QUAD E1 FRAMER IC

JUNE 2001 REV. P1.0.4

GENERAL DESCRIPTION

The XRT84V24 Quad E1 Framer contains four independent E1 Framer blocks. Each E1 Framer block contains its own Transmit and Receive E1 Framing function, Transmit HDLC Controller (which encapsulates contents of Transmit HDLC Buffers into LAPD Message frames) and Receiver HDLC Controller (which extracts payload content of "Receive LAPD Message" frames from the incoming E1 data stream and writes it into the Receive HDLC Buffer). Each framer also contains a Transmit Overhead data Input port interface which permits the external Terminal equipment to insert its own Data Link within the outbound E1 frames. Finally, each framer also contains a receive overhead data output interface port. The receive overhead data output interface block allows the terminal equipment to have direct access to the incoming data link bits.

FEATURES

- Four independent, ITU-T G.704 compliant Transceiver E1 Framers
- Supports Channel Associated Signaling (CAS)
- Supports Common-Channel and Primary Rate ISDN Signaling
- Supports FAS, CRC-Multiframe and CAS Multiframe framing structures
- Contains two 96 byte Transmit HDLC Buffers and two 96 byte Receive HDLC buffers for each channel
- Contains Microprocessor Interface for popular types of Microprocessors and supports Programmed I/O, Burst and DMA modes of Read/Write access
- Each framer block can encode or decode the E1 Frame data into/from the Single-Rail or Dual-Rail (AMI or HDB3 encoded) formats
- Detects and forces RAI and AIS Alarms
- Detects LOF, COFA and LOS conditions
- Each Framer Contains a 512 bit Elastic Store Buffer
- Uses a Single +3.3V Power Supply
- Available in a 208 pin PQFP package

APPLICATIONS

- SDH terminal or add/drop multiplexers supporting E1 framing
- E1 multiplexers
- Channel Service Units (CSUs)
- LAN routers with integrated E1 interfaces
- E1 Frame Relay Interface
- ISDN Primary Rate Interfaces
- Test Equipment

