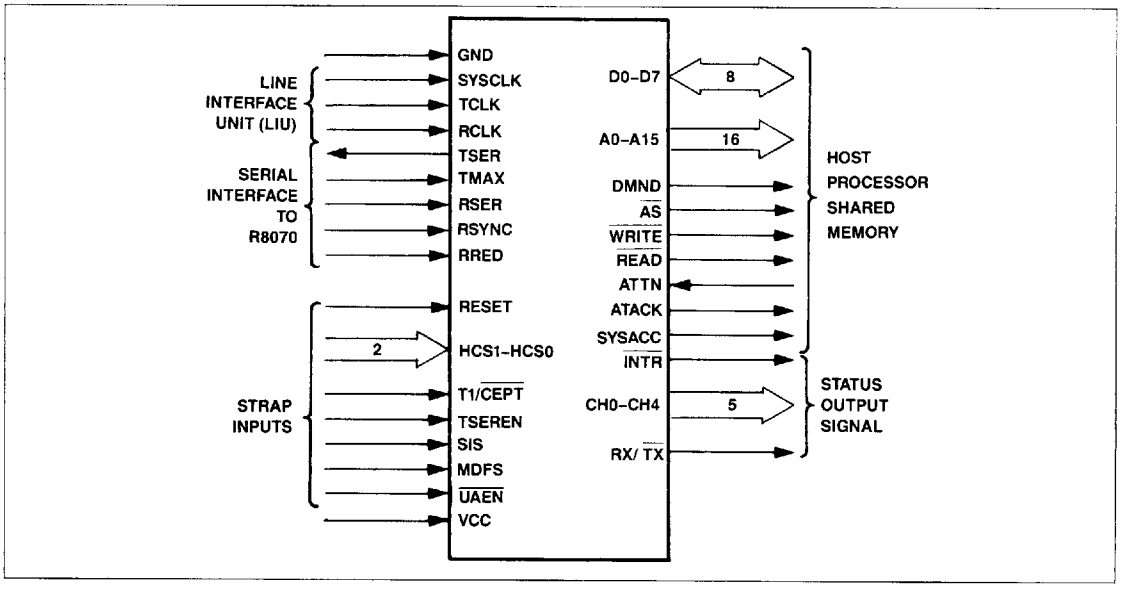
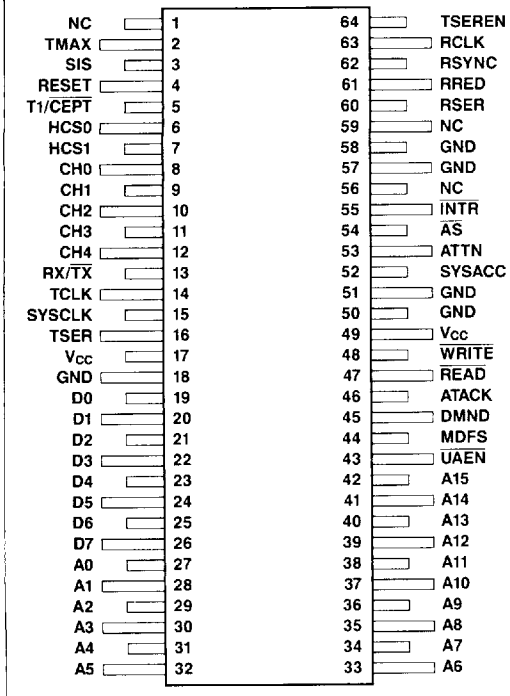
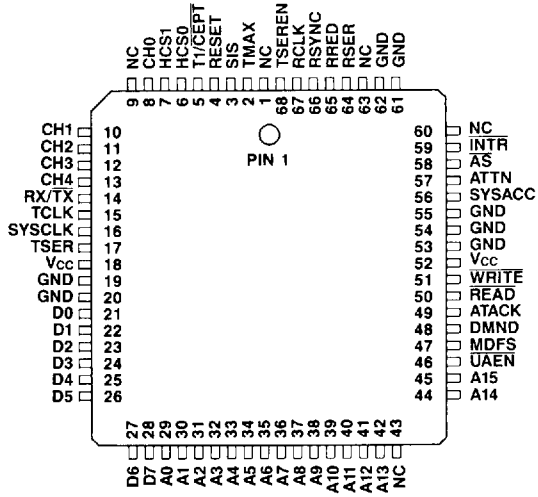


Figure 2. R8071A Interface Signals

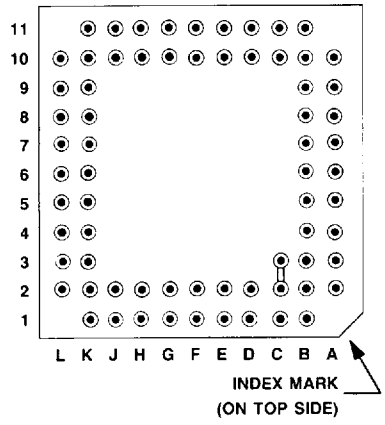


64-Pin QUIP



68-Pin PLCC

Back View of Pin Grid Array



PIN	SIGNAL	PIN	SIGNAL	PIN	SIGNAL	PIN	SIGNAL
A2	NC	B9	MDFS	F10	A6	K4	TCLK
A3	INTR	B10	A15	F11	A7	K5	TSER
A4	ATTN	B11	A14	G1	RESET	K6	VCC
A5	BMRT	C1	RSER	G2	SIS	K7	GND
A6	GND	C2	NC	G10	A4	K8	D1
A7	VCC	C10	A12	G11	A5	K9	D3
A8	READ	C11	A13	H1	HCS0	K10	D6
A9	DMND	D1	RSYNC	H2	T1/CEPT	K11	D7
A10	UAEN	D2	RRED	H10	A2	L2	CH3
B1	GND	D10	A10	H11	A3	L3	RX/TX
B2	GND	D11	A11	J1	CH0	L4	SYSCLK
B3	AS	E1	TSEREN	J2	HCS1	L5	VCC
B4	SYSACC	E2	RCLK	J10	A0	L6	GND
B5	GND	E10	A8	J11	A1	L7	D0
B6	VCC	E11	A9	K1	CH1	L8	D2
B7	WRITE	F1	TMAX	K2	CH2	L9	D4
B8	ATACK	F2	NC	K3	CH4	L10	D5

68-Pin PGA

NC = NO CONNECTION TO BE MADE TO THIS PIN.

Figure 3. R8071A Pin Assignments

Table 1. R8071A Interface Signal Definitions

Symbol	I/O	Signal Name/Description
Memory Interface		
D0–D7	I/O	**Memory Data Lines.** Bidirectional 8-bit memory data bus between the R8071A and the shared memory. D0 is the LSB and D7 is the MSB.
A0–A15	O	**Memory Address Lines.** Output lines to the shared memory. A0 is the LSB and A15 is the MSB.
DMND	O	**Memory Demand.** Active high output to shared memory. The R8071A accesses shared memory within one TCLK period after assertion (rising edge) of DMND. DMND is negated at completion of the memory access cycle.
$\overline{AS}$	O	**Memory Address Strobe.** A valid memory address is present on the memory address lines at the falling edge of the active low $\overline{AS}$.
$\overline{WRITE}$	O	**Memory Write.** Active low output to the shared memory to perform a write cycle.
$\overline{READ}$	O	**Memory Read.** Active low output to the shared memory to perform a read cycle. Data from memory is latched in the R8071A prior to the negation of $\overline{READ}$.
ATTN	I	**Attention.** Active high input that commands attention from the R8071A to shared memory locations which contain updates to status, modes, and/or buffer start address of a specified channel. A sequence of memory accesses are performed soon after ATTN is asserted. ATTN is negated in response to ATACK.
ATACK	O	**Attention Acknowledge.** Active high output asserted when the sequence of memory accesses (in response to ATTN) is complete. ATACK is negated in response to negation of ATTN.
SYSACC	O	**System Access.** Active high output asserted to indicate the R8071A is accessing one of the 129 system memory locations (Channel Activation Byte or Channel Buffer Pointers).
LIU Interface		
SYSCLK	I	**System Clock.** Square wave input from the LIU clock generator for internal use in the R8071A. Nominally, 3.088 MHz for T1 and 4.096 MHz for CEPT PCM 30.
TCLK	I	**Transmit Clock.** Square wave input from the LIU clock generator providing the master timing source for the transmit function. The frequency is one-half that of the SYSCLK.
RCLK	I	**Received Clock.** Input from the external LIU/Clock recovery for the R8071A to sample RSER, the received serial data. Nominally, 1.544 MHz for T1 and 2.048 MHz for CEPT PCM 30.
Serial Interface		
TSER	O	**Transmitter Serial Data.** Output from the R8071A to the R8070 T1/CEPT PCM Transceiver representing the transmit serial data bit stream.
TMAX	I	**Transmit Multiframe Sync.** Active high input pulse from the R8070 T1/CEPT PCM Transceiver indicating the beginning of a multiframe.
RSER	I	**Received Serial Data.** Input from the R8070 or RT9170 representing the received serial data bit stream.
RSYNC	I	**Receive Synchronization.** Active high input level or pulse from the R8070 or RT9170 for frame synchronization reference.
RRED	I	**Receive Red Alarm.** Active high input from the R8070 or RT9170 indicating a failure to frame on the T1 or CEPT PCM 30 time-division multiplexed (TDM) signals. Low indicates that frame alignment has been found.

Table 1. R8071A Interface Signal Definitions (Continued)

Symbol	I/O	Signal Name/Description
Strap Option Inputs		
T1/$\overline{\text{CEPT}}$	I	**T1 or CEPT Framing Select.** High selects the T1 framing mode. Low selects the CEPT PCM 30 framing mode.
HCS1, HCS0	I	**Hyperchannel Select.** Encoded inputs select the T1/CEPT PCM 30 hyperchannels. (See Figure 5.)

T1/$\overline{\text{CEPT}}$	HCS1	HCS0	Channel Selection
X	Low	Low	All channels are 64 Kbps
High	High	Low	Four channels of 384 Kbps (H0)[1]
High	Low	High	Single channel of 1.536 Mbps (H11)[1]
Low	High	Low	Single channel of 1.92 Mbps (H12)[2], time slots 0 and 16 are 64 kbps
X	High	High	Reserved
Low	Low	High	Reserved

Notes: "X" denotes "Don't Care".
1. Valid for T1 only.
2. Valid for CEPT only.

TSEREN	I	**TSER Enable.** Active high input that works in conjunction with the FILL/MASK bit as follows:

TSEREN	FILL/MASK Bit 0*	FILL/MASK Bit 1
High	Send a 1 on TSER	Send data on TSER
Low	High impedance output on TSER	Send data on TSER

* or any F bit, or any bit during RESET, R8071A initialization, and until a channel is activated.

SIS	I	**Serial Interface Select.** High level identifies an R8070 or RT9170 compatible serial interface. Low level identifies AT&T compatible interface.
MDFS	I	**Memory Data Format Select.** High indicates that the most and least significant bytes of next buffer start address, buffer size and data length in shared memory reside at even and odd addresses, respectively (68000 MPU word addressing compatible). Low indicates that the least and most significant bytes of next buffer start address, buffer size and data length reside at even and odd addresses, respectively (8086 MPU word addressing compatible).
$\overline{\text{UAEN}}$	I	**Upper Address Enable.** High level input causes the upper address bus lines (A8–A15) to be in the high impedance state during system shared memory access (when SYACC is asserted). Low input causes the upper address outputs to be forced low by the R8071A when SYACC is asserted.
Status Outputs		
$\overline{\text{INTR}}$	O	**Interrupt.** Active low output pulse of one SYSCLK period to the host system to indicate that the buffer status is being updated.
CH0–CH4	O	**Channel Number (0–4).** Encoded output indicating the channel number being served. CH0 is LSB and CH4 is MSB.
RX/$\overline{\text{TX}}$	O	**Receive/Transmit Channel.** Output used in conjunction with the channel number (CH0–CH4). High indicates a receive channel and low indicates a transmit channel.
Reset and Power		
RESET	I	**Reset.** Active high input that initializes all R8071A functions. Reset causes the R8071A to default to HDLC mode, causes all 0s in the FILL/MASK, and deactivates all channels. Inactive transmit channels output all 1s. R8071A initialization is complete within 90 SYSCLK periods after RESET returns low.
VCC		**Power Supply Voltage.** +5 Vdc with respect to VSS.
VSS		**Ground.** Ground reference voltage.

FUNCTIONAL DESCRIPTION

The R8071A fetches the data to be transmitted from the shared memory, processes the data for up to 32 channels, channel-by-channel, by performing protocol formatting and rate adaption, and transmits it to the transceiver in serial form. Similarly, the R8071A processes, on a channel-by-channel basis, the received serial data on up to 32 channels by performing protocol deformatting and rate adaption, and stores the data into the shared memory.

Each channel is processed depending on the operational mode specified by the host as set up in the shared memory. For any channel, the transmitter and the receiver operating modes may be specified independently of each other.

The internal functions of the R8071A are partitioned logically in to five major blocks (Figure 4):

- Transmit Bit-Level Processor
- Receive Bit-Level Processor
- Buffer Memory Manager
- Device Mode Controller
- System Monitor

TRANSMIT BIT-LEVEL PROCESSOR

HDLC and Non-HDLC Modes

The Transmit Bit-Level Processor performs basic HDLC* protocol formatting (DMI data modes 2 and 3, ISDN LAPD and IBM SNA) or other non-protocol transmit functions (DMI data modes 0 and 1, and bit-oriented signaling mode) for each channel independently of any another channel.

In the HDLC mode, it generates flags, abort and idle codes, inserts zeroes for bit transparency, computes the HDLC frame check sequence (FCS) and composes HDLC frames from the data provided in the shared memory.

In the non-HDLC data mode, the data from the shared memory is not framed.

In either mode, the R8071A performs rate adaption of sub-64 kbps data rates of the form

$$n \times 8 \text{ kbps } (n = 1 \text{ through } 8)$$

to the standard 64 kbps bearer rate (I.460, second stage RA2). An 8-bit FILL/MASK sequence (specified in in shared memory) is applied to the HDLC-formatted or non-HDLC data on a bit-by-bit basis (see FILL/MASK description on page 26). The resulting 8-bit sequence, consisting of the actual data bits and any 'time fill' bits (always a one) based on the FILL/MASK sequence, is then transmitted over the channel. Figure 5a illustrates this process.

In the HDLC mode, the R8071A adapts the standard sub-64 kbps data rates (CCITT X.1 or DMI mode 2, but not necessarily n x 8 kbps) directly to the 64 kbps bearer rate (I.462 and DMI).

A number (specified in the shared memory) of HDLC flags are appended to the end of an HDLC frame as time fill sequences.

*The R8071A does not distinguish between the the High Level Data Link Control (HDLC) and the Synchronous Data Link Control (SDLC) protocols but implements the common link-layer functions for both. Reference to HDLC in this document also implies SDLC unless otherwise stated.

The R8071A monitors the number of intentionally inserted zeroes (which may be viewed as non-data intra-frame time fill bits). The programmed number of flags are adjusted based on the number of zeroes inserted. Reset activates the HDLC mode for all channels.

Logical Inversion

Logical inversion of data, as well as abort, flag and FCS bits, before transmission is programmable. Reset activates inversion for all channels.

Loop Mode

Loop mode for both the transmit and receive channels is also programmable in shared memory. For a channel in the near-end loop mode, the R8071A stores the data transmitted during the channel period in an intermediate buffer. Such data is to be taken by the receive data channel programmed in the loop mode and eventually sent back to the shared memory. Only a single transmit channel and a single receive channel may be placed in the loop mode at one time for proper operation. Note that loop mode does not support H0 hyperchannel operation. Reset deactivates the loop mode for all channels.

Signaling

Bit-oriented signaling and LAPD-based message-oriented signaling channels are directed to the non-HDLC processing elements, without any special consideration.

ISDN Hyperchannels

The 64 kbps channels (or time slots) are grouped into the ISDN standard hyperchannels (Figure 6). One hyperchannel grouping is specified by the HCS1 and HCS0 inputs (see Table 1, Strap Input Options). Reset deactivates signaling mode for all channels.

Alternatively, the 64 kbps channels may be grouped randomly to form non-standard, fully Flexible Hyperchannels. Each hyperchannel is made up of any number of 64 kbps channels, which allows customizing the bandwidth of data channels.

Transmit Interface

The Transmit Bit-Level Processor interfaces directly to an R8070 T1/CEPT PCM Transceiver or RT9170 T1 Controller without any external circuitry. All the channel counter functions are built-in. The Transmit Serial Data (TSER) output is driven with the transmit serial data bit stream acquired from the shared memory. The Transmit Clock (TCLK) input is the timing reference for TSER. The Transmit Multiframe Sync (TMAX) input is the starting reference for the TDM frame.

TSER is placed in the high impedance (tri-state) mode whenever the TSEREN input is low and the FILL/MASK bit is zero (see Table 1, Strap Input Options). This enables the TSER outputs of up to eight R8071As to be connected together and allows the R8071As to be programmed with mutually exclusive FILL/MASK sequences in order to accomplish subrate time-division multiplexing over a 64 kbps channel.

In summary, each channel may be programmed by the host system independently of any other channel by specifying HDLC mode, FILL/MASK, data inversion, and loop mode options. Both the rate adaption recommendation, namely HDLC flag insertion and the second stage intermediate rate adaption, are implemented. In addition, various other HDLC formatted rates of the form $n \times 8$ kbps ($n = 1$ through 8) are also adapted to the 64 kbps bearer rate.

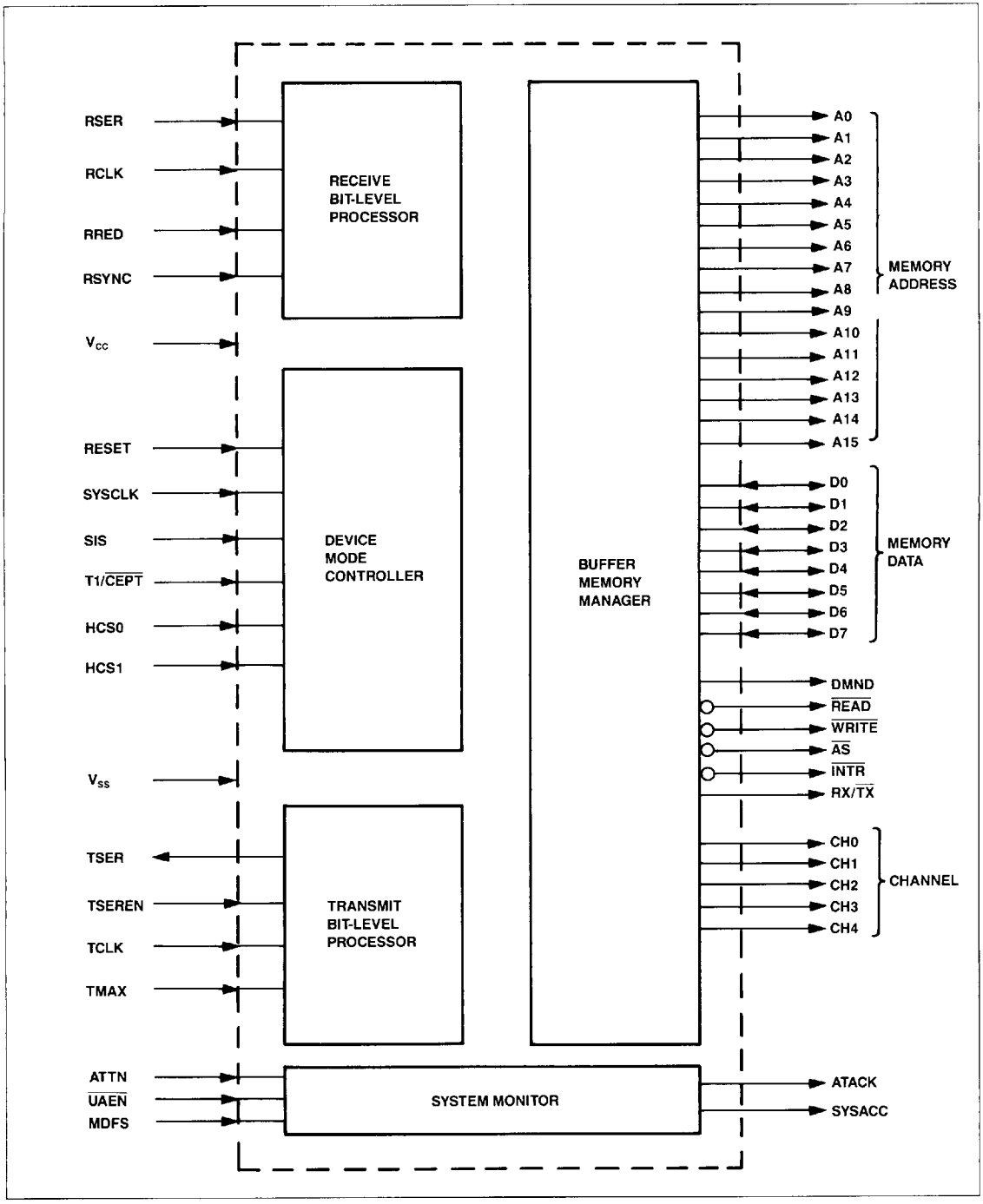
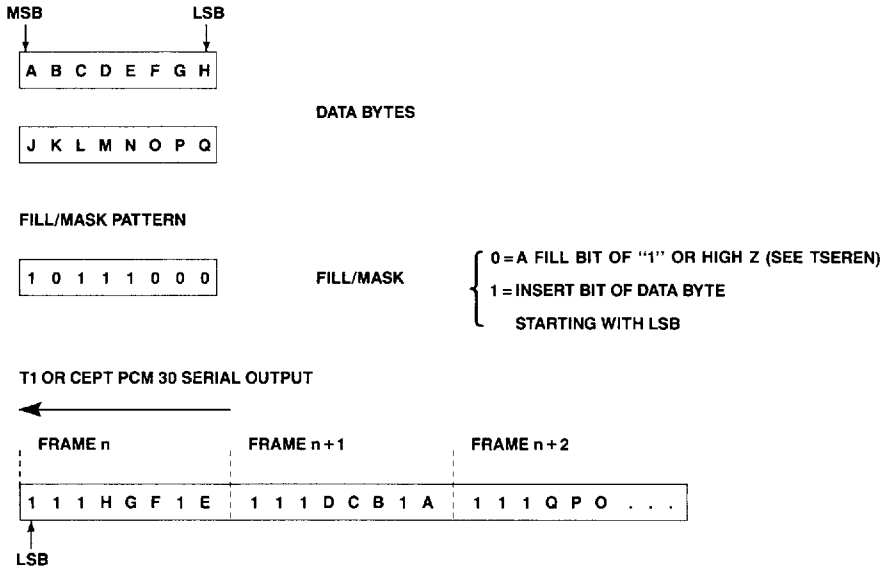
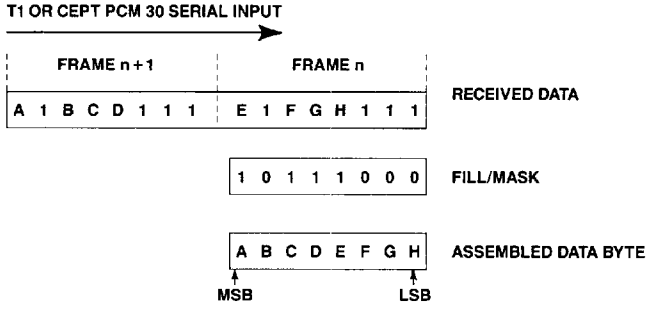