FIGURE 1. BLOCK DIAGRAM OF THE XRT84V24

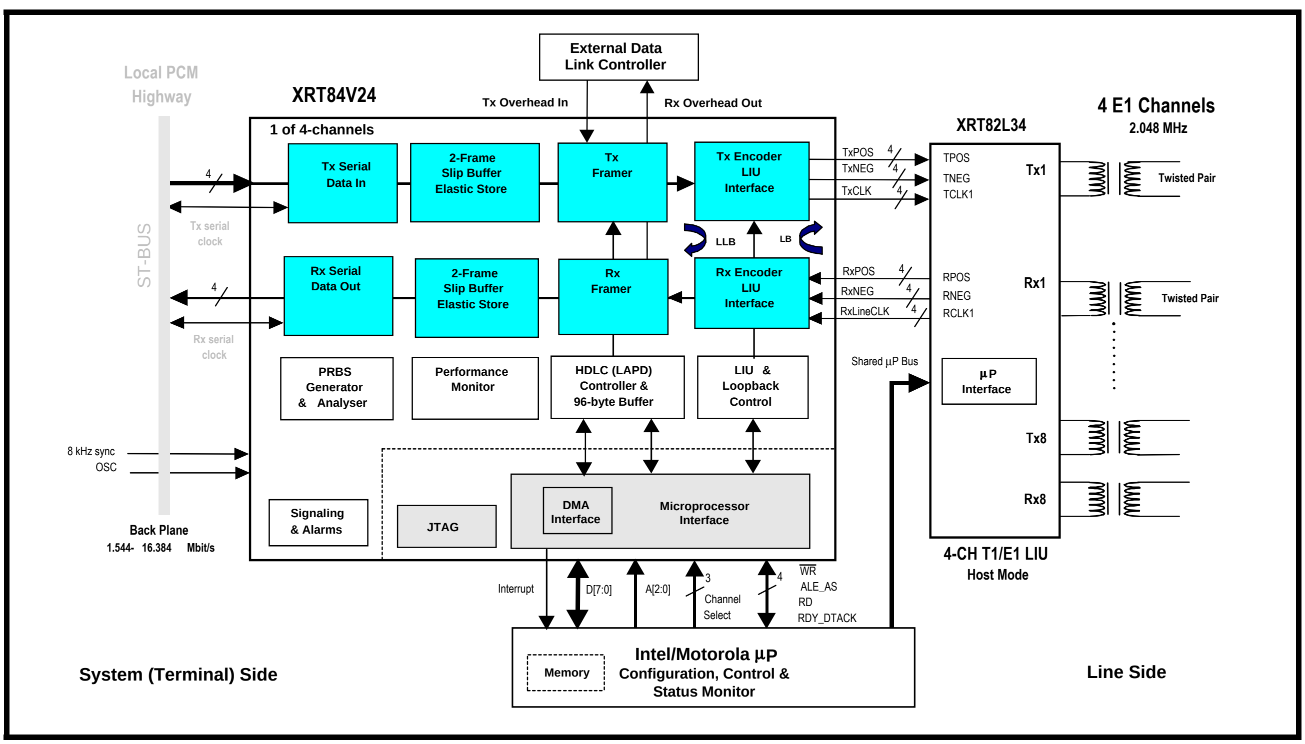
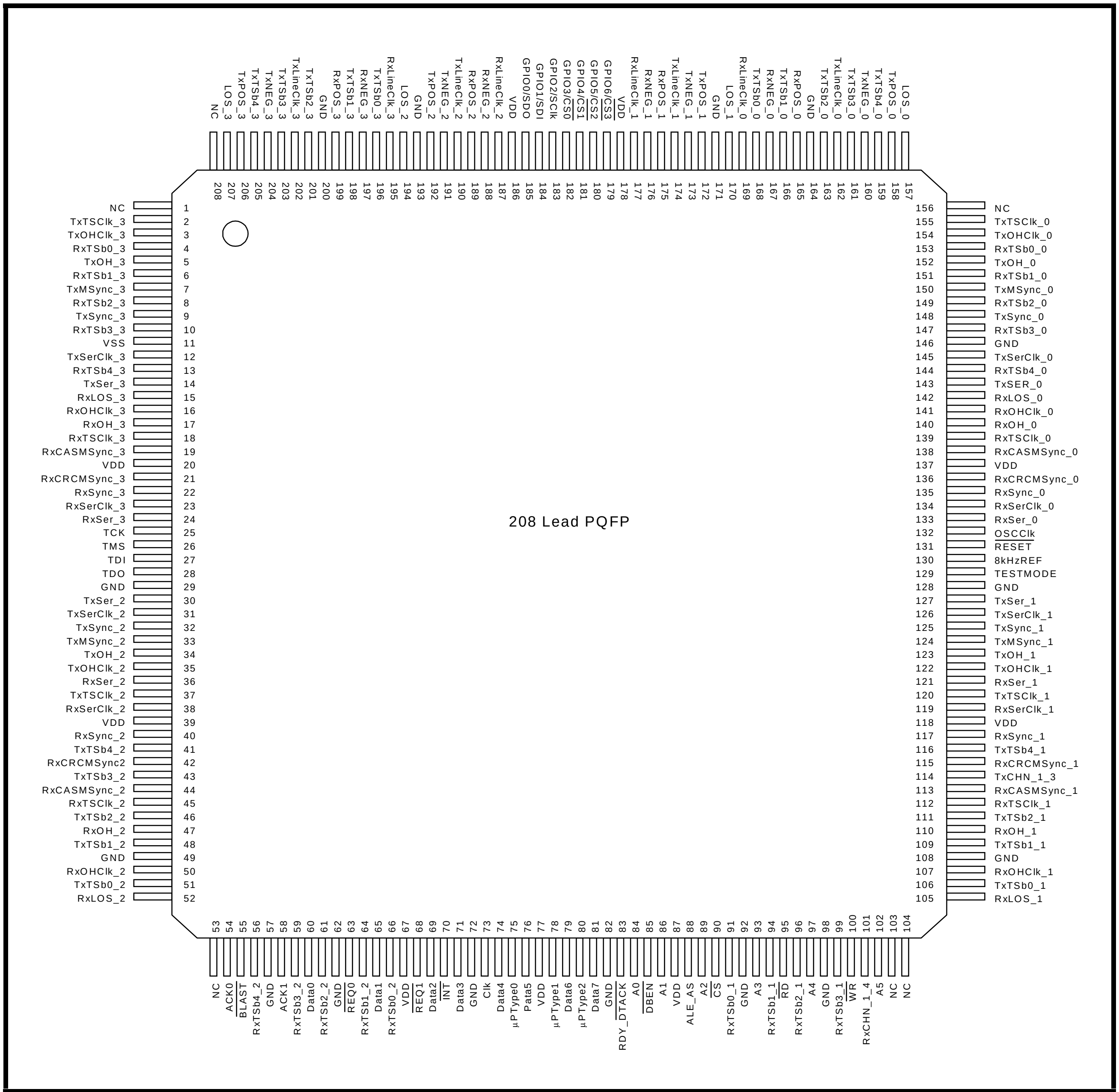


Figure 2. Pin Out of the XRT84V24 in the 208 pin PQFP package



ORDERING INFORMATION

Part Number	Package	Operating Temperature Range
XRT84V24IV-208	208 Lead PQFP	-40°C to +85°C

TABLE OF CONTENTS

PIN DESCRIPTIONS

TX SERIAL DATA INPUT

PIN #	NAME	TYPE	DESCRIPTION
(Framer_n Channel indicated by _n)			
155 120 37 2	TxTSClk_0 TxTSClk_1 TxTSClk_2 TxTSClk_3	O	**Transmit Channel Clock Output Signal—Framer_n:** Each of these output pins are a 256kHz clock output which pulses "high" whenever the Transmit Payload Data Input Interface block accepts the LSB of each of the 32 octets, within the E1 data stream, being processed via Framer _n. The Terminal Equipment should use this clock signal to sample the TxTSb0_n through TxTSb4_n output signals, and identify the time-slot being processed via the "Transmit Section" of each Framer_n.
150 124 33 7	TxMSync_0 TxMSync_1 TxMSync_2 TxMSync_3	I/O	**E1 Multiframe Sync Pulse Input/Output—Framer_n:** This pin is configured to be an input if the TxSerClk_n input pin is configured to be the timing reference for the Transmit section of Framer_n. Conversely, this pin will be configured as an output if the RxLineClk input pin or the OSCClk input pins are configured to be the timing reference for the Transmit section of Framer_n. The role of these pins when configured as an input or output, is described below. **When pin is configured to be an Input:** If this pin is configured to be an input, this pin must be pulsed "High" for one period of TxSerClk_n, the instant that the Transmit payload data Interface (of Framer_n) is processing the first International Bit (Si) of an "outbound" CRC payload data Multiframe. ***NOTES:*** *1. This input pin is ignored if CRC Multiframe Alignment has been disabled.* *2. It is imperative that the TxMSync_n input signal be synchronized with (e.g. derived from) the TxSerClk_n input signal.* **When pin is configured to be an Output:** If this pin is configured to be an output, then it will pulse "High", for one period of TxSerClk_n, when the Transmit payload data Input Interface (of Framer_n) is processing the last bit, within an "outbound" CRC Multiframe. ***NOTES:*** *1. This pin is inactive if CRC Multiframe Alignment has been disabled.* *2. The purpose of this output pin is to permit the Terminal Equipment to maintain alignment with the "outbound" CRC-Multi-frame structure.*