Figure 4. R8071A ISDN Link Layer Controller Functional Block Diagram

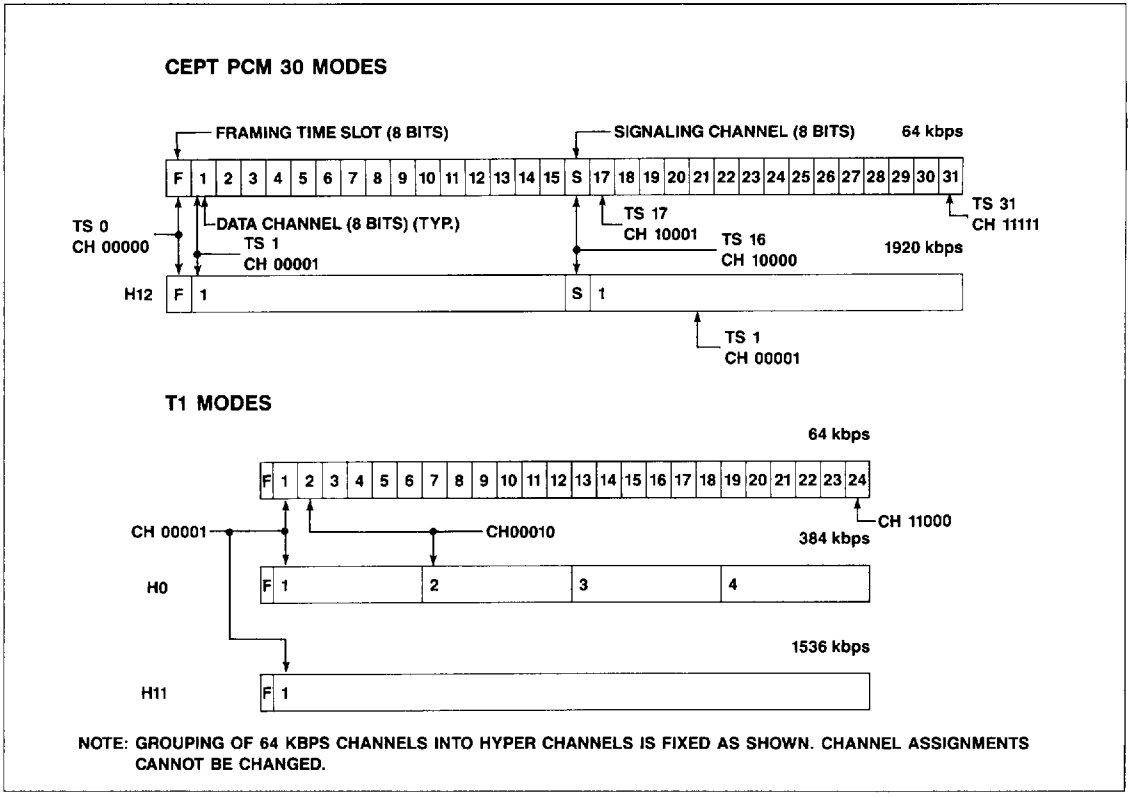


a. Single Transmit Channel



b. Single Receive Channel

Figure 5. 32 kbps Subrate Operation



NOTE: GROUPING OF 64 KBPS CHANNELS INTO HYPER CHANNELS IS FIXED AS SHOWN. CHANNEL ASSIGNMENTS CANNOT BE CHANGED.

Figure 6. R8071A Standard Hyperchannel Provisions

RECEIVE BIT-LEVEL PROCESSOR

HDLC and Non-HDLC Modes

The Receive Bit-Level Processor performs basic HDLC/SDLC protocol deformatting (DMI data modes 2 and 3, ISDN LAPD, and IBM SNA) or other non-protocol receive functions (DMI data modes 0 and 1, and bit-oriented signaling channels) for each channel independently of any other channel.

In the HDLC mode, the R8071A detects flags, aborts and inserted zeroes, and also checks the Frame Check Sequence (FCS). It also filters out any time fill patterns received by applying the 8-bit FILL/MASK sequence specified in the shared memory. The resulting serial data, including the HDLC header (i.e., address and control fields), is then assembled into bytes for storage in the shared memory (Figure 5b). The validity of every HDLC frame is checked and reported to the shared memory appropriately. Reset activates HDLC mode for all channels.

In the non-HDLC data mode, any time fill patterns received are also filtered out based on the 8-bit FILL/MASK sequence. The resulting serial data is simply grouped into bytes for transfer to the shared memory.

Logical Inversion

Logical inversion of all the received serial data is programmable in the external shared memory. Reset activates inversion for all channels.

Loop Mode

Loop mode for receive channels is also programmable in shared memory. For a channel in the near-end loop mode, the input data is taken from an internal buffer rather than from the external data. The internal buffer presumably has been filled with data from a transmit channel in the loop mode. Thereafter, the looped data is processed according to the specified mode of operation. Reset deactivates loop mode for all channels.

Non-HDLC Signaling Mode

In the non-HDLC signaling channel mode, the R8071A detects the multiframe (or extended superframe) alignment sequence for DMI bit-oriented signaling (G.732 and DMI). If a valid multiframe alignment is found, the received data is transferred to shared memory. If the multiframe alignment sequence is found to be in error, transfer of signaling data to the shared memory is suspended until a valid multiframe is detected (G.732 and DMI). Loss of multiframe is reported to the shared memory. Note that any channel(s) can be specified to receive bit-oriented signaling. This feature is very useful in central office switching applications. Reset deactivates signaling for all channels.

ISDN Hyperchannels

The 64 kbps channels are grouped into the ISDN standard hyperchannels (Figure 6) based on the input strap pins HCS0 and HCS1 (see Table 1, Strap Input Options).

Alternatively, the 64 kbps channels may be grouped randomly to form non-standard, fully Flexible Hyperchannels. Each hyperchannel is made up of any number of channels, which allows customizing the bandwidth of data channels.

Elastic Buffer

The received serial data from a T1 or CEPT PCM 30 multiframe, in general, has no relationship to the transmit data multiframe in terms of frame beginning. Both the Receive Bit-Level Processor and the Transmit Bit-Level Processor exchange data with the shared memory over a single data bus. In order to handle contention for the data bus, an elastic buffer is used in the Receive Bit-Level Processor.

The elastic buffer input is clocked by the Receive Data Clock (RCLK) and the output is re-timed using the Transmit Data Clock (TCLK). Thus the TCLK is used as a reference for both the transmit and the re-timed receive data. As a result, the shared memory access is simple and predictable. Note that the looped data bypasses the elastic buffer. Also, any overflow or underflow of the elastic buffer is reported to the shared memory for all the channels. The elastic buffer also protects the shared memory against underflow or overflow in the remote loopback (i.e., echo) mode.

Receive Interface

The Receive Bit-Level Processor interfaces directly to the R8070 without any need for additional logic. All the needed channel counters are supplied internally. Received serial data is extracted from the Received Serial Data (RSER) input bit stream on the falling edge of the Receive Clock (RCLK) input. (R8070/RT9170 mode, see SIS input.)

The Receive Synchronization (RSYNC) input provides a frame synchronization reference. The RRED input is monitored for loss of T1 or CEPT PCM 30 frame synchronization and reported to the shared memory.

BUFFER MEMORY MANAGER

The Buffer Memory Manager controls the flow of data between the Transmit Bit-Level Processor/Receive Bit-Level Processor and the data buffers in external shared memory. Shared memory is allocated for each transmit or receive channel as a linked list of buffers which are set up by the host. The shared memory is managed with minimal intervention from the host. The host simply has to allocate enough memory in the buffers such that the real-time operation of transmission and reception can take place without any data underrun or overrun, respectively.

The buffers contain information such as operational modes, buffer or HDLC frame completion status, size of the buffer, number of transmit or receive data bytes, link to the next buffer, and the transmit or receive data bytes.

The R8071A updates the status of each channel buffer as each individual buffer or HDLC frames are completed and simultaneously asserts the Interrupt Indication ($\overline{INTR}$) output to the host.

During transmission, the Empty (MPTY), Command (CMND), and Complete Frame/Partial Data Buffer (CF/$\overline{P}$) bits are monitored in the transmit status byte (see External Shared Memory Organization and Definition). The MPTY, Invalid Buffer Address (IVBA), and Underrun (UNDR) bits in the transmit status byte in shared memory are updated.

During reception, the Empty (MPTY), Command,(CMND), and Complete Frame/Partial Data Buffer (CF/$\overline{P}$) bits are monitored in the receive status byte. The MPTY, Invalid Buffer Address (IVBA), and CF/$\overline{P}$ bits in the receive status byte in shared memory are updated. Three encoded error reporting bits in the receive status byte, Abort (ABRT), Frame Check Error (FCER) and Short HDLC Frame Error (SHER), are also updated to report conditions such as invalid HDLC frame, frame check error, abort code received, loss of T1 or CEPT PCM 30 frame synchronization, loss of T1 or CEPT PCM 30 signaling channel multiframe alignment, and elastic buffer underrun or overrun.

Operational modes, loop, and invert commands as well as the FILL/MASK patterns are extracted from the transmit and receive command buffers and passed to the Transmit Bit-Level Processor and the Receive Bit-Level Processor, respectively. The modes are decoded from the HDLC Mode Select (HDLC) and Signaling Mode Select (SIG) bits. Loop and invert commands are pulled from the LOOP and INV bits, respectively.

The Buffer Memory Manager responds to host processor-initiated changes in the operational modes of a channel or relocation of the allocated buffers without affecting the operation of the other channels.

The Channel Number (CH0-CH4) and Receive/Transmit Channel (RX/$\overline{TX}$) outputs are updated to reflect to the channel being served.

The Buffer Memory Manager also causes mode changes in the Transmit Bit-Level Processor and the Receive Bit-Level Processor in response to the host processor-initiated mode changes.

DEVICE MODE CONTROLLER

The Device Mode Controller provides the central device timing and control for the other device functions. General and memory interface internal timing is derived from the System Clock (SYSCLK) input. Device reset to the other functions is distributed based on the Reset (RESET) input.

The selected carrier and framing format based on the T1/CEPT PCM 30 Carrier Select (T1/$\overline{CEPT}$) input and the encoded Hyperchannel Select inputs (HCS0 and HCS1), are passed to the Transmit Bit-Level Processor and the Receive Bit-Level Processor. The transceiver interface specified by the Serial Interface Select (SIS) input is also routed to those functions. (See Table 1, Strap Input Options.)

SYSTEM MONITOR

The System Monitor informs the Buffer Memory Manager of a host-initiated Attention (ATTN) command. Prior to asserting ATTN, the host will have set up, in shared memory, the actual channel number that needs the R8071A's attention, its mode of operation and the start address of the linked list of buffers.

Whenever the R8071A accesses the Channel Activation Byte or Channel Buffer Pointers in shared memory, the System Access (SYSACC) output is asserted indicating that system memory is being accessed. At that time, the upper order address lines (A8-A15) are placed in the high impedance state or driven low depending on whether the Upper Address Enable (UAEN) input is high or low, respectively. In the high impedance state, the host can drive the A8-A15 lines to any logic level. As soon as the R8071A completes the ATTN command processing, the Attention Acknowledgement (ATACK) output is asserted. The falling edge of ATTN causes ATACK to return low.

In addition, the relative locations of the upper (most significant) and the lower (least significant) bytes of certain 16-bit words (i.e., next buffer address, buffer size and data length) in the shared memory are determined based on the Memory Data Format Select (MDFS) input (see Table 1, Strap Input Options).

SERIAL INTERFACE TO R8070/RT9170 (SIS = HIGH)

TRANSMIT

T1 Mode (T1/$\overline{CEPT}$ input high). The serial data output (TSER) from the R8071A changes in response to the falling edge of the TCLK as shown in Figure 7. Setup and hold time periods for TSER are such that TSER can be sampled reliably at the next rising edge of the TCLK inside the R8070/RT9170 device. TSER is a tri-state output. Its actual logic level and impedance level over any bit period are determined by the combination of the corresponding FILL/MASK bit and the TSEREN input.

Transmit synchronous operation between the R8070/RT9170 and the R8071A is attained by TMAX application.

When TMAX is synchronously asserted, the R8071A will be transmitting the last bit of a frame. TMAX may be applied synchronously as frequently as a frame rate, or as seldom as when a system needs to reinitiate synchronism.

When TMAX is applied to initiate synchronism, the transmitter completes the processing of the current channel, fills the interim time with 1s (or goes high impedance — see TSEREN) and begins transmitting the first bit of time slot 1 which will occur nine or ten (CEPT/T1) bit times after TMAX.

CEPT PCM 30 Mode (T1/$\overline{CEPT}$ input low). TSER from the R8071A changes in response to the falling edge of the TCLK as shown in Figure 8. Setup and hold time periods for TSER are such that TSER can be sampled reliably at the next rising edge of the TCLK by the R8070/RT9170. TSER is a tri-state output. Its actual logic level and impedance level over any bit period are determined by the combination of the corresponding FILL/MASK bit and the TSEREN input.

RECEIVE

T1 Mode (T1/$\overline{CEPT}$ input high). The receive data (RSER) is processed serially and sampled at the negative edge of RCLK at a rate of 1.544 MHz. The R8070/RT9170 accomplishes multiframe synchronization at the third assertion of RSYNC after RRED (internally delayed) goes low. RSYNC is synchronous with the rising edge of RCLK and the first "F" bit of a multiframe. Figure 9 illustrates the timing.

CEPT PCM 30 Mode (T1/$\overline{CEPT}$ input low). RSER is processed serially and sampled by the negative edge of RCLK at a rate of 2.048 MHz. RSYNC is synchronous with the rising edge of RCLK 'and bit 1' in time slot zero of the first frame of a multiframe. Figure 10 illustrates the timing.

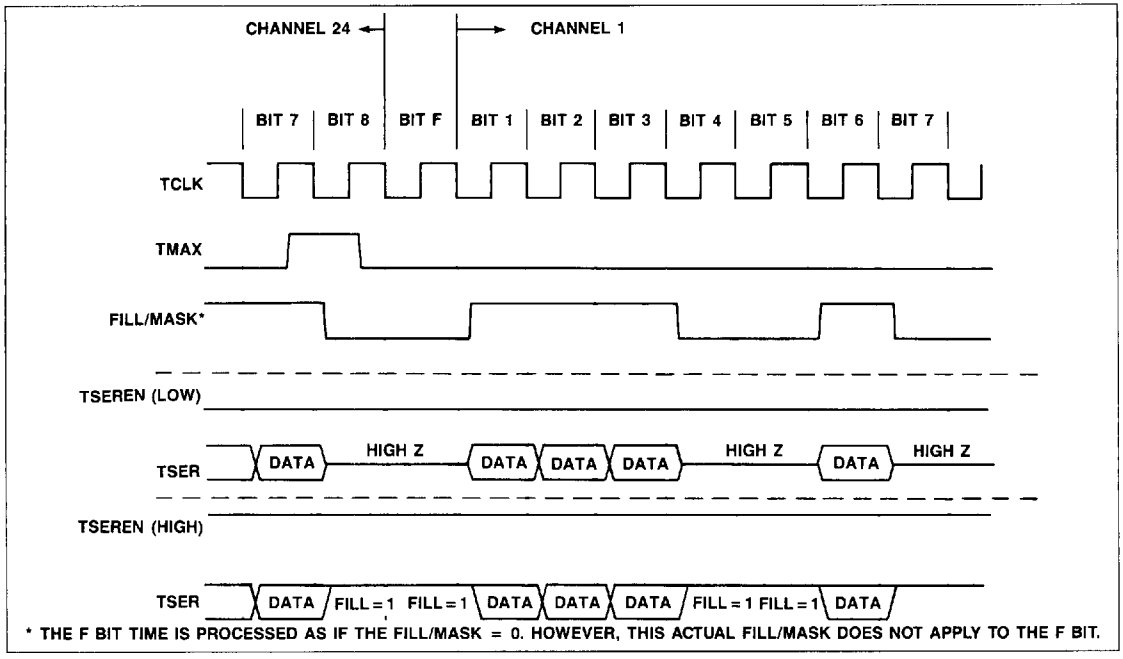


Figure 7. Transmit Frame Synchronization Timing—T1 Mode (R8070/RT9170 Interface)

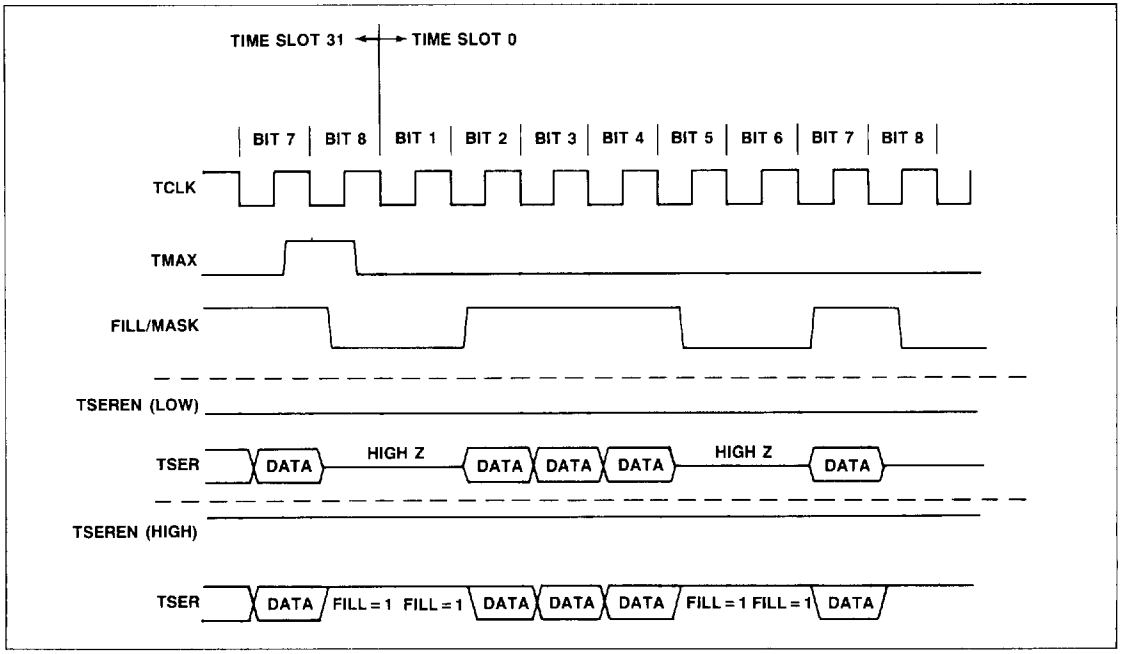


Figure 8. Transmit Frame Synchronization Timing—CEPT PCM 30 Mode (R8070 Interface)

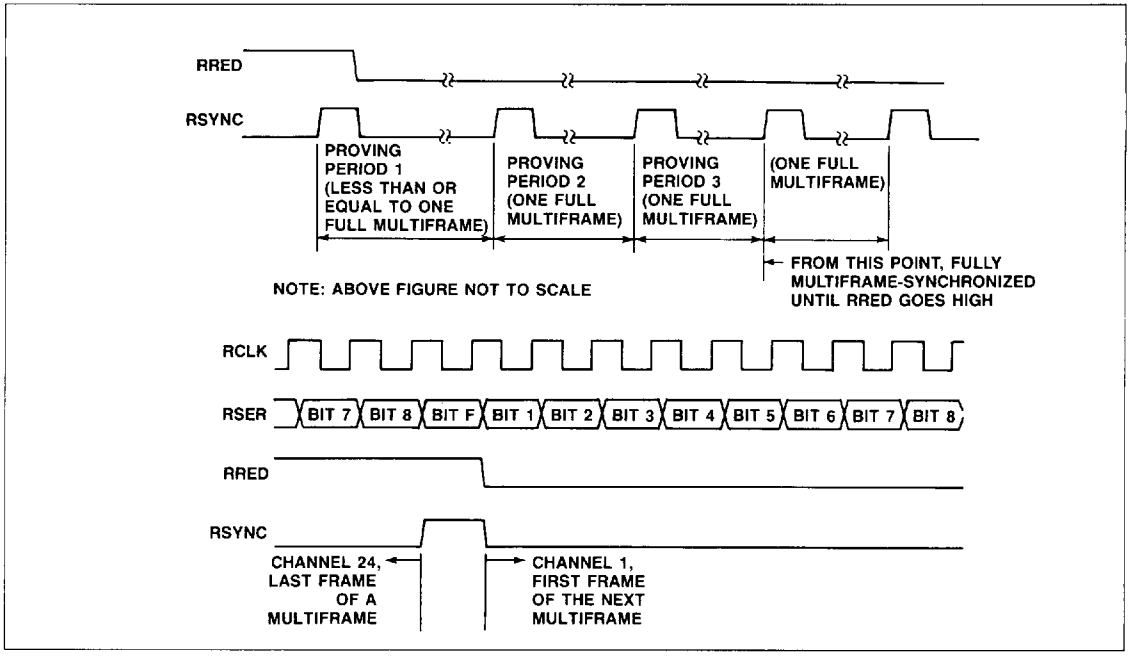


Figure 9. Receive Frame Synchronization Timing—T1 Mode (R8070/RT9170 Interface)