PIN #	NAME	TYPE	DESCRIPTION
148 125 32 9	TxSync_0 TxSync_1 TxSync_2 TxSync_3	I/O	**E1 Single Frame Sync Pulse Input/Output—Transmit Framer_0:** This pin is configured to be an **input** if the TxSerClk_0 input pin is configured to be the timing reference for the Transmit Section of Framer_n. This pin will be configured as an **output** if the RxLineClk input pin or the OSCClk input pins are configured to be the timing reference for the Transmit Section of Framer_n. The role of these pins when configured as an input or output, is described below. **When pin is configured to be an Input:** If this pin is configured to be an input, then the user must pulse this pin “High” for one period of TxSerClk_n, when the Transmit payload data Input Interface (of Framer_n) is processing the International Bit (Si) of an outbound E1 frame. ***NOTE:*** *It is imperative that the TxSync_n input signal be synchronized with (e.g. derived from) the TxSerClk_n input signal.* **When pin is configured to be an Output:** If this pin is configured to be an output, then it will pulse “High”, for one period of TxSerClk_n, when the Transmit payload data Input Interface (of Framer_n) is processing the last bit within a given outbound E1 frame.
145 126 31 12	TxSerClk_0 TxSerClk_1 TxSerClk_2 TxSerClk_3	I/O	**Transmit Serial Input—Clock Signal for Framer_n:** This signal is used by the Transmit payload data Input Interface, to latch the contents of the TxSer_n signal into the Quad E1 Framer IC. Data that is applied at the TxSer_n input is latched into the Transmit payload data Input Interface (for Framer_n) on the rising edge of TxSerClk_n. TxSerClk_n can either be an input or an output. If the Transmit Section of Framer_n has been configured to use the TxSerClk_n signal as the timing source, then this signal will be an **Input**. If the Transmit Section of Framer_n has been configured to use either the RxLineClk signal or the OSCClk signal as the timing source, then TxSerClk_n will be an **Output**.
143 127 30 14	TxSer_0 TxSer_1 TxSer_2 TxSer_3	I	**Transmit Serial Input—Framer_n:** This input pin, along with TxSerClk_n function as the Transmit payload data input port, for Framer_n. Any data that is applied to this pin, will be inserted into an outbound E1 frame, and ultimately output onto the E1 line, via the TxPOS_n and TxNEG_n output pins. All data that is intended to be transported via Time Slots 1 through 15 and Time slots 17 through 31, within each E1 frame must be applied to this input pin. If Framer_n is configured accordingly, data intended for Time Slots 0 and 16 can also be applied to input pin as well. The signal, applied to this input pin, is latched into the Transmit Payload Data Input Interface on the rising edge of TxSerClk_n.

Pin #	Name	Type	Description
168 166 163 161 159 106 109 111 114 116 51 48 46 43 41 196 198 201 203 205	TxTSb0_0 TxTSb1_0 TxTSb2_0 TxTSb3_0 TxTSb4_0 TxTSb0_1 TxTSb1_1 TxTSb2_1 TxTSb3_1 TxTSb4_1 TxTSb0_2 TxTSb1_2 TxTSb2_2 TxTSb3_2 TxTSb4_2 TxTSb0_3 TxTSb1_3 TxTSb2_3 TxTSb3_3 TxTSb4_3	O	**Transmit Framer_n-Time Slot Octet Identifier Output-Bit [0:4]:** These output signals (TxTSb4_n through TxTSb0_n) reflects the five-bit binary value of the Time Slot number (in the incoming E1 frame), being accepted and processed by the Transmit Payload Data Input Interface block associated with Framer_n. Terminal Equipment should use the TxTSClk_n clock signal to sample the five output pins of each channel in order to identify the time-slot being processed by the Transmit Payload Data Input Interface block of Framer_n.

E1 OVERHEAD INTERFACE

Pin #	Name	Type	Description
(Framer_n Channel indicated by _n)			
154 122 35 3	TxOHClk_0 TxOHClk_1 TxOHClk_2 TxOHClk_3	O	**Transmit OH Serial Clock Output Signal—Framer_n:** This output clock signal functions as a demand clock signal for the “Transmit Overhead data Input Interface” block associated with Framer_n. If the “Transmit Overhead data Input Interface” has been configured to be the source of Data Link information, then the Transmit Overhead data Input Interface block will provide a clock edge for each “Sa” bit that is carrying data link information. If the “Transmit Overhead data Input Interface has not been configured to be the source of the Data Link information, then this output signal will be inactive. Data Link Equipment, which is interfaced to this pin, should update its data (on the TxOH_n line) on the rising edge of this clock signal. The Transmit Overhead data Input Interface will latch the data (on the TxOH_n line) on the falling edge of this clock signal.
152 123 34 5	TxOH_0 TxOH_1 TxOH_2 TxOH_3	I	**Transmit Overhead Input—Framer_n:** This input pin, along with TxOHClk_n functions as the Transmit Overhead data input port for Framer_n. This input pin will become active if the Transmit Section of Framer_n has been configured to use this input as the source of Data Link bits. The data that is input into this pin will be inserted into the Sa4 through Sa8 bits (the National Bits) within the outbound non-FAS E1 frames. ***NOTE:*** *This input pin will be disabled if Framer_n is using the Transmit HDLC Controller, or the TxSer_n input as the source for the Data Link Bits.*