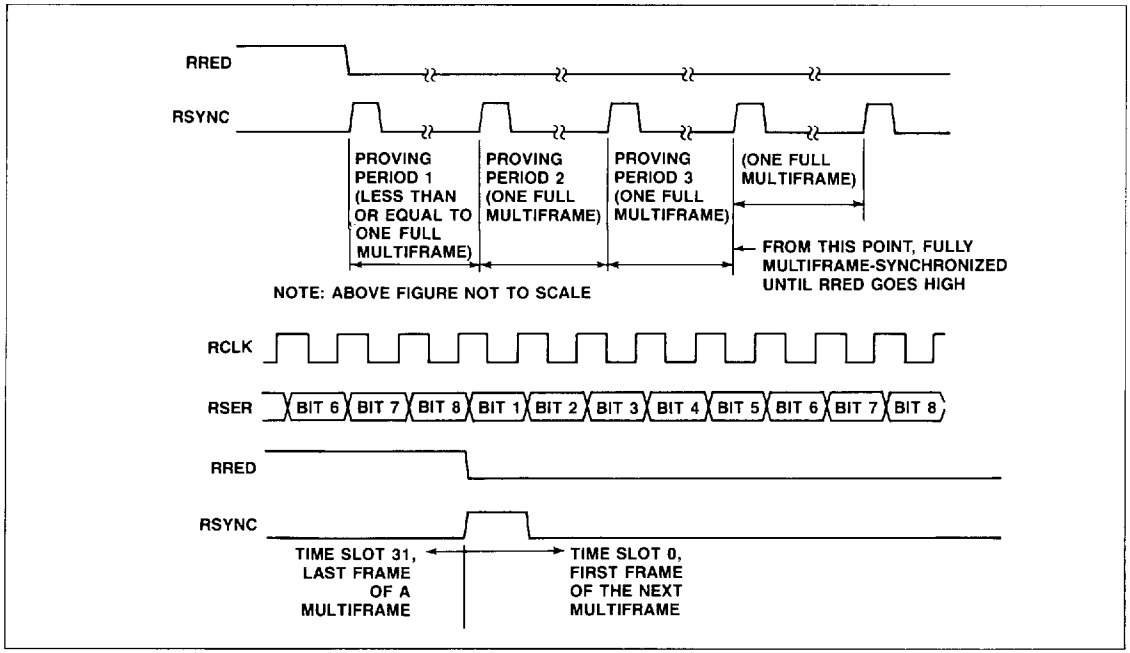


Figure 10. Receive Frame Synchronization—CEPT PCM 30 Mode (R8070 Interface)

EXTERNAL SHARED MEMORY ORGANIZATION AND DEFINITION

GENERAL STRUCTURE

Transmit data, received data, channel commands and channel pointers are organized in the external memory shared by the host and R8071A.

For each transmit and receive channel, the host must allocate shared memory for a channel activation byte, channel buffer pointers, and a set of channel data buffers. Figure 11 illustrates an arrangement of shared buffer memory within the host computer main memory.

The Channel Activation Byte and the Channel Buffer Pointers are located in a 256-byte address space referred to as system memory (Figure 12). The Channel Activation Byte is located at address j followed by 127 unassigned bytes. The Channel Buffer Pointers are located at addresses j + 128 through j + 255. The Channel Data Buffers are located at starting addresses specified by the channel pointers. The length of the data buffers is specified in data descriptors included within the data buffer.

CHANNEL ACTIVATION BYTE

The Channel Activation Byte (Figure 12) contains a command to activate or deactivate the channel number identified within the byte. The direction of data travel is also specified. The individual bits are defined as follows:

ACTIVE—Activate Channel. When set by the host, the indicated channel (CHANNEL number) is activated. When reset by the host, the channel is deactivated.

RX/$\overline{\text{TX}}$—Receive/Transmit. When set by the host, the indicated channel is a receive channel. When reset by the host, the channel is a transmit channel.

CHANNEL—Channel Number. Set by the host to select the number of the channel:

Bit 4	3	2	1	0	Channel Number
0	0	0	0	0	0
	.				
	.				
	.				
1	1	1	1	1	31

CHANNEL BUFFER POINTERS

The Channel Buffer Pointers specify the start addresses for the Channel Data Buffers. The pointers must be stored by the host in 128 contiguous bytes beginning at j + 128. For each channel, the buffer start address is to be specified as a 16-bit (2-byte) word. The relative location of the upper and lower bytes of the 16-bit word is determined by the MDFS input (Figure 12). Pointers for up to 32 transmit and 32 receive channels can be specified.

The upper address lines (A8-A15) are placed in the high impedance state by the R8071A during the System memory accesses (i. e., while the Channel Activation Byte or Channel Buffer Pointers are being accessed) when the UAEN input is high.

CHANNEL DATA BUFFERS

A buffer is a group of contiguous memory locations for each meaningful group of ordered data. The number of memory locations in a buffer depends on memory availability and user data frame size, if any. For example, a group of contiguous memory locations may be assigned to the data that has to be framed according to HDLC protocol. The data has to be grouped into an 8-bit entity, also referred to as octet or byte. If necessary, one octet is read from or written to the data buffer by the R8071A during a single memory access.

The Channel Data Buffers (also referred to as the data buffers) may reside anywhere in the memory within the addressing range of the Channel Buffer Pointers. Any number of data buffers may be assigned for any channel and their starting addresses may be changed at any time. A set of data buffers are usually assigned for each active channel for storing the data to be transmitted and another set for the received data.

TRANSMIT DATA/COMMAND BUFFER ORGANIZATION

A general organization of data within a buffer and the linking of buffers are illustrated in Figure 13. The detail contents of a Transmit Data Buffer are shown in Figure 14. The contents of a Transmit Command Buffer are shown in Figure 15. Information within the buffer is organized into two groups: descriptors and data.

The first group of seven bytes contains the buffer descriptors, i.e., information such as the link (pointer) to the next transmit data buffer, buffer size, the number of data bytes in the buffer and buffer status. This group of information is mandatory for each buffer.

The second group contains k-bytes of information, k being a variable number. This group contains user data (including any header) that has to be framed according to HDLC, data to be transmitted unframed, or channel mode and fill/mask information.

TRANSMIT CHANNEL DESCRIPTORS

The breakdown and the ordering of the seven bytes of descriptors appears in Figure 14. The first six bytes contain the next buffer address, the buffer size and the data length—each consisting of two bytes. The relative locations of the upper and the lower bytes are interchangeable by the use of the input strap pin MDFS. The seventh byte contains the status of the current transmit buffer as well as the status of the transmit channel. A byte not used by the R8071A, and free to be used by the host, will proceed or follow the status byte as determined by MDFS (Figure 14).

Next Buffer Start Address

Bytes 0 and 1 contain the 16-bit start address of the next buffer. The next buffer start address can be the same as that of the current buffer. Such a buffer is referred to as a recirculating buffer.

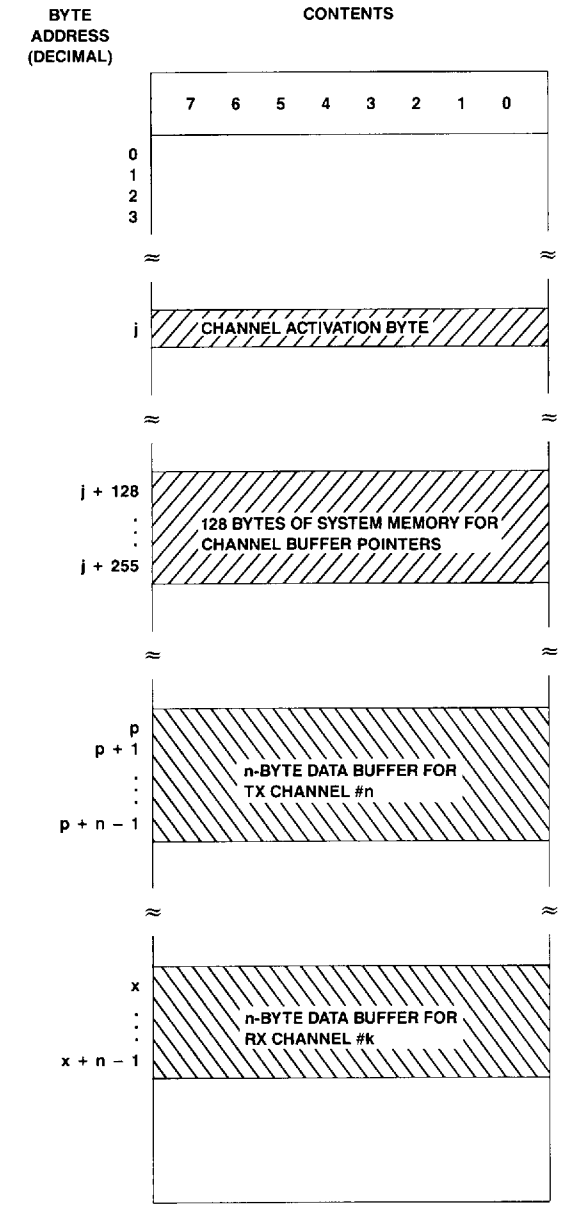


Figure 11. External Shared Memory Map—Top Level

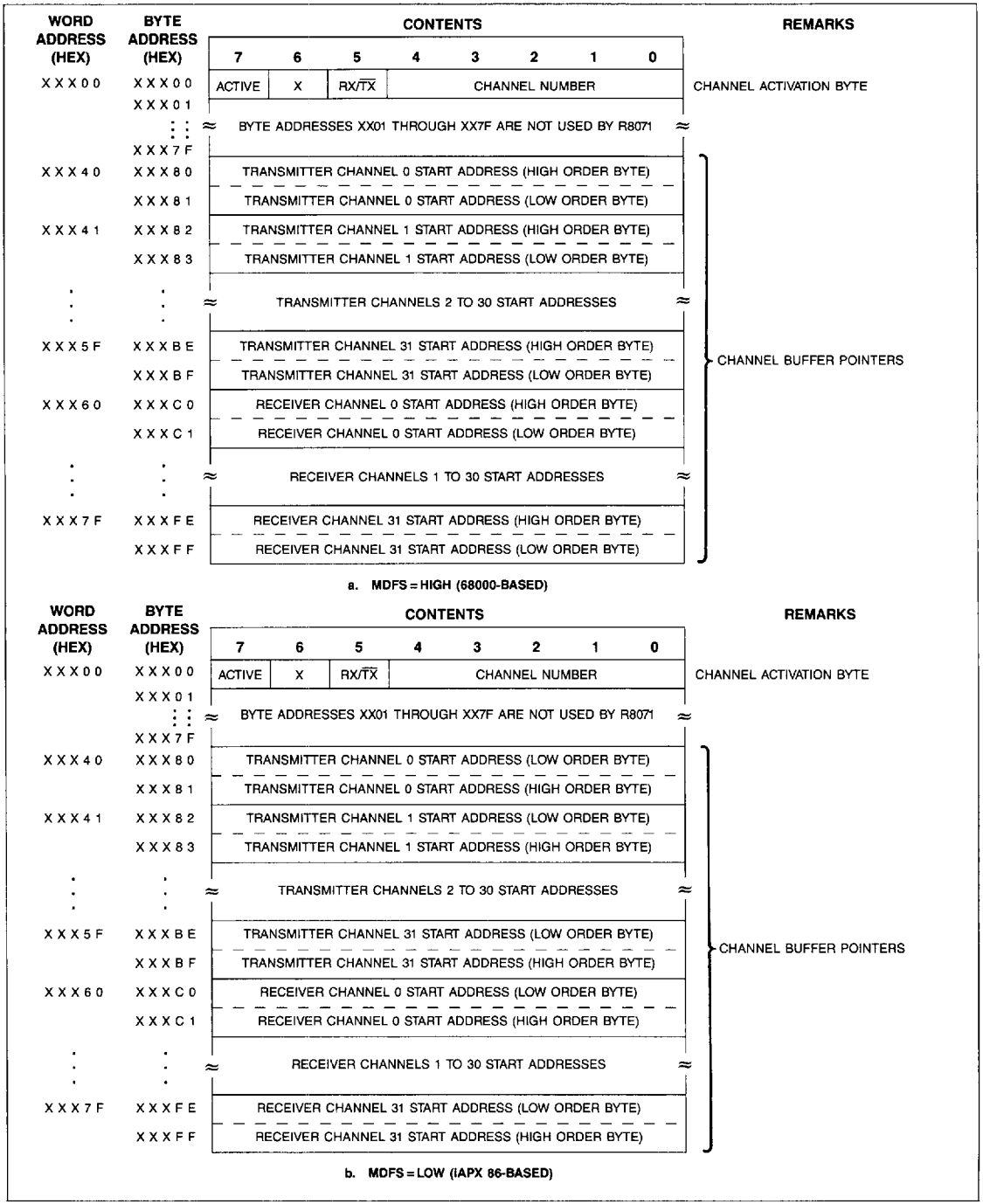


Figure 12. System Memory Map Locations

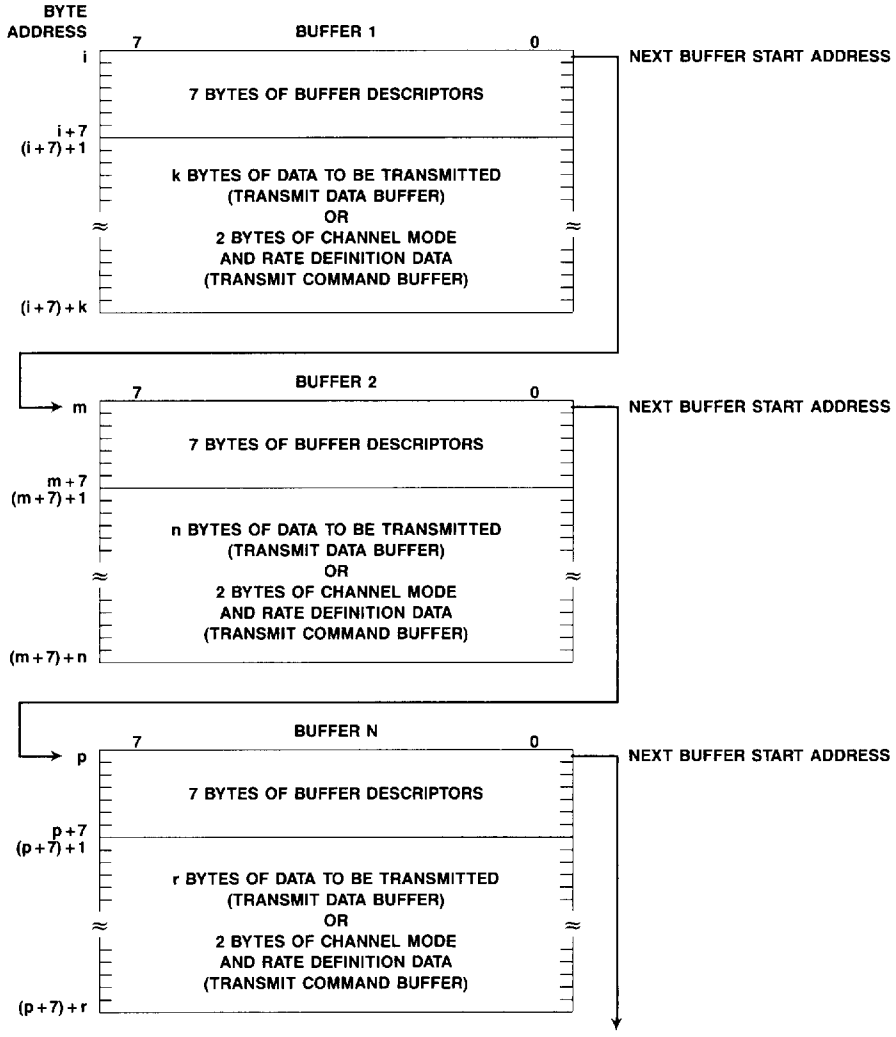


Figure 13. Organization and Linking of Transmit Buffers.

BYTE ADDRESS	CONTENTS 7	6	5	4	3	2	1	0
i	msb NEXT BUFFER ADDRESS							
i+1								lsb
i+2	X	X	X	X	msb BUFFER SIZE (k)			
i+3								lsb
i+4	FC	FO	X	X	msb DATA LENGTH (j)			
i+5								lsb
i+6	NOT USED BY R8071A							
i+7	STATUS UNDR	IVBA	X	X	X	CF/$\overline{P}$	(0) CMND	MPTY
(i+7)+1	FIRST DATA BYTE							
(i+7)+2	SECOND DATA BYTE							
(i+7)+j	LAST DATA BYTE							
(i+7)+j+1	FLAG COUNT (OPTIONAL)							
(i+7)+k	LAST LOCATION IN BUFFER							

a. MDFS = High

BYTE ADDRESS	CONTENTS 7	6	5	4	3	2	1	0
i	NEXT BUFFER ADDRESS							lsb
i+1	msb							
i+2								lsb
i+3	X	X	X	X	msb BUFFER SIZE (k)			
i+4								lsb
i+5	FC	FO	X	X	msb DATA LENGTH (j)			
i+6	STATUS UNDR	IVBA	X	X	X	CF/$\overline{P}$	(0) CMND	MPTY
i+7	NOT USED BY R8071A							
(i+7)+1	FIRST DATA BYTE							
(i+7)+2	SECOND DATA BYTE							
(i+7)+j	LAST DATA BYTE							
(i+7)+j+1	FLAG COUNT (OPTIONAL)							
(i+7)+k	LAST LOCATION IN BUFFER							

b. MDFS = Low

Figure 14. Transmit Data Buffer Contents

BYTE ADDRESS	CONTENTS 7	6	5	4	3	2	1	0
i	msb NEXT BUFFER ADDRESS (i)							
i+1								lsb
i+2	NOT USED BY R8071A							
i+3								
i+4	NOT USED BY R8071A							msb
i+5	DATA LENGTH (j)							lsb
i+6	NOT USED BY R8071A							
i+7	STATUS UNDR	IVBA	x	x	x	CF/$\overline{P}$	(1) CMND	MPTY
(i+7)+1	MODES 0	0	0	0	INV	LOOP	SIG	HDLC
(i+7)+2	FILL/MASK							
(i+7)+3	E	A	X	CHANNEL NUMBER				
(i+7)+2+j	E	A	X	CHANNEL NUMBER				

a. MDFS = High

BYTE ADDRESS	CONTENTS 7	6	5	4	3	2	1	0
i	NEXT BUFFER ADDRESS (i)							lsb
i+1	msb							
i+2	NOT USED BY R8071A							
i+3								
i+4	DATA LENGTH (j)							lsb
i+	NOT USED BY R8071A							msb
i+6	STATUS UNDR	IVBA	x	x	x	CF/$\overline{P}$	(1) CMND	MPTY
i+7	NOT USED BY R8071A							
(i+7)+1	MODES 0	0	0	0	INV	LOOP	SIG	HDLC
(i+7)+2	FILL/MASK							
(i+7)+3	E	A	X	CHANNEL NUMBER				
(i+7)+2+j	E	A	X	CHANNEL NUMBER				

b. MDFS = Low

Figure 15. Transmit Command Buffer Contents

Buffer Size

Bytes 2 and 3 contain the 12-bit BUFFER SIZE, k. The BUFFER SIZE specifies the total number of memory bytes allocated by the host processor for storing the data to be transmitted. The four most significant bits are not used by the R8071A.

The R8071A reads the buffer size only if the status indicates that the data buffer contains partial data and then interprets the buffer size to be the actual number of data bytes in this buffer. The R8071A does not read this word when the data buffer is a command buffer.

Data Length

Bytes 4 and 5 contain the 12-bit DATA LENGTH field and a 2-bit field containing host processor options for rate adaptation and timer functions. The remaining 2-bit field is not used by the R8071A.

Data Length—DATA LENGTH, j, specifies the actual number of data bytes in a data buffer to be transmitted. The R8071A reads DATA LENGTH only if the status indicates that the buffer contains the last byte of an HDLC frame or non-HDLC data (CF/$\overline{P}$ = 1).

In a command buffer, data length is used to define the hyperchannel configurations.

FO—Flag Offset Control. This bit is meaningful only when FC is set. When FO is set by the host, the transmitter of the specific channel counts the total number of HDLC zeroes intentionally inserted over the data and the CRC fields for the entire duration of transmission. At the end of each HDLC frame, it divides the accumulated number by eight and retains the remainder. The quotient is known as the flag count offset. The quotient represents the number of bytes of non-data entities transmitted and can be viewed as the HDLC intra-frame fill. The R8071A subtracts the flag count offset from the FLAG COUNT which was specified without the knowledge of the inserted zeroes. The resultant count equals the actual number of additional flags that are transmitted by the R8071A. This is extremely useful in synchronous data rate-adaption applications.

The flag count offset can be any number from zero through three, implying that HDLC frames long enough to cause up to 24 zero insertions can be monitored by the transmitter without any overflow of the internal 2-bit flag count offset. If the resultant flag count after subtracting the offset is zero or negative, no additional flags are transmitted.

When FC is a 0, the FLAG COUNT is not adjusted. For applications such as LAPD which require an opening flag and a separate closing flag, FC should be set, FO reset, and FLAG COUNT set to 1.

In a non-HDLC mode, the flag count offset will always be zero. The state of FO does not matter.