PIN #	NAME	TYPE	DESCRIPTION
141 107 50 16	RxOHClk_0 RxOHClk_1 RxOHClk_2 RxOHClk_3	O	**Receive OH Serial Clock Output Signal—Framer_n:** This pin, along with "RxOH_n" functions as the "Receive Overhead data Output Interface" for Framer_n. This pin outputs a clock edge corresponding to each National Bit that is carrying "Data Link" information. This output pin is inactive if the Receive HDLC Controller (within Framer_n) has been enabled.
140 110 47 17	RxOH_0 RxOH_1 RxOH_2 RxOH_3	O	**Receive Overhead Output—Framer_n:** This pin, along with RxOHClk_n functions as the "Receive Overhead data Output Interface" for Framer_n. This pin unconditionally outputs the contents of the National Bits (the "Sa4" through the "Sa8" bits). This output pin is active even if the Receive HDLC Controller (within Framer_n) is active. If Framer_n has been configured to interpret the National bits of the incoming E1 frames as carrying "Data Link" information; then the Receive Overhead Output Interface will provide a clock pulse (via the RxOHClk_n output pin) for each "Sa" bit carrying Data Link information.

RX SERIAL DATA OUTPUT

PIN #	NAME	TYPE	DESCRIPTION
(Framer_n Channel indicated by _n)			
153 151 149 147 144 91 94 96 99 101 66 64 61 59 56 4 6 8 10 13	RxTSb0_0 RxTSb1_0 RxTSb2_0 RxTSb3_0 RxTSb4_0 RxTSb0_1 RxTSb1_1 RxTSb2_1 RxTSb3_1 RxTSb4_1 RxTSb0_2 RxTSb1_2 RxTSb2_2 RxTSb3_2 RxTSb4_2 RxTSb0_3 RxTSb1_3 RxTSb2_3 RxTSb3_3 RxTSb4_3	O	**Receive Framer_n-Time Slot Octet Identifier Output-Bit [0:4]:** These output signals reflect the five-bit binary value of the Time Slot number (in the incoming E1 frame) being received and output to the Terminal Equipment via the Receive Payload Data output Interface block associated with Framer_n. The Terminal Equipment should use the RxTSClk_n clock to sample these five output pins in order to identify the time-slot being processed by the Receive Section of Framer_n.
142 105 52 15	RxLOS_0 RxLOS_1 RxLOS_2 RxLOS_3	O	**Receive Loss of Signal Output Indicator—Framer_n:** This output pin will toggle "High" (declare LOS) if the Receive E1 Framer block associated with Framer_n determines that neither the RxPOS_n or the RxNEG_n inputs have received a "High" level pulse in the last 32 bit periods. This output pin will toggle "Low" if the Receive E1 Framer block, associated with Framer_n, detects a string of 32 consecutive bits, that does not contain a string of 4 consecutive "0's". ***NOTE:*** *This output pin will also toggle "High" if the LOS_n input pin is asserted (e.g., toggled "High" by the LIU LOS output pin).*

PIN #	NAME	TYPE	DESCRIPTION
139 112 45 18	RxTSClk_0 RxTSClk_1 RxTSClk_2 RxTSClk_3	O	**Receive Channel Clock—Framer_n:** Each of these output pins is a 256kHz clock signal pulses “High” coincident with the Receive Payload Data Output port (or Receive Overhead Data Output port) places the LSB of a given Time Slot (within the inbound E1 data stream) on the RxSer_n or RxOH_n output pins. The Receive Terminal Equipment can use this clock signal in order to identify the time-slot being received by the “Receive Section” associated with Framer_n. The Terminal Equipment accomplishes this by sampling the RxTSb0_n through RxTSb4_n output pins.
138 113 44 19	RxCASMSync_0 RxCASMSync_1 RxCASMSync_2 RxCASMSync_3	O	**Receive “CAS Multiframe” Sync Output Signal—Framer_n:** This output pin pulses “High” for one period of RxSerClk_n whenever the Receive Payload Data Output Interface block outputs the first bit, within a given “CAS Multiframe”. ***NOTE:*** *This output pin is inactive if Common Channel Signaling is enabled.*
136 115 42 21	RxCRCMSync_0 RxCRCMSync_1 RxCRCMSync_2 RxCRCMSync_3	O	**Receive “CRC Multiframe” Sync Output signal—Framer_n:** This output pin pulses “High” for one period of RxSerClk_n whenever the Receive E1 Output Interface of Framer_n outputs the first bit, within a given “CRC Multiframe”. ***NOTE:*** *This output pin is inactive if CRC Multiframe Alignment is disabled.*
135 117 40 22	RxSync_0 RxSync_1 RxSync_2 RxSync_3	I/O	**Receive E1 Single Frame Sync Pulse Input/Output pin—Framer_n:** This pin is configured to be an **Input** if the Slip Buffer associated with Framer_n is enabled. Conversely, this pin will be configured to be an **Output** if the “Slip-Buffer” is “by-passed”. **Input:** If this pin is configured to be an input, then this pin must be pulsed “High” for one period of RxSerClk_n, when the Receive E1 Serial (or Overhead) Output Interface, outputs the International bit (Si) of an inbound E1 frame. **Output:** If this pin is configured to be an output, then it will pulse “High” for one period of RxSerClk_n, when the Receive E1 Serial (or Overhead) output Interface outputs the last bit, in an inbound E1 frame.
134 119 38 23	RxSerClk_0 RxSerClk_1 RxSerClk_2 RxSerClk_3	I	**Receive E1 Serial Output Interface—Clock Input Signal for Framer_n:** This signal is used by the Receive E1 Serial Payload Data Output interface to clock out inbound E1 frame data via the RxSer_n output pin.
133 121 36 24	RxSer_0 RxSer_1 RxSer_2 RxSer_3	O	**Receive E1 Serial Output—Framer_n:** This output pin, along with RxSerClk_n function as the Receive Payload Data Output Interface port for Framer_n. Much of the data that is received from the line via the RxPOS_n and RxNEG_n input pins, will be decoded and output via this pin, in a binary format. All data that is transported via Time Slots 1 through 15 and Time Slots 17 through 31, within each incoming E1 frame, will be output via this pin. If Framer_n is configured accordingly, the data for Time Slots 0 and 16 will also be output via this pin.

RX DECODER LIU INTERFACE

PIN #	NAME	TYPE	DESCRIPTION
(Framer_n Channel indicated by _n)			
165 175 189 199	RxPOS_0 RxPOS_1 RxPOS_2 RxPOS_3	I	**Receive Positive Polarity Pulse—Framer_n:** This input pin is intended to be connected to the RxPOS or RPDATA output of the LIU (Line Interface Unit) IC associated with Framer_n. The LIU will assert this input signal (pulse it “High”), when it is receiving a positive-polarity pulse from the line. This input signal is sampled and latched into the Quad E1 Framer, on the “user-selectable” edge (rising or falling) of the RxLineClk_n signal.
167 176 188 197	RxNEG_0 RxNEG_1 RxNEG_2 RxNEG_3	I	**Receive Negative Polarity Pulse—Framer_n:** This input pin is intended to be connected to the RxNEG or RNDATA output of the LIU (Line Interface Unit) associated with Framer_n. The LIU will assert this input signal (pulse it “High”), when it is receiving a negative-polarity pulse from the line. This input signal is sampled and latched into the Quad E1 Framer, on the user-selectable edge (rising or falling) of the RxLineClk_n signal.
169 177 187 195	RxLineCLK_0 RxLineCLK_1 RxLineCLK_2 RxLineCLK_3	I	**Receive Line Clock Input—Framer_n:** This input pin is intended to be connected to the RxClk output of the LIU (Line Interface Unit) IC associated with Framer_n. The Quad E1 IC Framer uses the user-selectable edge of this signal to sample (and latch) the signals at the RxPOS_n and RxNEG_n input pins.