FC—Flag Control. When set by the host, this bit specifies that the corresponding HDLC channel transmitter contains a certain number of HDLC flags after the CRC. A minimum of one FLAG that plays the dual role of the closing FLAG of the current frame and the opening FLAG of the next frame is sent regardless of FC. The actual number of additional flags to be transmitted when FC is a 1 is dependent on the optional FLAG COUNT byte shown in Figure 14.

This flag control feature is very useful in rate adaption of sub-64 kbps data rates to the 64 kbps bearer channel rate and also as a timer. The R8071A automatically goes to the next buffer after sending the specified number of flags.

In the non-HDLC data mode, the R8071A sends the specified number of all ones octets after the last data byte.

For the non-HDLC signaling channel, buffers need to be specified to be partial data buffers for meaningful operation. In such a case, FC will not be read.

When FC is reset, no additional flags are transmitted. The R8071A does not process the FC bit in a command buffer.

Transmit Buffer Status

The Transmit Buffer STATUS byte contains the status of the current transmit buffer as well as the status of the transmit channel (Figure 14). The individual bits are defined as follows.

MPTY—Empty. This bit is set by the host to inform the R8071A that the data buffer is empty, i.e., data is not ready for transmission. The host resets this bit when the buffer contains valid data ready for transmission. When the buffer is empty, the R8071A keeps polling this bit until it is non-empty.

This bit is set by the R8071A to inform the host that the R8071A has completed transmission of all the data in this buffer or completed processing a command buffer.

CMND—Command. This bit is set by the host to inform the R8071A that this buffer is a command buffer. A command buffer contains channel-specific mode definition and FILL/MASK information. This bit is reset by the host to indicate that the buffer is a data buffer which contains transmit data. Upon writing status, the R8071A will update the CMND bit according to the buffer type just processed.

CF/$\overline{P}$—Complete Frame/Partial Data Buffer. This bit is set by the host to indicate that a data buffer contains the last byte of a sequence of bytes to be formatted according to HDLC. The R8071A automatically appends CRC and flag to the data before looking for more data in the next buffer. The actual number of data bytes is specified by the 12 bit DATA LENGTH words.

In non-HDLC applications, this bit must be reset to indicate continuous data transmission; otherwise the all ones octet pattern will be transmitted after the last byte of data in a buffer.

This bit is reset by the host to indicate that this buffer contains only a part of the data to be transmitted; the rest perhaps is in one or more succeeding buffers. Such a buffer is referred to as a partial data buffer. In this case, the R8071A transmits all the data in this buffer and then automatically transmits any data in the next buffer. The actual number of data bytes is specified by the 12-bit BUFFER SIZE word.

This bit is also used in a command buffer to indicate Partial or Complete Command Buffer.

IVBA—Invalid Buffer Address. This bit is set by the R8071A if it encounters an invalid next buffer address, i.e., a next buffer address with a starting address of 16 zeroes or hexadecimal FFFX (X = don't care). In this case, the specific transmit channel of the R8071A enters the inactive state and continuously transmits octets of all ones until a channel is reactivated by the host.

UNDR—Underrun. This bit is set by the R8071A when its transmit channel runs out of data. Such is the case when the R8071A encounters either an invalid buffer address, an empty data buffer, or a command buffer following a partial data buffer. In HDLC mode, the transmitter of the specific channel automatically transmits an ABORT code, followed by FLAGs until the condition is cleared. In all cases of underrun, the non-HDLC transmit channel sends the all ones octet pattern repeatedly until a valid non-empty data buffer is set up by the host. The remaining bits in the status byte will not be read by the R8071A, however, they will be reset upon a status update.

TRANSMIT DATA BUFFER

A transmit data buffer contains actual data to be transmitted and optional FLAG COUNT byte for rate adaption (Figure 14).

TRANSMIT COMMAND BUFFER

A transmit command buffer contains modes, fill/mask, and optional hyperchannel configuration data following the seven bytes of descriptors (Figure 15).

The first byte (MODE) defines the channel modes of operation—specifically HDLC, signaling, data inversion and loop back. The second byte (FILL/MASK) defines the data rate. The breakdown and the ordering of bytes within the command buffer are illustrated in Figure 15.

For a command buffer, the R8071A will not process the bytes at addresses [i + 2, i + 3] but will read the data length. The R8071A will read the next buffer address at locations i and i + 1 as part of processing the command buffer. As mentioned before, the relative locations of the upper and the lower bytes (of the next buffer address and data length) are interchangeable by means of MDFS. The mode and FILL/MASK bytes locations are not interchangeable by MFDS.

MODES

The MODES byte specifies the operational modes of the given channel—specifically, HDLC or non-HDLC, signaling channel or not, data to be inverted bit-by-bit prior to transmission or not and channel transmit data to be looped back via the receiver to the host shared memory or not (Figure 14).

SIG and HDLC— Mode Select. These two bits select the R8071A framing mode.

SIG	HDLC	Mode Selected
1	0	Non-HDLC Signaling Channel Mode
0	0	Non-HDLC Data Channel Mode
0	1	HDLC Data Channel Mode
1	1	Reserved

Non-HDLC Signaling Channel Mode. The channel carries bit-oriented signaling data without an HDLC format. The R8071A treats this channel without any special consideration to signaling. However, the R8071A assumes that only one, or at the most two, linked data buffers are assigned to the signaling channel by the host. Additionally, the last data buffer (even if it is the only buffer) is assumed to be a recirculating buffer.

Non-HDLC Data Channel Mode. The channel is a non-HDLC data channel. In DMI applications, data modes 0 and 1 may be specified by this combination. The CF/$\overline{P}$ bit of the status byte of the allocated data buffers must be reset for uninterrupted data transmission, otherwise, the R8071A will transmit the all ones octet pattern repeatedly after the last byte as many times as is dictated by FC, FLAG COUNT and the availability of the data in the next buffer. The channel time fill and the idle codes are one and the same.

HDLC Data Channel Mode. The channel is an HDLC data channel or a LAPD message-oriented HDLC signaling channel. No distinction is made between an HDLC data channel and a LAPD channel. No special handling is done on the header, i.e., address and control fields of the HDLC frame. The information field is assumed to be an integer number of octets or bytes. The 16-bit CRC-CCITT generator polynomial, $X^{16} + X^{12} + X^5 + 1$ is used for calculating the FCS. The transmitted ABORT sequence has 14 consecutive ones to satisfy SDLC and HDLC requirements.

INV—Invert Data. When set by the host, the R8071A inverts the data prior to transmission whenever the channel is active. When reset by the host, the R8071A sends the data non-inverted, i.e., as it is read from the transmit buffer. With INV bit set by the host (when the channel is idle), an octet of eight zeroes (0s) is sent for HDLC or non-HDLC channels. However, when INV is not set, an octet of eight ones (1s) is sent for idle code. All other data including HDLC flag and ABORT is conditioned by the INV bit.

Note that the combination of the HDLC procedure and data inversion guarantee that there will not be more than five consecutive zero bits in any primary rate channel during data transmission or seven consecutive zero bits during ABORT transmission.

LOOP—Loop Mode. When set by the host, the associated transmit channel data is stored internally in R8071A in addition to being transmitted. If the LOOP bit for the corresponding receive channel is also set, the previously stored transmit channel data can be looped back to the shared memory through the receive channel.

Only one channel can be placed in LOOP mode at any time for reliable loop operation. But the loop channel number, the FILL/MASK and its mode can be specified independently for any transmit and receive channel and need not be identical. Such a provision makes possible powerful software-based diagnostics routines. Hyperchannel LOOP mode is not supported by the R8071A.

Note that the loop mode operation will fail without the host being informed if the host programs only a transmit channel in the loop mode without programming a receive channel. However, the host shared memory will still be filled with the external serial data of the channel, if the channel is active.

FILL/MASK

The second byte of a command buffer contains the FILL/MASK pattern. It is used as a masking pattern on the HDLC-formatted (including FLAG, header, data, CRC and ABORT code) or non-HDLC data to adapt subrates that are multiples of 8 kbps to the 64 kbps rate.

Zero Byte Buffers

Zero Byte Buffers allow the R8071A to fill the gap between Transmit data buffers with either flags (HDLC) or 1s (non-HDLC). For example, a channel is expected to empty out its Transmit data buffer before the host controller can get the next buffer ready. So, the data buffer's NEXT BUFFER ADDRESS points to a Zero Byte Buffer, which automatically sends flags or 1s until it's told to continue on with data transmission. Two types of command buffers are provided in the R8071A for this purpose: the COMPLETE COMMAND and the PARTIAL COMMAND.

Complete Command Buffer

This buffer is coded with both the CMND bit and CF/$\overline{P}$ bit of the status byte in the command buffer set. This is processed by the R8071A as a COMPLETE COMMAND in which case the MODES, FILL/MASK and optional Flexible Hyperchannel information will be read and processed. When processing this type of buffer, the R8071A will transmit an HDLC ABORT if it is in the HDLC mode. This is the only method implemented in the R8071.

Partial Commands

This buffer is coded with the CMND bit set and CF/$\overline{P}$ bit reset. This is processed as a PARTIAL COMMAND, in which case the R8071A reads the NEXT BUFFER ADDRESS, sends one HDLC flag or one non-HDLC all 1s octet, and proceeds to the new buffer chain (without change of MODE or FILL/MASK) as if a normal buffer completion has taken place by setting the MPTY and CF/$\overline{P}$ bits of the Status register. If the NEXT BUFFER ADDRESS is null the channel will be deactivated, with the status update indicating the appropriate error condition. Each PARTIAL COMMAND sends one HDLC flag or one non-HDLC all 1s octet. Chaining PARTIAL COMMAND buffers sends multiple flags or 1s. If a PARTIAL COMMAND buffer is processed after an incomplete data frame, then HDLC **ABORT** or non-HDLC all 1s will be sent since the initial buffer chain was not properly terminated.

FLEXIBLE HYPERCHANNELS

Flexible Hyperchannels are created by grouping any number of channels into a hyperchannel. These channels may be non-contiguous. Any number of hyperchannels may operate along with any number of channels. A channel may be assigned to only one hyperchannel. Any Transmit channel may be grouped into a particular hyperchannel by configuring the hyperchannel's transmit command buffer. (Figure 15) DATA LENGTH in the command buffer determines Flexible Hyperchannel or normal channel assignment; i.e.:

Command Buffer DATA LENGTH = 0,1,2:normal channel process
Command Buffer DATA LENGTH > 2 :hyperchannel process

In Flexible Hyperchannel mode, the least significant byte of DATA LENGTH is read to determine the number of additional channels to be assigned to the hyperchannel. The data bytes containing the additional channel numbers are read from the command buffer in sequence and passed to the Transmitter. Previously active channels may be appropriated into a Flexible Hyperchannel. When they are appropriated, buffer activity on the original channel ceases; when released from the hyperchannel, the original channel will return to the state from which its activity was suspended. The channel map may be updated within the time of a single pass through the channel counter. The number of the channel that activates the Flexible Hyperchannel becomes the number of the hyperchannel. Each data byte in the command buffer is structured as shown in Figure 15.

$\underline{E}$	$\underline{A}$	
0	X	Hyperchannel assignment remains unchanged
1	0	Delete channel number in bits 0–4 from hyperchannel
1	1	Add channel number in bits 0–4 to hyperchannel

Whenever a hyperchannel is deactivated, it is automatically cleared to the default channel value. This takes one frame time. The R8071A supports independent transmit and receive hyperchannel assignments. When using Flexible Hyperchannel, the HCS0 and HCS1 bits should both be reset to zero. Otherwise, the standard hyperchannel defined by these two pins supercedes Flexible Hyperchannel.

TRANSMIT CONSIDERATIONS

Minimum Number of Data Bytes in a Buffer

There is a minimum number of data bytes required in each buffer in order for the R8071A to perform the buffer maintenance and still effect a smooth transition to the next buffer. The minimum number of bytes depends on the type of the buffer and that of the following buffer. Table 2 gives a summary.

Table 2. Minimum Number of Data Bytes

Current Buffer Status as Set Up by Host			Next Buffer Status as Set Up by Host			Min. No. of Data Bytes in Next Buffer*	Remarks
CMND	MPTY	CF/$\overline{P}$	CMND	MPTY	CF/$\overline{P}$		
1	0	X	0	0	1	2	Complete frame buffer following a command buffer
1	0	X	0	0	0	5	Partial data bufffer following a command buffer
X	X	X	1	0	X	2	Command buffer following any buffer
0	0	1	0	0	0	5	Partial data buffer following a complete frame buffer
0	0	1	0	0	1	2	Complete frame buffer following a complete frame buffer
0	0	0	0	0	1	3	Complete frame buffer following a partial data buffer
0	0	0	0	0	0	6	Partial data buffer following a partial data buffer

*Data byte refers only to the actual header data or information or mode but not the buffer descriptors or the optional flag count.

Maximum Number of Data Bytes in a Buffer

The number of data bytes in a complete frame buffer, as specified by the DATA LENGTH word, should not exceed 4095 (2^{12}-1). This does not include the optional one-byte FLAG COUNT. The number of data bytes in a partial data buffer, as specified by the BUFFER SIZE word, should not exceed 4095 (2^{12}-1).

RECEIVE DATA BUFFER/COMMAND ORGANIZATION

A general organization of data within a buffer and the linking of receive buffers is illustrated in Figure 16. The detail content of a Receive Data Buffer is shown in Figure 17. The content of a Receive Command Buffer is shown in Figure 18. Information within the buffer is organized in to two groups: descriptors and data.

The first group of bytes contains the buffer descriptors, i.e., information such as the link to the next buffer, buffer size, the number of data bytes in the buffer and buffer status. This group of information is mandatory for each buffer.

The second group contains k-bytes of information, k being a variable number. They may be the received data (including any header) after processing by R8071A (if necessary), or channel mode and data rate definition information as specified by the host processor.

RECEIVE CHANNEL DESCRIPTORS

The breakdown and the ordering of the seven bytes of descriptors is shown in in Figure 17. The first six bytes contain the next buffer address, the buffer size and the data length—each consisting of two bytes. The relative locations of the upper and the lower bytes are interchangeable by the use of the input strap pin MDFS. The seventh byte contains the status of the current receive buffer as well as the status of the receive transmit channel. A byte not used by the R8071A, and free to be used by the host, will proceed or follow the status byte as determined by MDFS (Figure 14).

Next Buffer Start Address

Bytes 0 and 1 contain the 16-bit NEXT BUFFER ADDRESS written by the host. The meaning of invalid and recirculating buffers are the same as those for the transmit buffer.

Buffer Size

Bytes 2 and 3 contain the 12-bit BUFFER SIZE, k, written by the host. The BUFFER SIZE specifies the total number of memory bytes allocated by the host for storing the data to be received. The four most significant bits are not used by the R8071A.

The R8071A reads it and stores it for all except command buffers. If the last byte of a receive HDLC frame is not received before the buffer is completely filled, the R8071A automatically searches for the availability of the next buffer specified.

Data Length

Bytes 4 and 5 contain the 12-bit DATA LENGTH field, j, written by the R8071A. DATA LENGTH specifies the actual number of received data bytes transferred to the receive data buffer by the R8071A. The four most significant bits are not used. (The R8071A clears these bits to zeroes.)

DATA LENGTH is written by the R8071A after it receives the last byte of an HDLC frame, receives the HDLC ABORT code, or upon the loss of multiframe alignment error from a non-HDLC signaling channel. DATA LENGTH is not written if the end of the allocated buffer is reached before the last byte is received (data frame length greater than buffer size). In such a case, the data length is equal to the given buffer size. Also, data length may not be written if the ATTN input is asserted, resulting in the deactivation or reactivation of an active channel. DATA LENGTH will not exceed the programmed buffer size. DATA LENGTH >2 in a command buffer is used to define hyperchannel configurations.

Receive Buffer Status

The Receive Buffer STATUS byte specifies the status of the current receive buffer as well as the status of the receive channel. The individual bits are defined as follows.

MPTY—Empty. This bit is set by the host to inform the R8071A that the buffer is empty, i.e., available for storing the received data. When the bit is reset, the buffer is not empty , i. e., not available to store the received data. The R8071A polls this bit until it is empty before it writes the received data.

This bit is reset by the R8071A whenever it updates the buffer status. This is the case even if the R8071A writes only a single byte and then is forced to update the buffer status because of abnormal conditions.

CMND—Command. This bit is set by the host to inform the R8071A that this buffer is a command buffer. The command buffer contains channel-specific mode definition and FILL/MASK information at the next two bytes following the STATUS byte. This bit is reset by the host to indicate that this buffer is a data buffer meant to store received data.

CF/$\overline{P}$—Complete Frame/Partial Data Buffer. This bit may be reset by the host at buffer initialization. This bit is set by the R8071A to indicate that this buffer contains the last byte of an HDLC frame. The R8071A automatically verifies the CRC and the starts the next HDLC frame before writing the data of the next HDLC frame in the next available buffer. In HDLC and Non-HDLC modes, if the reception of data is truncated by a resync condition asserted by an ABORT, RSYNC, TMAX or RRED, the R8071A sets the CF/$\overline{P}$ bit and writes the DATA LENGTH of the truncated buffer. In signalling Channel Mode, when two consecutive signalling synchronization errors are encountered, the R8071A will also set the CF/$\overline{P}$ bit.

This bit is reset by the R8071A to indicate that this buffer contains only a part of the received data and that more data is expected to be placed in one or more succeeding buffers. In HDLC mode, it implies that the last byte of the HDLC frame is not in this buffer. For non-HDLC data and signaling channels, this bit will be invariably reset after the buffer is filled. The R8071A does not read the CF/$\overline{P}$ bit in a command buffer. Note that the host can detect whether a received HDLC frame size exceeded the maximum anticipated buffer size by simply checking for CF/$\overline{P}$ to be set in the starting buffer of the HDLC frame.

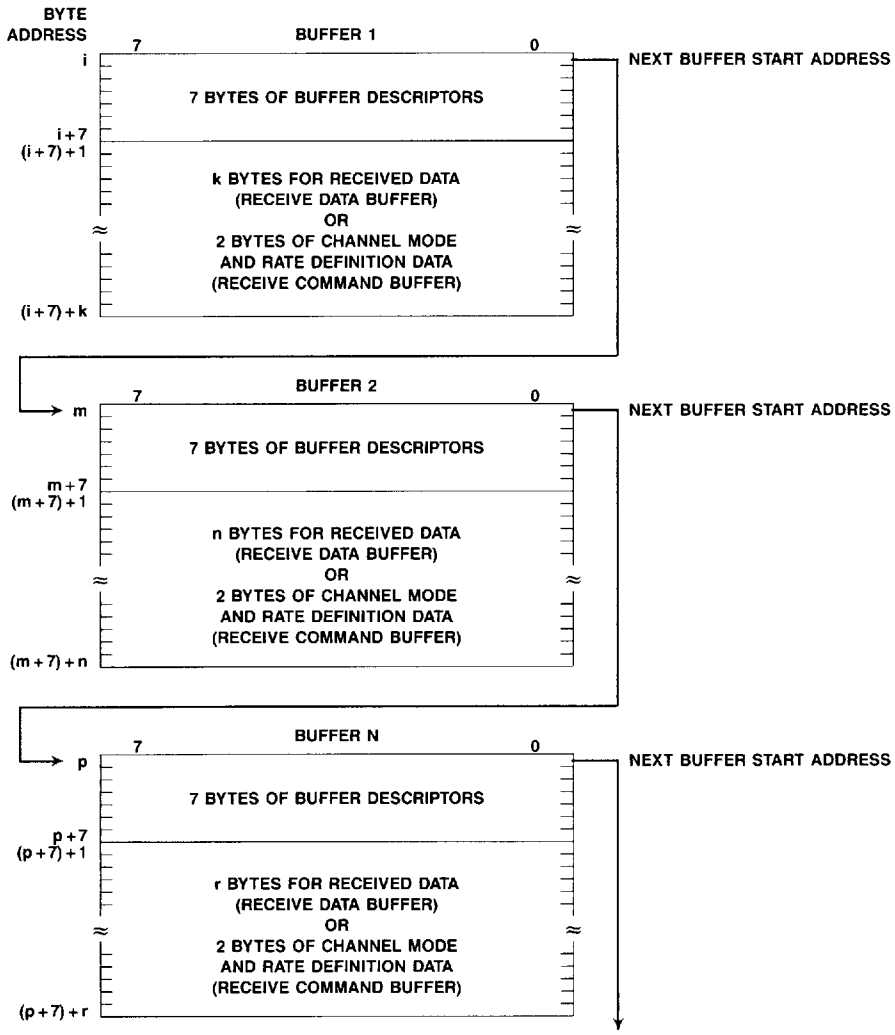


Figure 16. Organization and Linking of Receive Data Buffers

BYTE ADDRESS	CONTENTS 7	6	5	4	3	2	1	0
i	msb NEXT BUFFER ADDRESS							
i+1								lsb
i+2					msb BUFFER SIZE (k)			
i+3								lsb
i+4	X	X	X	X	msb DATA LENGTH (j)			
i+5								lsb
i+6	NOT USED BY R8071A							
i+7	STATUS OVER	IVBA	ABRT	FCER	SHER	CF/$\overline{P}$	(0) CMND	MPTY
(i+7)+1	FIRST DATA BYTE							
(i+7)+2	SECOND DATA BYTE							
⋮	⋮				⋮			
(i+7)+j	LAST DATA BYTE							
⋮	⋮				⋮			
(i+7)+k	LAST LOCATION IN BUFFER							

a. MDFS = High

BYTE ADDRESS	CONTENTS 7	6	5	4	3	2	1	0
i	NEXT BUFFER ADDRESS							lsb
i+1	msb							
i+2	BUFFER SIZE (k)							lsb
i+3	msb							
i+4	DATA LENGTH (j)							lsb
i+5	X	X	X	X	msb			
i+6	STATUS OVER	IVBA	ABRT	FCER	SHER	CF/$\overline{P}$	(0) CMND	MPTY
i+7	NOT USED BY R8071A							
(i+7)+1	FIRST DATA BYTE							
(i+7)+2	SECOND DATA BYTE							
⋮	⋮				⋮			
(i+7)+j	LAST DATA BYTE							
⋮	⋮				⋮			
(i+7)+k	LAST LOCATION IN BUFFER							

b. MDFS = Low

Figure 17. Receive Data Buffer Contents

BYTE ADDRESS	CONTENTS 7	6	5	4	3	2	1	0
i	msb NEXT BUFFER ADDRESS (i)							
i+1								lsb
i+2	NOT USED BY R8071A							
i+3								
i+4	NOT USED BY R8071A							msb
i+5	DATA LENGTH (j)							lsb
i+7	NOT USED BY R8071A							
i+6	STATUS OVER	IVBA	x	x	x	CF/$\overline{P}$	(1) CMND	MPTY
(i+7)+1	MODES X	X	X	X	INV	LOOP	SIG	HDLC
(i+7)+2	FILL/MASK							
(i+7)+3	E	A	X	CHANNEL NUMBER				
⋮	⋮			⋮				
(i+7)+2+j	E	A	X	CHANNEL NUMBER				

a. MDFS = High

BYTE ADDRESS	CONTENTS 7	6	5	4	3	2	1	0
i	NEXT BUFFER ADDRESS (i)							lsb
i+1	msb							
i+2	NOT USED BY R8071A							
i+3								
i+4	DATA LENGTH (j)							lsb
i+	NOT USED BY R8071A							msb
i+6	STATUS UNDR	IVBA	x	x	x	CF/$\overline{P}$	(1) CMND	MPTY
i+7	NOT USED BY R8071A							
(i+7)+1	MODES 0	0	0	0	INV	LOOP	SIG	HDLC
(i+7)+2	FILL/MASK							
(i+7)+3	E	A	X	CHANNEL NUMBER				
⋮	⋮			⋮				
(i+7)+2+j	E	A	X	CHANNEL NUMBER				

b. MDFS = Low

Figure 18. Receive Command Buffer Contents

ABRT—Abort. This bit is written by the R8071A and, in conjunction within the FCER and SHER bits, reports abnormal conditions detected by the R8071A (see Table 3).

Table 3. Receive Buffer Status—Error Table

ABRT	FCER	SHER	Description
0	0	0	No errors detected
0	0	1	Short or Non-integer HDLC Frame Error
0	1	0	CRC Error
0	1	1	CRC Error & Non-integer Error
1	0	0	HDLC ABORT Code Received
1	0	1	Non-HDLC Multiframe Alignment Lost
1	1	0	Elastic Buffer Error & RSYNC Error
1	1	1	RRED Alarm

FCER—Frame Check Error. This bit is written by the R8071A and, in conjunction within the ABRT and SHER bits, reports abnormal conditions detected by the R8071A (see Table 3).

SHER—Short HDLC Frame Error. This bit is written by the R8071A and, in conjunction within the ABRT and FCER bits, reports abnormal conditions detected by the R8071A (see Table 3).

IVBA—Invalid Buffer Address. This bit is set by the R8071A if it encounters an invalid next buffer address. In this case, the specific receive channel enters the idle state and will not receive more data until re-activated by the host.

OVER—Overrun. This bit is set by the R8071A after its receive channel has no next data buffer available for received data. Note that a command buffer is not available for data. The OVER bit is written as part of the status of the just completed buffer. Also note that no overrun will be reported for non-HDLC signaling channel data buffers. New data will be written in place of any earlier received signaling data.

RECEIVE COMMAND BUFFER

A receive command buffer is identical to the transmit command buffer in that it contains modes, fill/mask, and optional hyperchannel configuration data following the STATUS byte for defining the channel modes, data rate, and hyperchannel configuration.

The first byte (MODE) defines the channel modes of operation—specifically HDLC, signaling, data inversion and loop back. The second byte (FILL/MASK) defines the data rate. A data buffer contains the actual data received after processing by the R8071A.

The breakdown and the ordering of bytes within the command buffer are illustrated in Figure 18.

For a command buffer, the R8071A does not process the bytes at addresses [i + 2, i + 3] but will read the data length. The R8071A reads the next buffer address at locations i and i + 1 as part of processing the command buffer. As mentioned before, the relative locations of the upper and the lower bytes (of the next buffer address and data length) are interchangeable by means of the MDFS input. The mode and FILL/MASK bytes locations are not interchangeable by MDFS.

MODES

The MODES byte is the first byte following the status byte and specifies the operational modes of the given channel—specifically, HDLC or non-HDLC, signaling channel or not, data to be inverted bit-by-bit prior to receiver processing or not and channel receive data source to be the loop register or not.

SIG	HDLC	Mode
1	0	Non-HDLC Signaling Channel Mode
0	0	Non-HDLC Data Channel Mode
0	1	HDLC Data Channel Mode
1	1	Reserved

Non-HDLC Signaling Channel Mode. The R8071A processes the received bit-oriented signaling data without the HDLC format (G.732 or DMI). The R8071A arranges the received signaling data as in Figure 19 for easy association of the channel number and its signaling bits. In addition, errors in multiframe alignment sequence will be detected and any resulting loss of multiframe alignment will be reported in the STATUS byte.

Non-HDLC Data Channel Mode. The channel is a non-HDLC data channel that can support DMI data modes 0 or 1.

As soon as a channel receiver is activated with mode, the R8071A checks the availability of the allocated buffer and starts placing the received data in the buffer. After filling a buffer, it updates the status of the just completed buffer, simultaneously asserting $\overline{INTR}$. It then moves on to the next allocated buffer. Since non-HDLC data has no frame boundary, this process will continue forever unless the host interrupts by an ATTN or the system runs out of allocated buffers.

HDLC Data Channel Mode. The channel is to receive HDLC-formatted data. It can imply an HDLC-formatted data channel, such as the ones in DMI mode 2 or 3 or X.25 LAPB, or a message-oriented signaling channel as in LAPB. The R8071A treats each one of them the same way and deformats the data. No special handling is performed on the header, i.e., the address and control fields of the HDLC frame. The information field is assumed to be an integer number of octets or bytes. The 16-bit CRC-CCITT generator polynomial, $X^{16} + X^{12} + X^5 + 1$ is used for recomputing the FCS. Abort and flag characters are recognized as are intentionally inserted zeroes.

INV—Invert Data. When set by the host, the R8071A inverts the received data prior to processing. When INV is reset by the host, the R8071A does not invert the received data. The INV bit is applied to every received bit.

LOOP—Loop Mode. When set by the host, the R8071A selects as its input the serial output data from the internal loop data buffer, as opposed to the externally supplied serial data. If an identically numbered transmit channel was also programmed to be in the loop mode earlier, data from that channel is looped back to the shared memory through the receive channel. The loop channel number, FILL/MASK, and its mode can be specified independently for any transmit and receive channel and need not be identical. If no transmit channel has loop activated, the R8071A processes the channel as if a receiver RRED condition were active for that channel duration. Hyperchannel LOOP mode is not supported by the R8071A.

FILL/MASK

The second byte of a command buffer contains the FILL/MASK pattern. It is used as a masking pattern on the HDLC-formatted (including flag, header, data, CRC, and ABORT code) or non-HDLC data to adapt subrates that are multiples of 8 kbps to the 64 kbps rate.

BYTE ADDRESS	CONTENTS							
	7	6	5	4	3	2	1	0
i	NEXT BUFFER ADDRESS = i or j							
i+1								
i+2	REST OF THE DESCRIPTORS							
i+7								
(i+7)+1	X	1	X	X	X	B1	A1	A13
(i+7)+2	X	1	X	X	X	B2	A2	A14
(i+7)+3	X	1	X	X	X	B3	A3	A15
⋮	⋮							
(i+7)+11	X	1	X	X	X	B11	A11	A23
(i+7)+12	X	1	X	X	X	B12	A12	A1
(i+7)+13	X	1	X	X	X	B13	A13	A1
(i+7)+14	X	1	X	X	X	B14	A14	A2
(i+7)+15	X	1	X	X	X	B15	A15	A3
(i+7)+16	X	1	X	X	X	B16	A16	A4
(i+7)+17	X	1	X	X	X	B17	A17	A5
⋮	⋮							
(i+7)+23	X	1	X	X	X	B23	A23	A11
(i+7)+24	1	0	Ys	0	1	1	1	A12

a. T1 Mode

BYTE ADDRESS	CONTENTS							
	7	6	5	4	3	2	1	0
i	NEXT BUFFER ADDRESS = i or j							
i+1								
i+2	REST OF THE DESCRIPTORS							
i+7								
(i+7)+1	D17	C17	B17	A17	D1	C1	B1	A1
(i+7)+2	D18	C18	B18	A18	D2	C2	B2	A2
(i+7)+3	D19	C19	B19	A19	D3	C3	B3	A3
⋮	⋮							
(i+7)+11	D27	C27	B27	A27	D11	C11	B11	A11
(i+7)+12	D28	C28	B28	A28	D12	C12	B12	A12
(i+7)+13	D29	C29	B29	A29	D13	C13	B13	A13
(i+7)+14	D30	C30	B30	A30	D14	C14	B14	A14
(i+7)+15	D31	C31	B31	A31	D15	C15	B15	A15
(i+7)+16	1	1	Ys	1	0	0	0	0

b. CEPT PCM 30 Mode

Figure 19. Receive Buffer Data Arrangement for Non-HDLC Bit-Oriented Signaling Channel

Table 4 shows the data rates of the form n × 8 kbps (n = 1, 2, . . . , 8) and several examples of codes for the FILL/MASK to adapt the subrates to the 64 kbps rate. The actual data bit is transmitted on TSER output as long as the FILL/MASK is a 1 at the corresponding bit position. If the FILL/MASK = 0, a FILL bit of 1 is transmitted in place of the data bit on TSER as long as TSEREN = 1. The data bit that is held in favor of the FILL bit is buffered internally until all the FILL bits have been transmitted corresponding to the FILL/MASK bits equalling 0.

When the FILL/MASK bit becomes a 1 again, the buffered data bit is transmitted on a first in-first out basis (Figure 7.)

Thus, there is a one-to-one correspondence between the FILL/MASK bit and the T1 or CEPT PCM 30 channel serial data associated with the FILL/MASK bit position.

When the FILL/MASK bit is 0, TSEREN determines what is to be transmitted on TSER:

Data Bit	FILL/MASK Bit	TSEREN	TSER (Output)
1	1	X	1
0	1	X	0
X	0	1	1
X	0	0	High Z

The data bit may be the same as, or logical complement (inverted version), of the actual data.

A bit-oriented 64 kbps signaling channel should be programmed with 1111 1111 as the FILL/MASK. The R8071A will not override any other user-supplied FILL/MASK pattern even if it is erroneous, i.e., not equal to 1111 1111. An IDLE bit-oriented signaling channel will carry 1111 1111 as long as the channel is IDLE. Any remote receiver will not be able to achieve signaling channel multiframe alignment if 1111 1111 is received continuously.

FLEXIBLE HYPERCHANNELS

Flexible Hyperchannels are created by grouping any number of channels into a hyperchannel. These channels may be non-contiguous. Any number of hyperchannels may operate along with any number of channels. A channel may be assigned to only one hyperchannel. Any Receive channel may be grouped into a particular hyperchannel by configuring the hyperchannel's Receive Command buffer. (Figure 18) DATA LENGTH in the command buffer determines Flexible Hyperchannel or normal channel assignment, i.e.:

Command Buffer DATA LENGTH = 0,1,2:normal channel process
Command Buffer DATA LENGTH > 2 :hyperchannel process

In Flexible Hyperchannel mode, the least significant byte of DATA LENGTH is read to determine the number of additional channels to be assigned to the hyperchannel. The data bytes containing the additional channel numbers are read from the command buffer

Table 4. Examples of FILL/MASK Options

Option No.	Data Rate	Bit 7 (MSB)	6	5	4	3	2	1	0 (LSB)	Remarks
0*	0 Kbps	0	0	0	0	0	0	0	0	No data will be sent. A time FILL of eight 1s will be sent provided TSEREN = 1.
1	8 Kbps	0	0	0	0	0	0	0	1	
		0	0	1	0	0	0	0	0	Arbitrary-user defined. A 1 in any one bit position but only one 1.
2	16 Kbps	0	0	0	0	0	0	1	1	
		0	0	0	0	1	1	0	0	User defined patterns; a 1 in any 2 bit positions but only two 1s.
		0	0	1	1	0	0	0	0	
		1	1	0	0	0	0	0	0	
3	24 Kbps	0	0	0	0	0	1	1	1	A total of three 1s anywhere as defined by user.
		0	1	0	1	0	1	0	0	
4	32 Kbps	0	0	0	0	1	1	1	1	A total of four 1s anywhere as defined by user.
		1	1	1	1	0	0	0	0	
		1	0	1	0	1	0	1	0	
5	40 Kbps	0	0	0	1	1	1	1	1	A total of five 1s anywhere as defined by user.
6	48 Kbps	0	0	1	1	1	1	1	1	A total of six 1s anywhere as defined by user.
7	56 Kbps	0	1	1	1	1	1	1	1	Standard rate in Digital Data Service, restricted version of 64 Kbps.
		1	1	1	1	1	1	1	0	A total of seven 1s anywhere as defined by user.
8	64 Kbps	1	1	1	1	1	1	1	1	

*Special purpose mode, transmitter operates as if at 64 Kbps including fetching data from shared memory, even though no data is transmitted.

in sequence and passed to the appropriate Receiver. Previously active channels may be appropriated into a Flexible Hyperchannel. When they are appropriated, buffer activity on the original channel ceases; when released from the hyperchannel, the original channel will return to the state from which its activity was suspended. The channel map may be updated within the time of a single pass through the channel counter. The number of the channel that activates the Flexible Hyperchannel becomes the number of the hyperchannel. Each data byte in the command buffer is structured as shown in Figure 18.

E	A	
0	X	Hyperchannel assignment remains unchanged
1	0	Delete channel number in bits 0–4 from hyperchannel
1	1	Add channel number in bits 0–4 to hyperchannel

Whenever a hyperchannel is deactivated, it is automatically cleared to the default channel value. This takes one frame time minimum. The R8071A supports independent transmit and receive hyperchannel assignments. When using Flexible Hyperchannel, the HCS0 and HCS1 bits should both be reset to zero. Otherwise, the standard hyperchannel defined by these two pins supercedes Flexible Hyperchannel.

RECEIVE CONSIDERATIONS

Minimum Buffer Size

There is a minimum number of memory locations that have to be allocated in each buffer for the R8071A to perform the buffer maintenance and still effect a smooth transition to the next buffer without losing data. The minimum buffer allocations for data must allow six data bytes in addition to the seven bytes of descriptors. Command buffers have exactly two bytes and are processed without regard for the BUFFER SIZE word.

Note: The minimum HDLC frame received can have as few as two bytes of data and the R8071A will still function properly; however, it is essential that a buffer size of six must be allocated as a minimum since the received frame size is not known apriori.

SHARED MEMORY ACCESS

The R8071A accesses shared memory for buffer maintenance, command data, information and also for channel activation. It manages the buffer memory for up to 64 channels (32 transmit and 32 receive).

The T1 or CEPT PCM 30 data throughput requirements demand that one octet of data be supplied to the transmitter and one octet of data be taken from the receiver in a single channel period (8 TCLK periods). The host and the R8071A must work cooperatively to meet the data throughput requirements. The R8071A uses a memory access scheme that simplifies the system design in achieving the required data throughput.

The R8071A processes the memory requirements of up to 32 channels in the same order in which they are multiplexed in a T1 or CEPT PCM 30 carrier system. Typically, during TX channel m (for example), the R8071A fetches a single data byte from memory for the transmitter so that it can transmit it over channel m at the next appropriate T1 or CEPT PCM 30 frame. Similarly, the last data byte received by channel j is written to the memory by the R8071A during the next appropriate receive channel j. Then it services TX channel (m + 1) and RX channel (j + 1) and so on.

The R8071A divides a channel period in to two halves, each for a duration of 4 TCLK periods. During the first half-channel period, it accesses the shared memory for channel command information, buffer descriptors, and/or transmit buffer data (including mode definition data) for a transmit channel. During the second half-channel period, it accesses the shared memory for channel command information, buffer descriptors (including mode definition data), and/or received data for a receive channel. Since the transmit and the receive channel boundaries are generally unrelated, an elastic buffer is used to synchronize the receive channel boundary to that of the transmit channel. Hence, no contention exists between a transmit channel and receive channel for the shared memory.

In each half-channel period, under normal circumstances, the R8071A accesses shared memory once for data. If descriptor information is also to be updated, it accesses shared memory a second time. Additionally, if system memory access is also required (as determined by assertion of the ATTN input), it accesses shared memory a third time. In summary, different states of buffer processing will cause anywhere from zero to three accesses to shared memory for an active channel during a half-channel period.

At the start of every half-channel period, the R8071A outputs the binary code for the 5-bit channel number (CH0–CH4) being served. It also specifies whether it is the receive or transmit channel via the RX/$\overline{TX}$ output. About one-half TCLK period later, the R8071A asserts the Memory Demand (DMND) output. DMND rising edge informs any external shared memory arbitration logic that the R8071A needs unconditional access to the shared memory within one TCLK period from DMND rising edge. Note that the R8071A will not wait for a memory acknowledge to start memory access. Thus there is an implied memory acknowledge after one TCLK. See Figure 20 for timing. Prior to asserting $\overline{READ}$ or $\overline{WRITE}$ strobes, the R8071A asserts Memory Address Strobe ($\overline{AS}$).

At $\overline{AS}$ falling edge, the memory address on the A0–A15 lines is valid. Simultaneously, when the memory address changes, the output SYSACC is asserted provided the system memory is being addressed. Moreover, the R8071A will selectively tri-state the high order memory address lines (A8–A15) during the system memory accesses when specified so by UAEN.

Following $\overline{AS}$, the R8071A asserts the $\overline{READ}$ or $\overline{WRITE}$ output strobe for read or write operation, respectively. The data on the data bus (D0–D7) is latched by R8071A just prior to the rising edge of $\overline{READ}$ during a read operation. Data placed on the data bus is written to the memory during the period that $\overline{WRITE}$ is low.

Address setup time, address hold time, data setup time, and data hold time are specified such that a wide variety of off-the-shelf RAM devices may be used. The $\overline{READ}$ output from R8071A may be used as an Output Enable ($\overline{OE}$) input to the RAM devices. Since the R8071A uses its SYSCLK input to generate the various strobes for memory access, the access time requirements are automatically scaled depending on the T1 or CEPT PCM 30 application.

Once the R8071A makes the first memory access, it assumes that continued access to the memory is guaranteed as long as DMND is active. It no longer waits for one TCLK period before the actual memory access. At the most, there may be two more memory accesses. Such a case is illustrated in Figure 20. After completing the needed memory accesses, the R8071A negates the DMND output indicating that it no longer needs access to memory. It also negates the SYSACC output as long as it does not access system memory locations.

The minimum one TCLK latency (two TCLK periods with ATTN inactivity) between DMND and the actual memory access is considered to be sufficient for an external arbitration logic to release the memory bus to the R8071A. Failure to do so may cause loss of data and unpredictable operation. The time between DMND going low and the start of the first memory access by R8071A is considered to be sufficiently long either for a single complete memory bus cycle by the host or for the completion of a pending host memory bus cycle. Since the R8071A does not wait for a memory acknowledgement, DMA-like operation using a single external shared address and a data bus cannot be guaranteed.

Examples of the R8071A to shared memory interface waveforms are shown in Figure 20.

MEMORY ADDRESS EXTENSION

The 16-bit memory address output by the R8071A may be extended to more than 16 bits by the use of the channel number (CH0–CH4) and RX/$\overline{TX}$ bits. These six bits may be used directly as higher order address bits for a 22-bit address, or they can be mapped by an external look-up table to another set of n bits (where n is specified by the host). Since the channel number and RX/$\overline{TX}$ are output by the R8071A well in advance of the 16-bit address, address translation time is not of any concern.

Address selection for the system memory locations can be achieved by the host by using the SYSACC output and the UAEN input. External hardware can jam any address on the upper eight bits of the R8071A memory address, i.e., A8–A15, since the R8071A tri-states them (if UAEN is active during SYSACC.)

MEMORY ADDRESS RESTRICTIONS

System Memory Address

The R8071A checks the start address of the first buffer of any channel for an invalid address. It does the above check immediately after it reads the 2-byte start address from the system memory, as part of servicing the ATTN interrupt from the host. If an invalid start address is detected for any channel, that channel is forced inactive automatically.

Data Buffer Memory Addresses

The R8071A checks the next buffer address for an invalid address for all the channels. If it is an invalid address, it forces the corresponding channel to an inactive state and also sets the IVBA status bit. Recovery from an idle state to an active state is possible only if the host system asserts the ATTN input to the R8071A. In host systems using more than 16 bits for shared memory address, an invalid address, as interpreted by the R8071A, refers to all addresses divisible exactly by 65,536 or an address of the form $n \times 2^{16} \times (1111\ 1111\ 1111\ XXXX)$, where n is a power of 2. Within each 64 kbyte address block, only addresses 0001–FFEF (hex) are valid.

The R8071A BMM internal adder calculates the absolute memory address from the given buffer start address and any offset needed to locate either the bookkeeping information or the data byte. The maximum address within a buffer for a given channel is the address of the last byte of the buffer. Since it is always represented by a 16-bit binary number, it is restricted to 65,535 (decimal). In other words, it is reduced to modulo 65,536. Hence, the following bound,

16-bit address of the last byte = 16-bit buffer start address
+ 610 (for bookkeeping)
− 12-bit data length or buffer size

≤ 65,535 (decimal) or FFFF (hexadecimal)

should be strictly adhered to when programming the **buffer start address** and the **Data Length/Buffer Size**; otherwise, the R8071A will access memory locations not intended for that channel.

It is to be emphasized that the R8071A **does check** for buffer start addresses in the range FFF0 through FFFF and declares them as invalid addresses. For systems using more than 65,535 byte addresses, all shared memory addresses must be within one 64 kbyte page or bank.

INTERRUPT INDICATION

The R8071A asserts the Interrupt Indication ($\overline{INTR}$) output anytime the status of any buffer is updated (written) by the R8071A. See Figure 20 for timing illustration. The active period of $\overline{INTR}$ is one-half TCLK period. At the rising edge of $\overline{INTR}$, the channel number (including RX/$\overline{TX}$) and its current buffer status placed on the data bus are guaranteed to be valid so that they can be captured in an external FIFO queue. The R8071A does not queue the interrupts and their causes internally nor does it wait for an interrupt acknowledge from the host before removing the interrupting channel number and its buffer status. The R8071A processes a channel only for a half-channel period and then moves on to the next channel.

In addition to capturing the channel number and buffer status, external hardware can also capture the actual memory address of the status byte in another FIFO queue. By reading such a queue, the host system can reallocate the completed buffers in any way it sees fit and also cross-check against its own list of linked buffer addresses. If all the buffer start addresses are divisible exactly by eight, they can be derived from the STATUS byte addresses in the FIFO queue by simply setting the three LSB addresses to zero.

DEVICE INITIALIZATION

Upon reset, all the transmit channels are forced to the inactive state. All the transmit channels are initialized to the HDLC INV, NON-SIG data mode with a FILL/ MASK byte of eight 0s. No data is transferred from memory. All the receive channels are forced to the idle state. They are initialized to the HDLC data mode with a FILL/MASK byte of eight 0s. No data is written to the shared memory. The input strap pins define the TDM format, i.e., T1 or CEPT PCM 30, and also the hyperchannel grouping. The modes bits are assumed to be (INV = 1, LOOP = 0, SIG = 0) in addition to HDLC = 1 for each channel. Flexible Hyperchannels are disabled.

The Transmit Multiframe Sync (TMAX) pulse is assumed to be valid for the purpose of generating an internal channel number. The Receive Multiframe Sync (RSYNC) pulse and the RRED input are monitored by the R8071A to ascertain the receiver framing synchronization.

CHANNEL INITIALIZATION

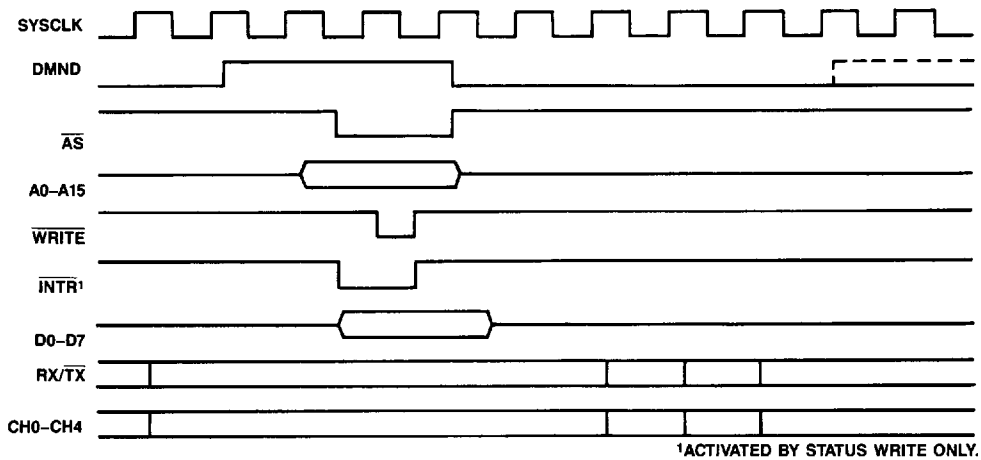
The host may activate any transmit or receive channel to any mode, by pointing it to a command buffer. It does so in a simple and systematic manner as illustrated in Figure 21.

It chooses a starting address for a command buffer and writes the 2-byte starting address as the data at the system memory location dedicated to the channel to be initialized. It then prepares a command buffer at the above starting address by specifying the descriptor information and the mode of operation. A linked list of data buffers is set up by the host following the command buffer. If Flexible Hyperchannels are to be used then the primary channel buffers will require additional memory to be allocated due to the high data rate. This completes the preparation for activating a channel.

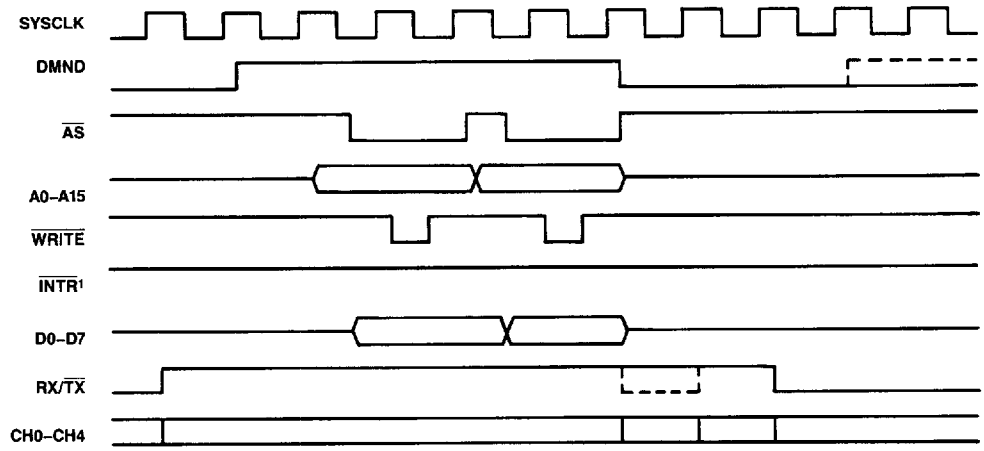
As the last step, the host writes to Channel Activation Byte containing the the channel number, channel direction and the activation command then asserts the ATTN input to the R8071A.

When ATTN is asserted, the R8071A first reads the Channel Activation Byte. Based on the channel number, it then reads the starting address of the first buffer from the Channel Buffers Pointers, one byte at a time. It stores the starting buffer address internally and acknowledges the task completion by asserting ATACK. The host system must respond to ATACK by negating ATTN.

The negation of ATTN causes ATACK output to be negated. Thus the channel initialization process is complete. This process can be repeated for each channel that needs to be initialized. During each System Memory ccess the R8071A asserts the SYSACC output. It needs to make three System Memory accesses to complete the channel ATTN processing. The worst case time delay from ATTN assertion to ATACK assertion is three T1 or CEPT PCM 30 channel periods. The earliest is $1^1/_2$ channel periods. Since this is guaranteed, the host need not poll the ATACK output blindly nor service the ATACK as an interrupt. Sample channel initialization sequences are shown in Figures 22 and 23.

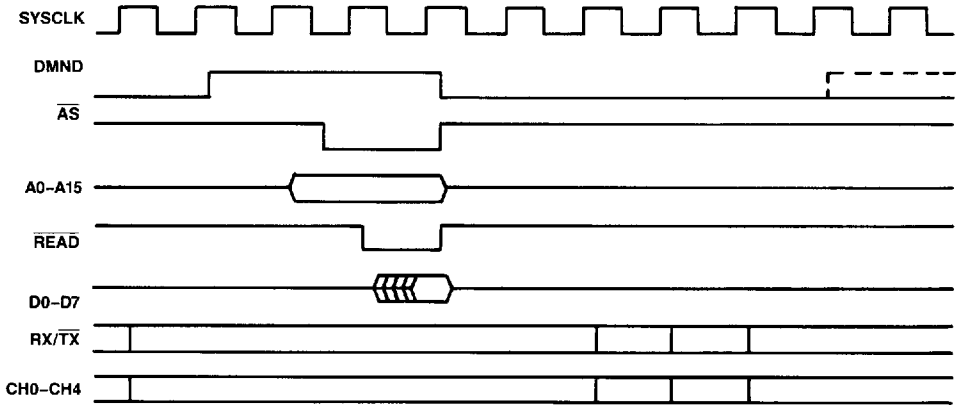


a. Single Write Memory Access

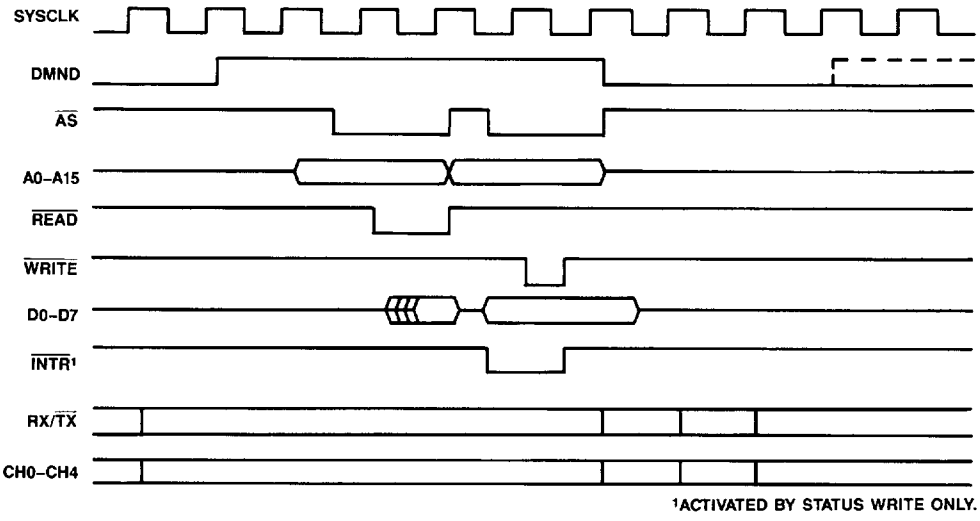


b. Double Write Memory Access

Figure 20. R8071A Shared Memory Example Interface Waveforms

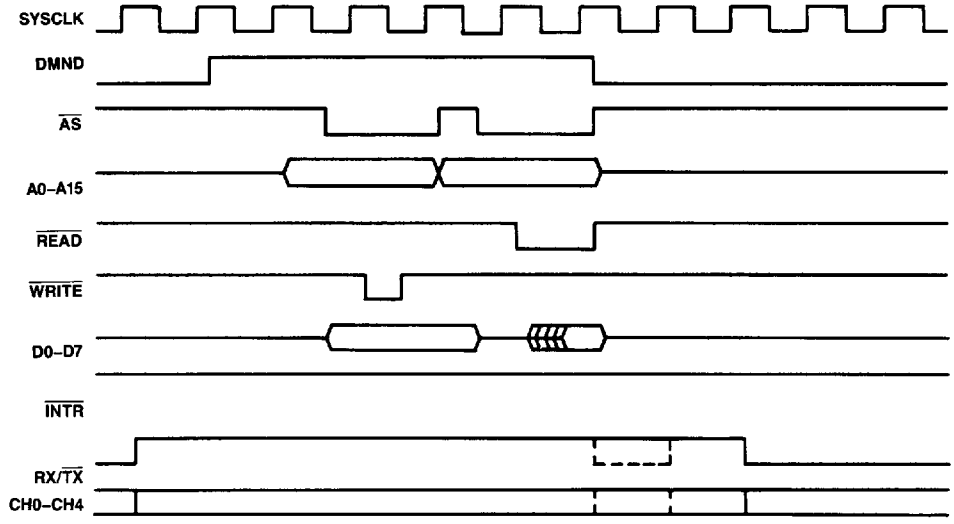


c. Single Read Memory Access

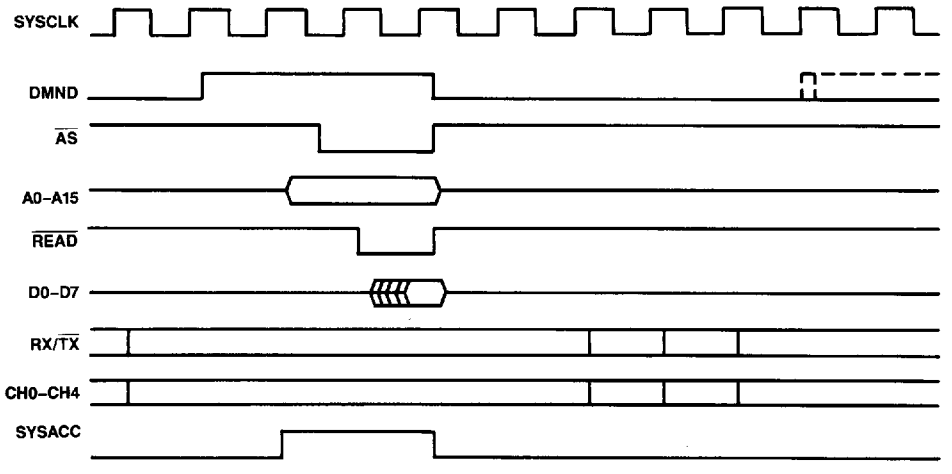


d. Read/Write Double Memory Access

Figure 20. R8071A Shared Memory Example Interface Waveforms (Continued)

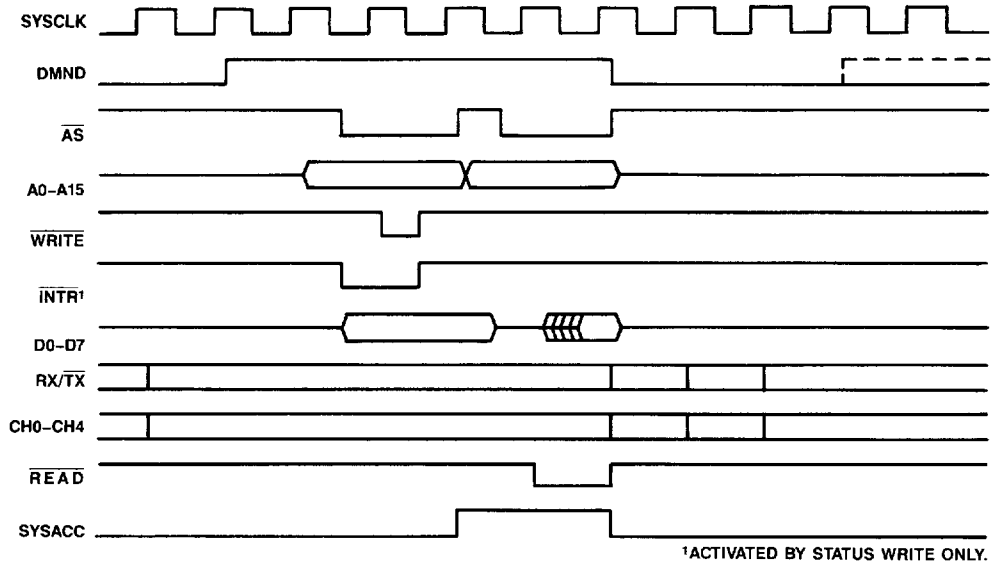


e. Write/Read Double Memory Access

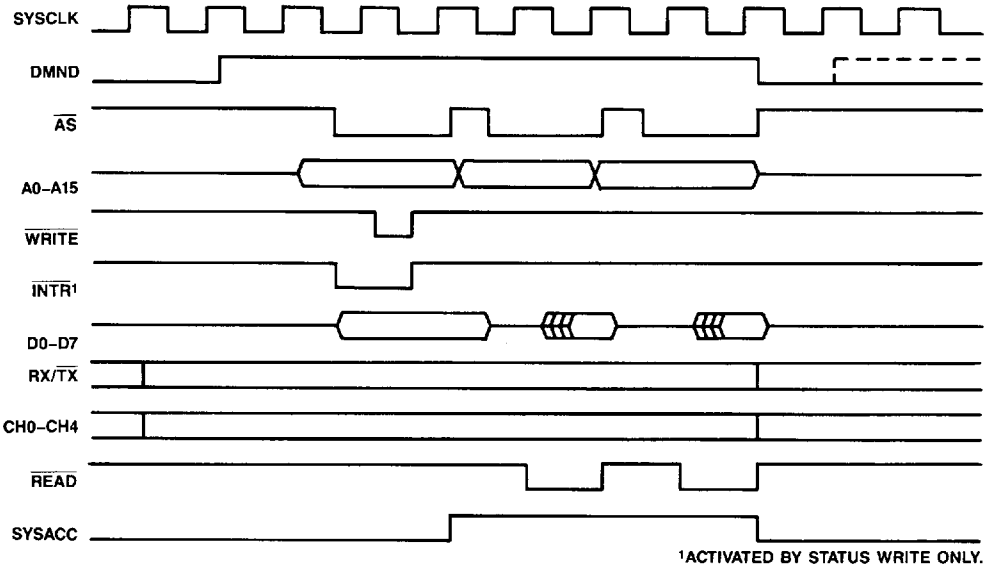


f. Single System Read Memory Access

Figure 20. R8071A Shared Memory Example Interface Waveforms (Continued)

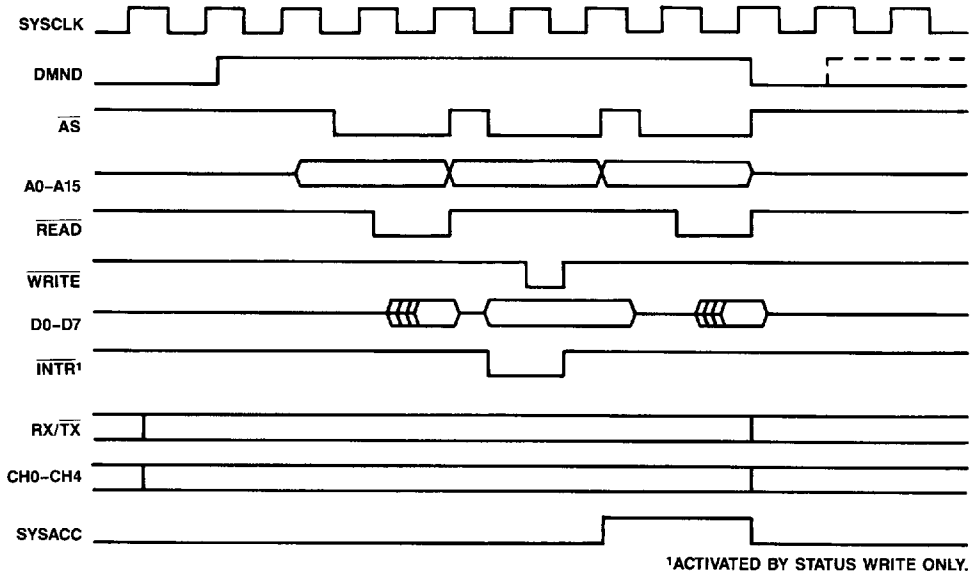


g. Single Write Memory Access Plus a Single System Read Access



h. Single Write Memory Access Plus a Double System Read Access

Figure 20. R8071A Shared Memory Example Interface Waveforms (Continued)



i. Write/Read Double Memory Access Plus Single System Read Access

Figure 20. R8071A Shared Memory Example Interface Waveforms (Continued)

ORDER OF DATA

Transmission

The R8071A transmits data bytes in the same time sequence as they are arranged in ascending addresses in the external buffers. The data at byte address m is transmitted first, the one at address m + 1 is transmitted next, and so on as long as the data bytes are in the same buffer. After the data in a single buffer is exhausted, the R8071A starts to transmit the next byte from the next buffer whose address is specified in the current buffer. The transition to the next buffer is transparent to the host while maintaining the flow of actual data.

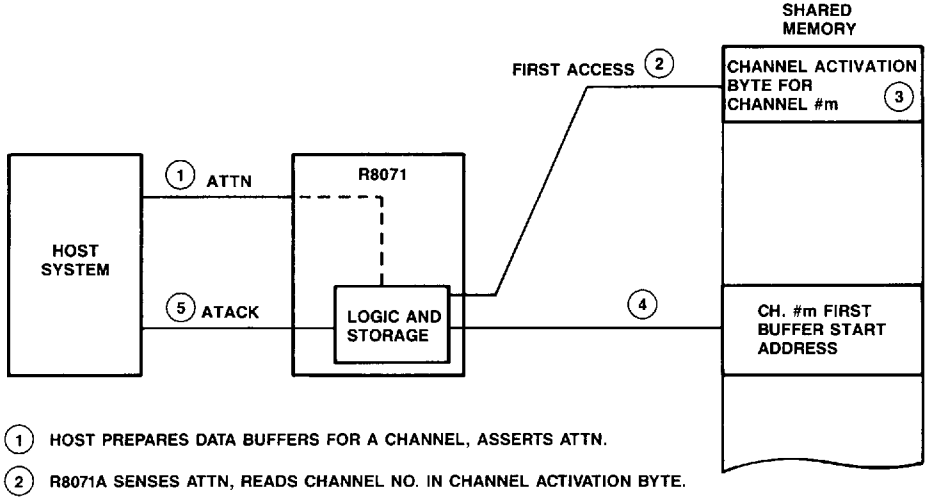
This "natural" sequence of data flow is maintained for Flexible Hyperchannels as well; it is not possible to transmit hyperchannel data in any other sequence than first in, first out.

The R8071A transmits the LSB (D0) of a data byte first; the next LSB is transmitted second; the MSB (D7) is transmitted last. The only exception is that the MSB of the HDLC FCS (CRC-CCITT) is transmitted first; the LSB is transmitted last.

Reception

The R8071A writes received data bytes in the external shared memory in the same order in which they are received in time. The first received byte is written at byte m, the second received at byte address m + 1, and so on as long as the buffer is not completely filled or an end-of-frame is not reached. After the end of the frame or the end of the buffer (whichever occurs first) is detected, the R8071A writes the next received data byte at the first allocated address of the next available buffer. The transition to the next buffer is transparent to the host and maintains the flow of the actual data.

The R8071A writes the first received data bit of an octet at the LSB (D0) position of the external buffer byte; the second received data bit at the next to LSB position and so on. The last (eighth) received data bit of an octet is written at the MSB (D7) position of the data byte.



1. HOST PREPARES DATA BUFFERS FOR A CHANNEL, ASSERTS ATTN.
2. R8071A SENSES ATTN, READS CHANNEL NO. IN CHANNEL ACTIVATION BYTE.
3. R8071A ADDRESSES THE MEMORY LOCATION DEDICATED FOR THE SPECIFIC CHANNEL.
4. R8071A READS AND STORES THE START ADDRESS OF THE FIRST BUFFER ALLOCATED FOR THE CHANNEL.
5. R8071A ACKNOWLEDGES TASK COMPLETION BY ASSERTING ATACK.

Figure 21. Channel Initialization

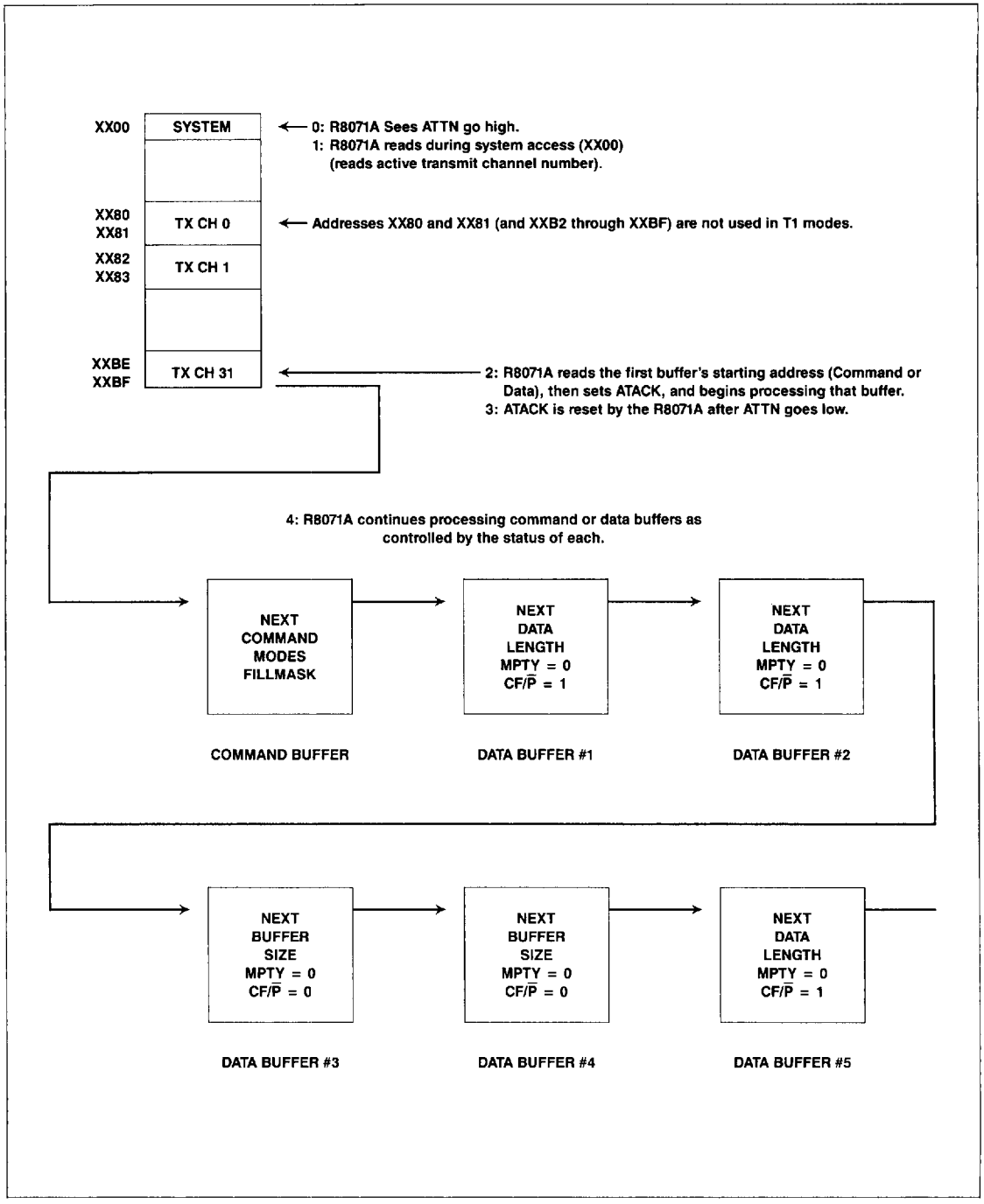


Figure 22. A Typical Linked Buffer Transmit Sequence

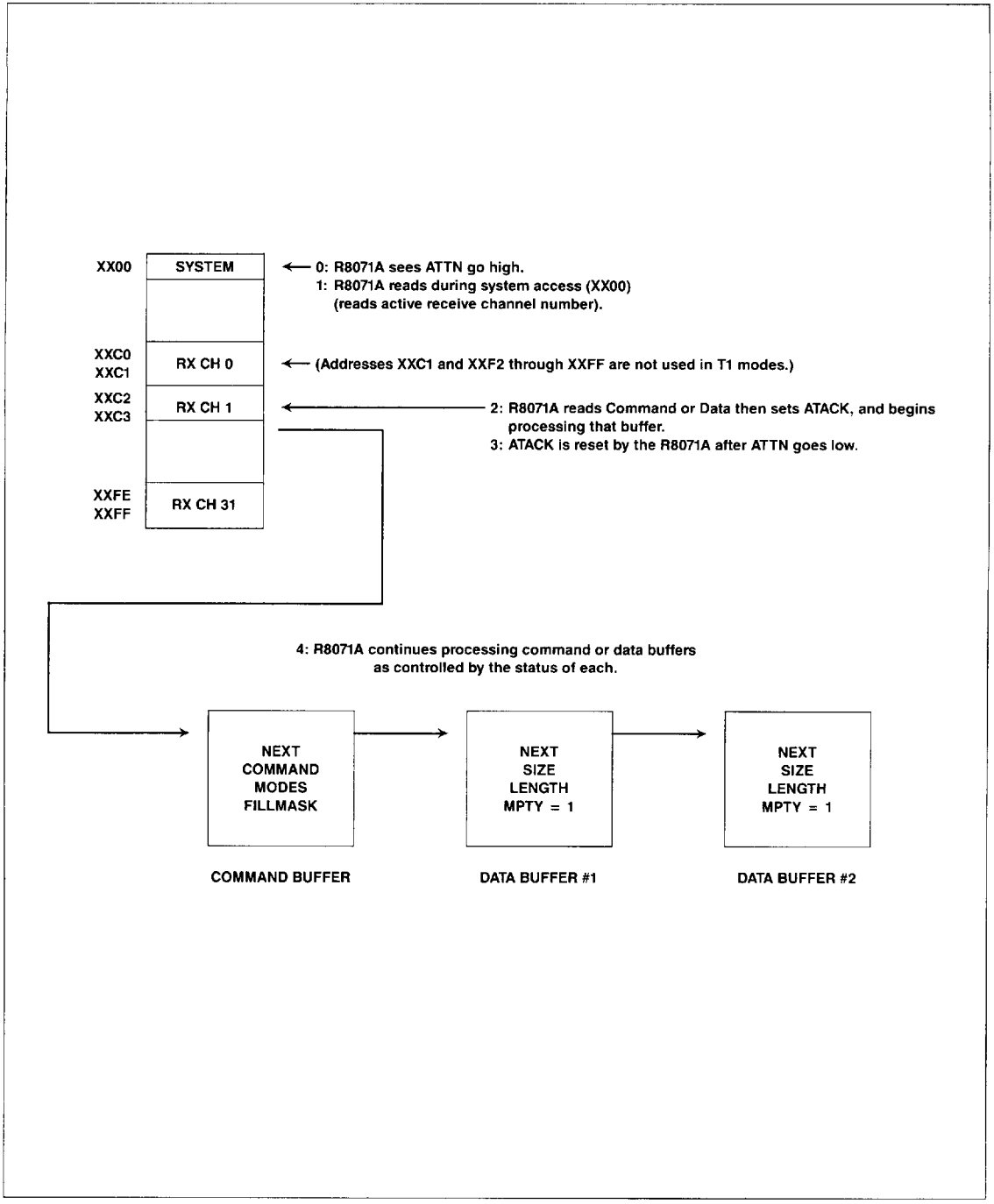
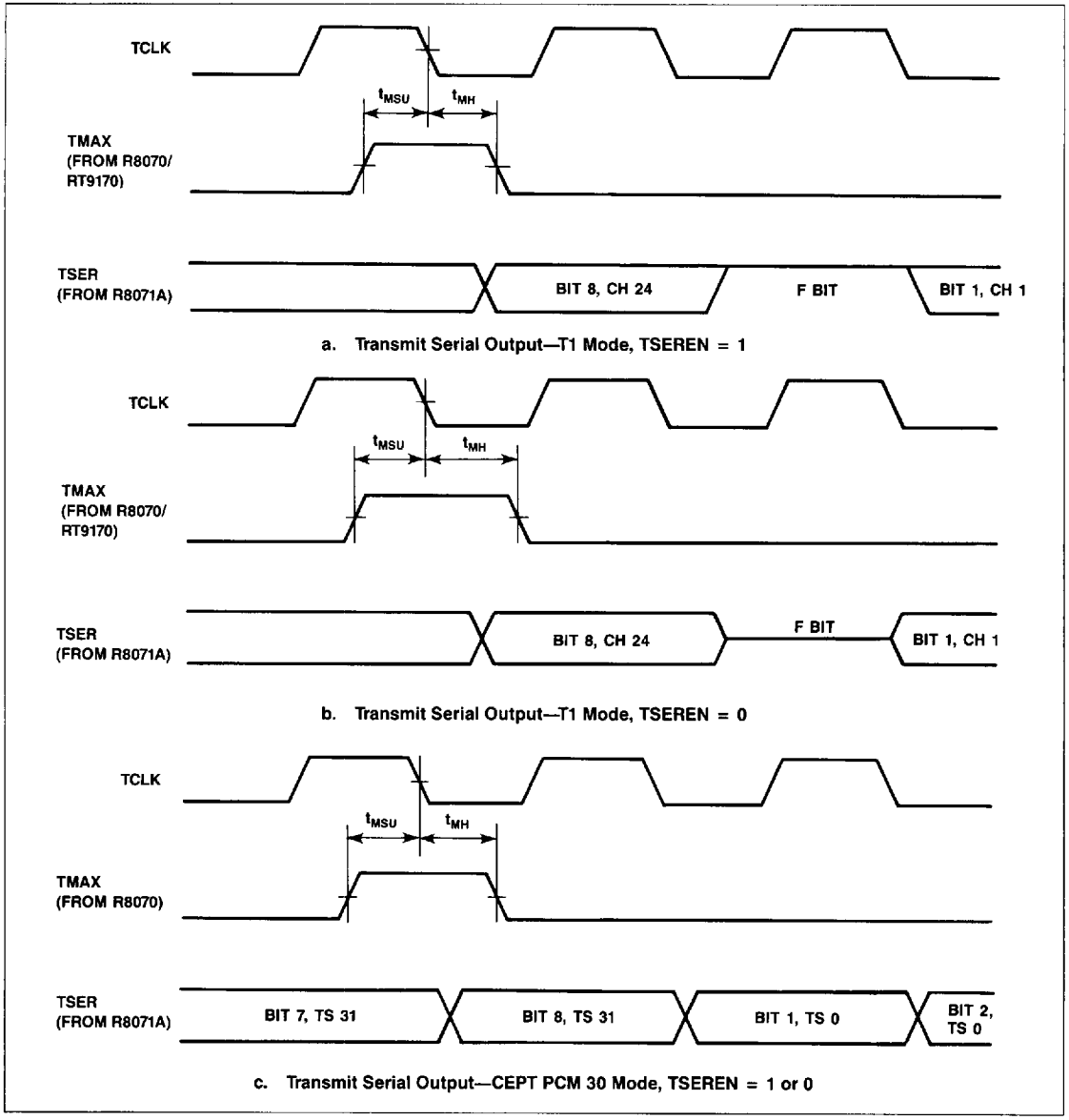


Figure 23. A Typical Linked Buffer Receiver Activity

SWITCHING CHARACTERISTICS

R8070/RT9170 INTERFACE—R8071A TRANSMIT FRAME SYNCHRONIZATION TIMING



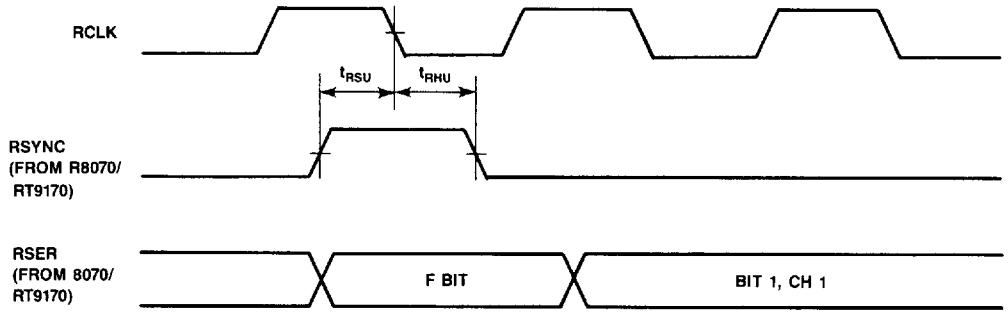
R8071A Transmit Frame Snchronization Waveforms

R8071A Transmit Frame Synchronization Timing

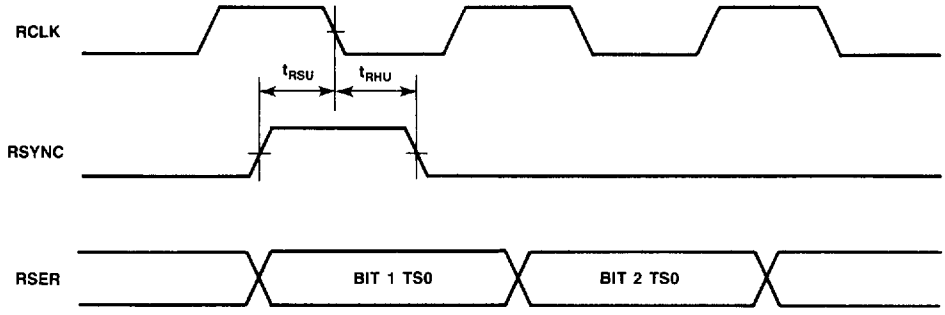
Symbol	Parameter	Min.	Max.	Units
t_{MSU}	TMAX Setup time	60	—	ns
t_{MH}	TMAX Hold time	60	—	ns

SWITCHING CHARACTERISTICS (Cont'd.)

R8070/RT9170 INTERFACE (Cont'd.)



a. Receiver Serial Input—T1 Mode



b. Receiver Serial Input—CEPT PCM 30 Mode

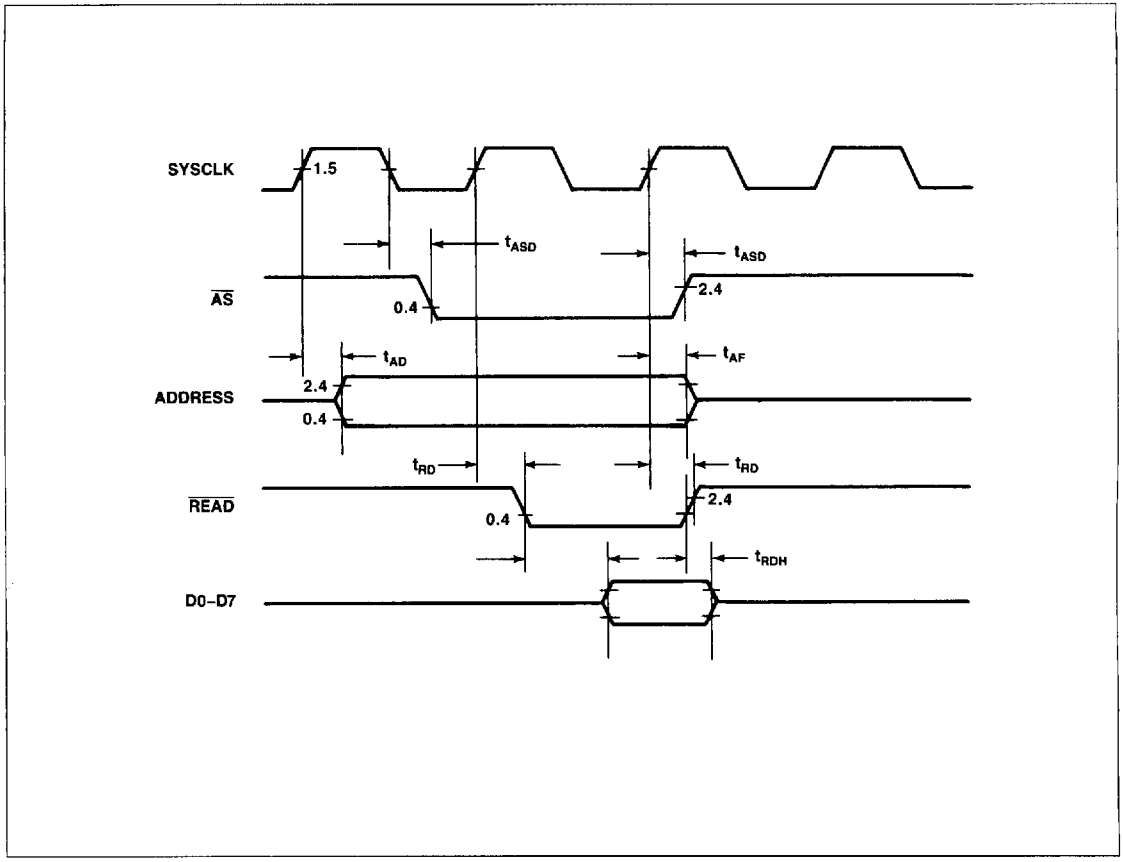
R8071A Receive Frame Synchronization Waveforms

R8071A Rescue Frame Synchronization Timing

Symbol	Parameter	Min.	Max.	Units
t_{RSU}	RSYNC Setup time	50	—	ns
t_{RHU}	RSYNC Hold time	50	—	ns

SWITCHING CHARACTERISTICS (Cont'd.)

SHARED MEMORY INTERFACE



Read Cycle Waveforms

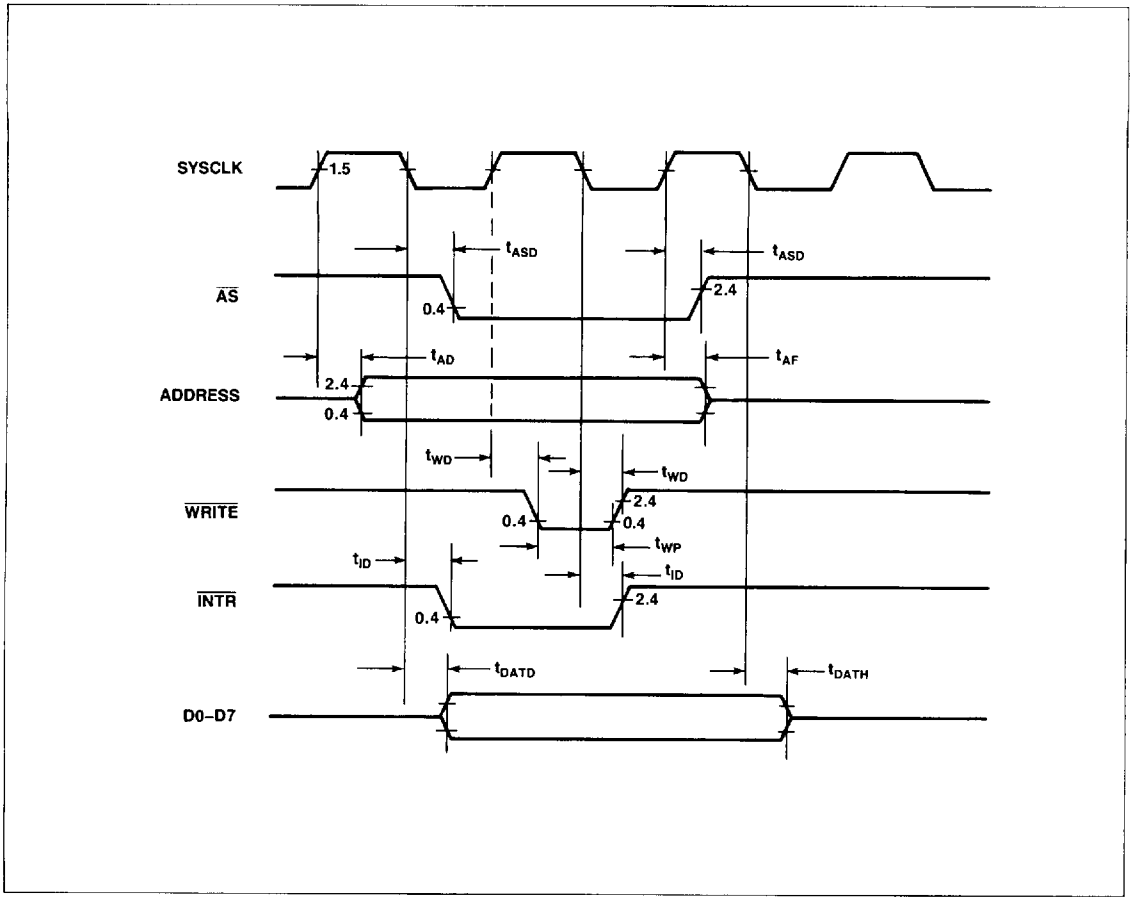
Read Cycle Timing

Parameter	Symbol	Min.	Max.	Units
Address Strobe Delay	t_{ASD}	10	75	ns
Address Delay	t_{AD}	10	90	ns
Address Float Delay	t_{AF}	10	90	ns
Read Enable Delay	t_{RD}	10	75	ns
Read Data Access Time	t_{RDA}	—	Note 1	ns
Read Data Hold Time	t_{RDH}	0	Note 2	ns

Notes:
1. Read Data Access time for shared memory = t_{SCP} – 125 ns.
2. Data Drive to Data Bus Float = t_{SCPW} – 65 ns.

SWITCHING CHARACTERISTICS (Cont'd.)

SHARED MEMORY INTERFACE (Cont'd.)



Write Cycle Waveforms

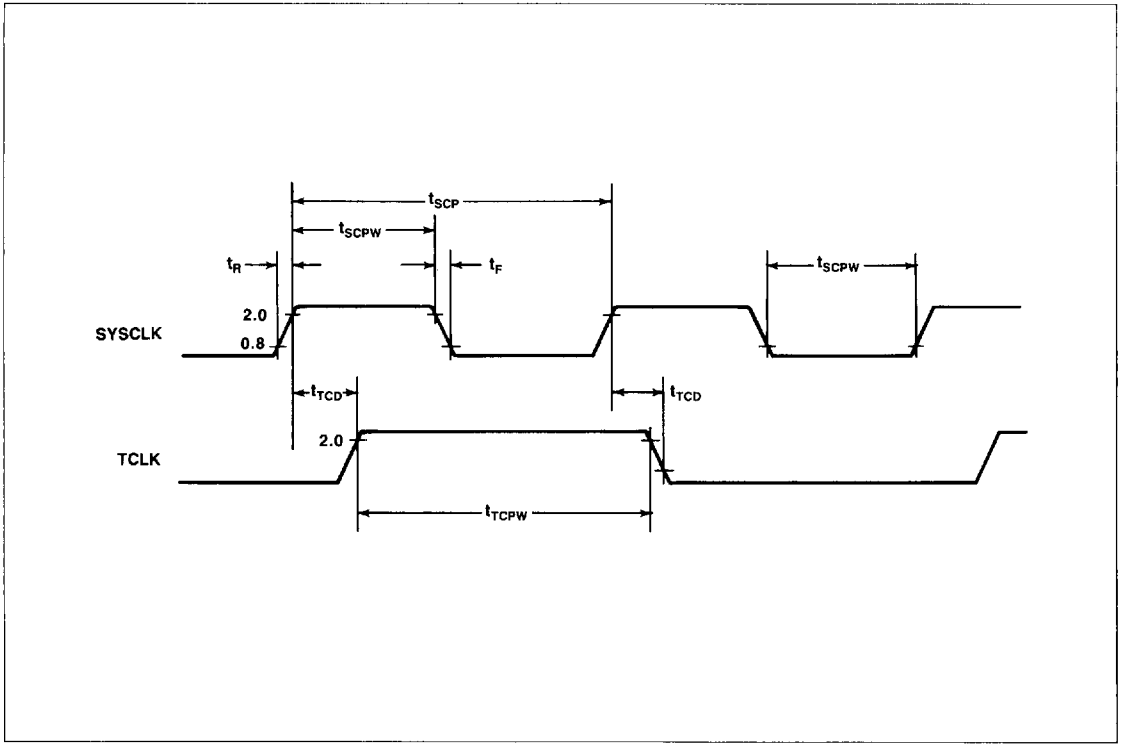
Write Cycle Timing

Parameter	Symbol	Min.	Max.	Units
Address Strobe Delay	t_{ASD}	10	75	ns
Address Delay	t_{AD}	10	90	ns
Address Float Delay	t_{AF}	10	90	ns
Write Delay	t_{WD}	10	75	ns
Write Pulse Width	t_{WP}	80	—	ns
Interrupt Delay	t_{ID}	10	75	ns
Write Data Delay	t_{DATD}	10	90	ns
Write Data Hold Time[1]	t_{DATH}	10	90	ns

Note:
1. Data Drive to Data Bus Float time

SWITCHING CHARACTERISTICS (Cont'd.)

LIU INTERFACE



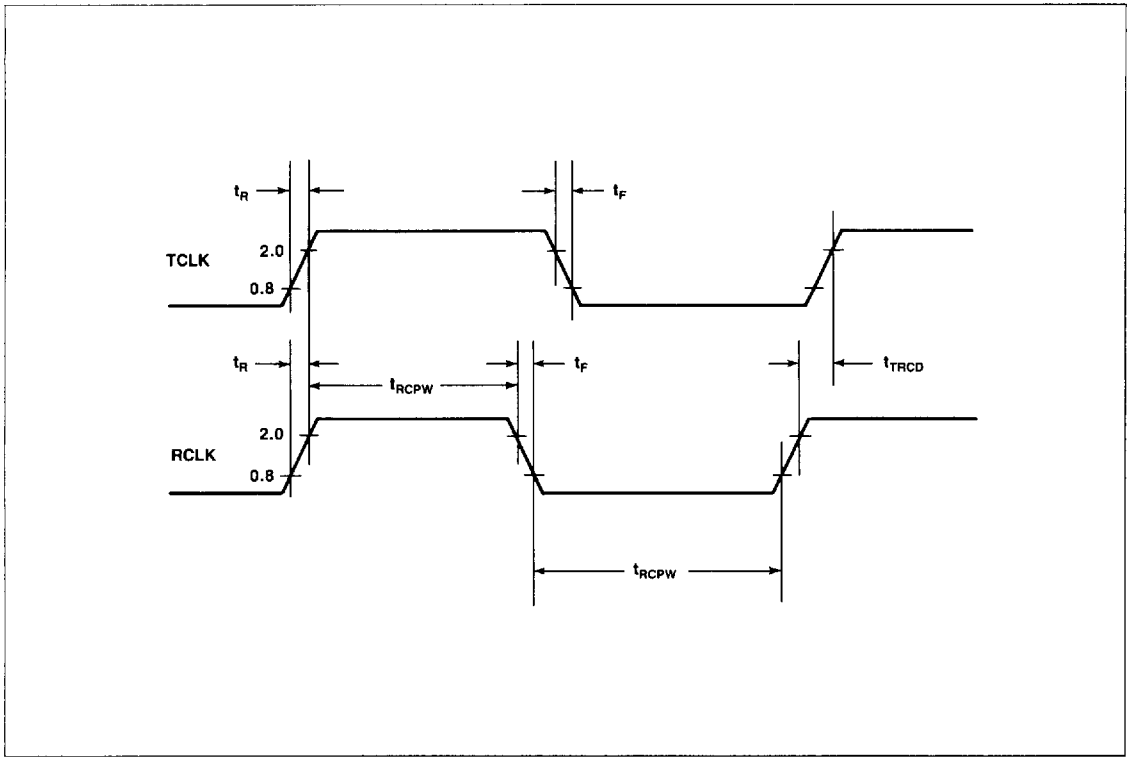
R8071A Clock Waveforms

R8071A Clock Timing

Parameter	Symbol	Min.	Max.	Units
TCLK Delay	t_{TCD}	0	50	ns
SYSCLK Pulse Width	t_{SCPW}	110	—	ns
TCLK Pulse Width	t_{TCPW}	200	—	ns
SYSCLK Period	t_{SCP}	240	10,000	ns
Rise, Fall Time	t_r, t_f	—	5	ns

SWITCHING CHARACTERISTICS (Cont'd.)

LIU INTERFACE (Cont'd.)



R8071A TCLK — RCLK Relationship Waveforms

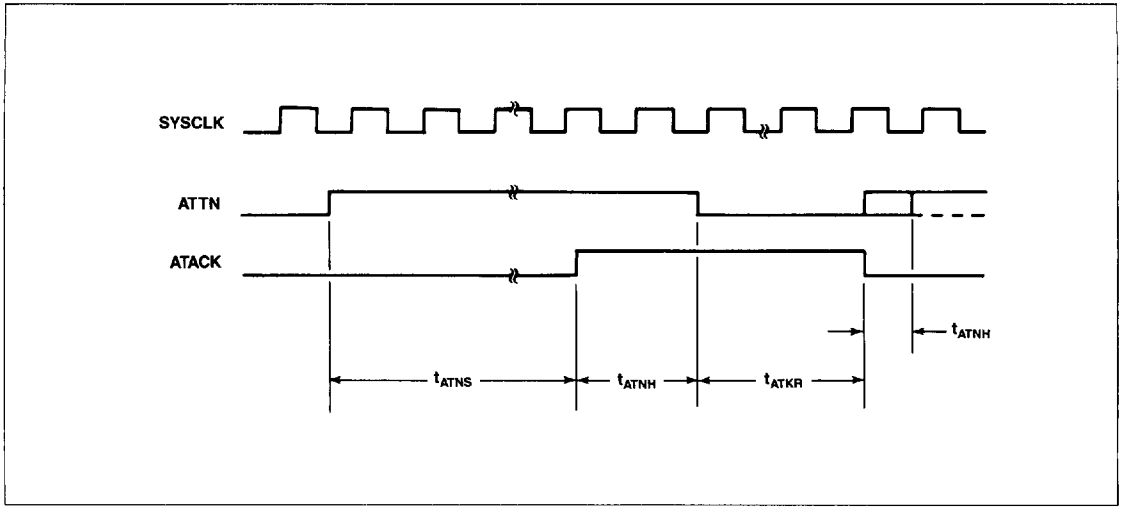
R8071A TCLK — RCLK Relationship Timing

Parameter	Symbol	Min.	Max.	Units
Rise, Fall Time	t_r, t_f	—	10	ns
RCLK Pulse Width	t_{RCPW}	190	—	ns
TCLK, RCLK Difference	t_{TRCD}	—	Note 1	ns

Note:
1. RCLK is to be centered around TCLK. The summation of RCLK and TCLK periodic differences over any duration of time must never exceed 14 TCLK periods.

SWITCHING CHARACTERISTICS (Cont'd.)

CHANNEL ACTIVATION/DEACTIVATION



Channel Activation/Deactivation Waveforms

Channel Activation/Deactivation Timing

Parameter	Symbol	Min.	Max.	Units
ATTN to ATACK Response Time	t_{ATNS}	20	48	SYSCLKS
ATTN Hold time	t_{ATNH}	0	—	ns
ATACK Reset Delay	t_{ATKR}	2	4	SYSCLKS

R8071A INPUT AC ELECTRICAL CHARACTERISTICS

Signal Name	Reference Signal	Edge[1]	Setup (Min.)	Hold (Min.)	Units
ATTN	SYSCLK	PE	50	50	ns
RESET	TCLK	NE	60	60	ns
D0–D7	$\overline{\text{READ}}$	NE/PE	50	0	ns
TMAX (SIS = I)	TCLK	NE	60	60	ns
TMAX (SIS = O)	TCLK	PE	60	60	ns
RSER (SIS = I)	RCLK	NE	50	50	ns
RSER (SIS = O)	RCLK	PE	50	50	ns
RRED (SIS = I)	RCLK	NE	50	50	ns
RRED (SIS = O)	RCLK	PE	50	50	ns
RSYNC (SIS = I)	RCLK	NE	50	50	ns
RSYNC (SIS = O)	RCLK	PE	50	50	ns

Notes:
1. PE = positive edge; NE = negative edge.
2. All input AC Timing measurements are referenced to the 0.8 and 2.0 Vdc logic levels.

R8071A OUTPUT AC ELECTRICAL CHARACTERISTICS

Signal Name	Reference Signal	Edge[1]	Max. DELAY	Min. HOLD	Units
DMND	SYSCLK	PE	75	10	ns
$\overline{\text{AS}}$	SYSCLK	PE/NE	75	10	ns
A0–A15	SYSCLK	PE	90	10	ns
SYSACC	SYSCLK	PE	75	10	ns
$\overline{\text{READ}}$	SYSCLK	PE	75	10	ns
$\overline{\text{WRITE}}$	SYSCLK	PE/NE	75	10	ns
D0–D7	SYSCLK	NE	90	10	ns
$\overline{\text{INTR}}$	SYSCLK	NE	75	10	ns
CH0–CH4	SYSCLK	PE	140	10	ns
RX/$\overline{\text{TX}}$	SYSCLK	PE	140	10	ns
ATACK	SYSCLK	PE	75	10	ns
TSER	TCLK	NE	75	10	ns

Notes:
1. PE = positive edge; NE = negative edge.
2. All output AC Timing measurements are referenced to the 0.4 and 2.4 Vdc logic levels.

ABSOLUTE MAXIMUM RATINGS*

Parameter	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to 7.0	Vdc
Operating Temperature	T_A	0 to +70	°C
Storage Temperature	T_{STG}	−55 to +150	°C

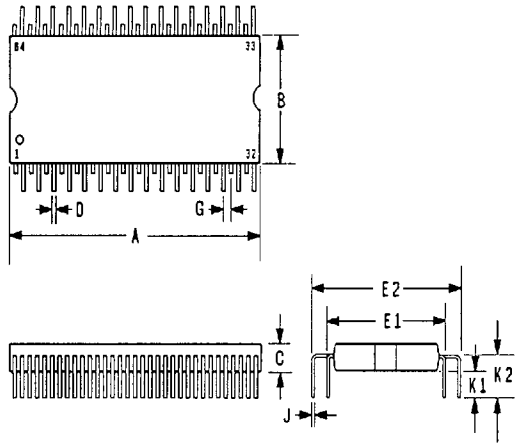
***NOTE:** Stresses above those listed may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the other sections of this document is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

DC ELECTRICAL CHARACTERISTICS

(V_{CC} = 5 Vdc ±5%, V_{SS} = 0 Vdc, T_A = 0 to +70°C, unless otherwise noted)

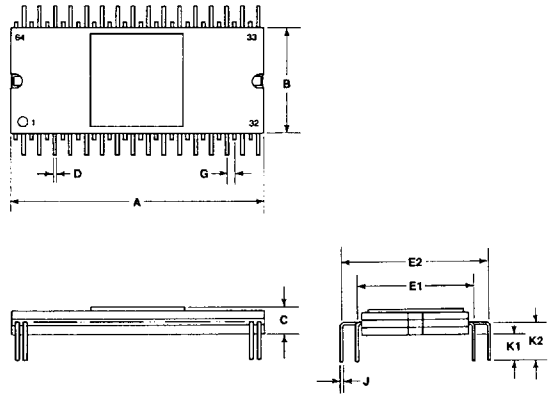
Parameter	Symbol	Min	Max	Unit	Test Condition
Input Low Voltage	V_{IL}	−0.3	0.8	V	
Input High Voltage	V_{IH}	2.0	V_{CC} + 0.3	V	
Output Low Voltage	V_{OL}	—	0.4	V	I_{LOAD} = +1.6 mA
Output High Voltage CMOS	V_{OH}	 3.5	 —	V	 I_{LOAD} = +100 µA
Output Low Current CH0–CH4, RX/$\overline{TX}$, TSER Others	I_{OL}	 +3.2 +1.6	 — —	mA	V_{OL} = 0.4V
Output High Current CH0–CH4, RX/$\overline{TX}$, TSER Others	I_{OH}	 −200 −100	 — —	µA	V_{OH} = 3.5V
Input Capacitance	C_{IN}	—	5	pF	
Output Capacitance (Load) TSER All Others	C_{OUT}	— — —	 100 50	 pF pF	
Power Dissipation	P_{WD}	—	250	mW	

PACKAGE DIMENSIONS



DIM.	MILLIMETERS MIN.	MAX.	INCHES MIN.	MAX.
A	41.10	41.61	1.618	1.638
B	17.02	17.23	0.670	0.690
C	3.56	4.58	0.140	0.180
D	0.48	0.56	0.018	0.022
E1	19.05 BSC		0.750 BSC	
E2	23.50 BSC		0.925 BSC	
G	1.27 BSC		0.050 BSC	
J	0.18	0.33	0.007	0.013
K1	2.92	3.18	0.115	0.125
K2	4.83	5.34	0.190	0.210

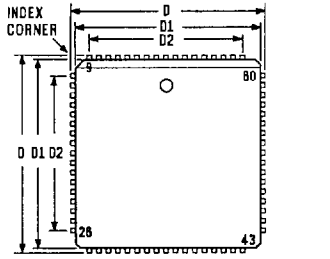
64-Pin Plastic Quad In-Line Package (QUIP)



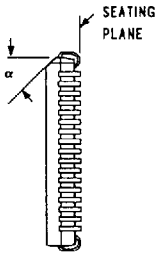
DIM.	MILLIMETERS MIN.	MAX.	INCHES MIN.	MAX.
A	40.77	41.28	1.605	1.625
B	16.76	17.27	0.660	0.680
C	3.56	4.58	0.140	0.180
D	0.48	0.56	0.018	0.022
E1	19.05 BSC		0.750 BSC	
E2	23.50 BSC		0.925 BSC	
G	1.27 BSC		0.050 BSC	
J	0.20	0.30	0.008	0.012
K1	2.92	3.18	0.115	0.125
K2	4.83	5.34	0.190	0.210

64-Pin Cerpac Quad In-Line Package (QUIP)

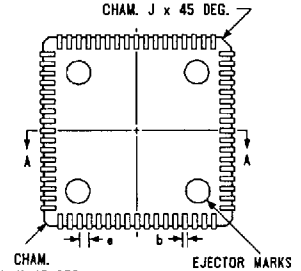
PACKAGE DIMENSIONS



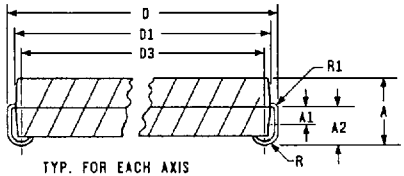
TOP VIEW



SIDE VIEW



BOTTOM VIEW

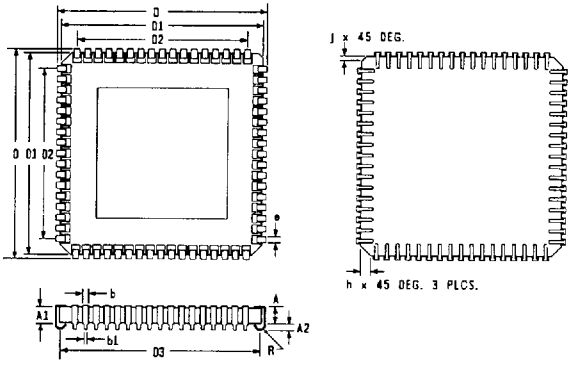


SECTION A-A

DIM.	MILLIMETERS MIN.	MAX.	INCHES MIN.	MAX.
A	4.14	4.39	0.163	0.173
A1	1.37	1.47	0.054	0.058
A2	2.31	2.46	0.091	0.097
b	0.457 TYP		0.018 TYP	
D	25.02	25.27	0.985	0.995
D1	24.00	24.26	0.945	0.955
D2	20.19	20.45	0.795	0.805
D3	23.24	23.50	0.915	0.925
e	1.27 BSC		0.050 BSC	
h	0.254 TYP		0.010 TYP	
J	1.15 TYP		0.045 TYP	
α	45° TYP		45° TYP	
R	0.89 TYP		0.035 TYP	
R1	0.254 TYP		0.010 TYP	

68-Pin Plastic Leaded Chip Carrier (PLCC)

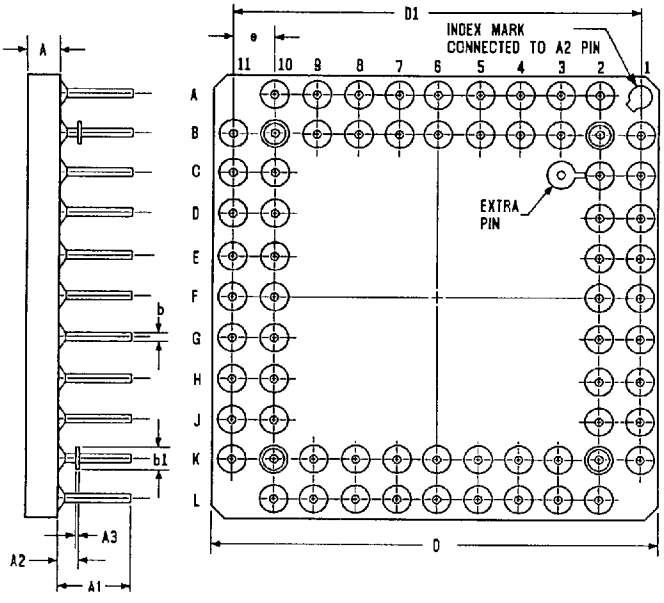
PACKAGE DIMENSIONS



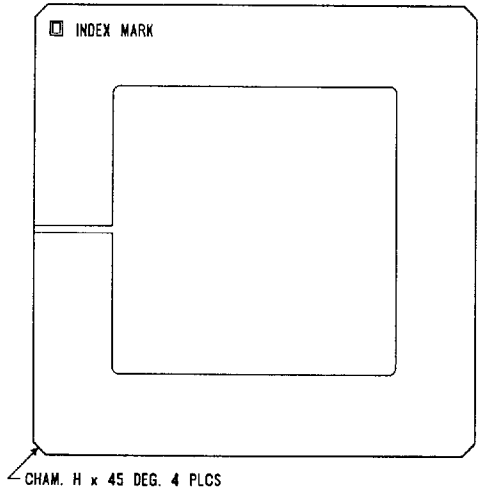
DIM.	MILLIMETERS MIN.	MILLIMETERS MAX.	INCHES MIN.	INCHES MAX.
A	1.83	2.24	0.072	0.088
A1	2.41 REF		0.095 REF	
A2	0.89 REF		0.035 REF	
D	24.64	25.65	0.970	1.010
D1	23.88	24.51	0.940	0.965
D2	23.11	24.13	0.910	0.950
D3	20.20	20.45	0.795	0.805
e	1.27 TYP		0.050 TYP	
h	1.02 REF		0.010 TYP	
J	0.51 REF		0.020 REF	

68-Pin Ceramic Leaded Chip Carrier (CLCC)

PACKAGE DIMENSIONS



DIM.	MILLIMETERS MIN.	MAX.	INCHES MIN.	MAX.
A	1.70	2.11	0.067	0.083
A1	4.45	4.70	0.175	0.185
A2	1.19	1.35	0.047	0.053
A3	0.20 REF		0.008 REF	
b	0.41	0.51	0.016	0.020
b1	1.19	1.35	0.047	0.053
D	27.58	28.19	1.086	1.110
D1	25.15	25.65	0.990	1.010
e	2.41	2.67	0.095	0.105
h	0.51 REF		0.020 REF	



68-Pin Grid Array (PGA)