TX ENCODER LIU INTERFACE

PIN #	NAME	TYPE	DESCRIPTION
(Framer_n Channel indicated by _n)			
158 172 192 206	TxPOS_0 TxPOS_1 TxPOS_2 TxPOS_3	O	**Transmit Positive Polarity Pulse—Framer_n:** This output pin is intended to be connected to the TxPOS or TPDATA input of the LIU (Line Interface Unit) IC associated with Framer_n. The Quad E1 Framer will assert this signal when it wishes for the Line Interface Unit (LIU) IC associated with Framer 0, to transmit a positive polarity pulse on the line.
160 173 191 204	TxNEG_0 TxNEG_1 TxNEG_2 TxNEG_3	O	**Transmit Negative Polarity Pulse—Framer_n:** This output pin is intended to be connected to the TxNEG or TNDATA input of the LIU (Line Interface Unit) IC associated with Framer_n. The Quad E1 Framer IC will assert this signal when it wishes for the Line Interface Unit (LIU) IC associated with Framer_n, to transmit a negative polarity pulse on the line.
162 174 190 202	TxLineCLK_0 TxLineCLK_1 TxLineCLK_2 TxLineCLK_3	O	**Transmit Line Clock Output—Framer_n:** This output pin is intended to be connected to the TxClk input of the LIU (Line Interface Unit) associated with Framer_n. The LIU uses this pin to sample and latch the signals at its TPDATA and TNDATA input pins.

TIMING

PIN #	NAME	TYPE	DESCRIPTION
(Framer_n Channel indicated by _n)			
130	8kHzREF	I	**8kHz Reference Clock Input**
132	OSCClk	I	**Oscillator Clock Input:** The user can apply any one of three frequencies to this input pin: 16.384 MHz, 32.768 MHz, or 65.536 MHz.

LIU CONTROL

PIN #	NAME	TYPE	DESCRIPTION
(Framer_n Channel indicated by _n)			
157 170 194 207	LOS_0 LOS_1 LOS_2 LOS_3	I	**Loss of Signal Input—LIU Interface-Framer_n:** This input pin is intended to be connected to the RxLOS output pin of the LIU IC associated with Framer_n. If the LIU IC detects an LOS condition and asserts (toggles “High”) this input pin, then the Receive E1 Framer associated with Framer_n will declare an “LOS” condition. Asserting this input pin “High” will result in the Quad E1 Framer IC declaring a LOS condition and asserting the RxLOS_n output pin.
179	GPIO6/$\overline{\text{CS3}}$	O	**General Purpose Output pin/Chip Select Output pin:** The exact role of this output pin depends upon whether the LIU Controller block is operating in the Hardware or Host Mode. **Hardware Mode: GPIO6** This pin is a general purpose output pin that is controlled by the contents of bit-field 6, within the Line Control Register (Address = 0x00, 0x02). **Host Mode:$\overline{\text{CS3}}$** This pin is a chip select output pin that is asserted (toggles “Low”) following a write operation to the LIU Access Register 1, associated with framer 3. This output pin is intended to be tied to the chip select input of an LIU (or other peripheral device) that is configurable via a Microprocessor Serial Interface. Once the Host Mode serial port has completed its read or write operation, then it will negate (toggle “High”) this output pin.
180	GPIO5/$\overline{\text{CS2}}$	O	**General Purpose Output pin/Chip Select Output pin:** The exact role of this output pin depends upon whether the LIU Controller block is operating in the Hardware or Host Mode. **Hardware Mode: GPIO5** This pin is a general purpose output pin that is controlled by the contents of bit-field 5, within the Line Control Register (Address = 0x00, 0x02). **Host Mode:$\overline{\text{CS2}}$** This pin is a chip select output pin that is asserted (toggles “Low”) following a write operation to the LIU Access Register 1, associated with framer 2. This output pin is intended to be tied to the chip select input of an LIU (or other peripheral device) that is configurable via a Microprocessor Serial Interface. Once the Host Mode serial port has completed its read or write operation, then it will negate (toggle “High”) this output pin.

PIN #	NAME	TYPE	DESCRIPTION
181	GPIO4/$\overline{CS1}$	O	**General Purpose Output pin/Chip Select Output pin:** The exact role of this output pin depends upon whether the LIU Controller block is operating in the Hardware or Host Mode. **Hardware Mode: GPIO4** This pin is a general purpose output pin that is controlled by the contents of bit-field 4, within the Line Control Register (Address = 0x00, 0x02). **Host Mode: $\overline{CS1}$** This pin is a chip select output pin that is asserted (toggles "Low") following a write operation to the LIU Access Register 1, associated with framer 1. This output pin is intended to be tied to the chip select input of an LIU (or other peripheral device) that is configurable via a Microprocessor Serial Interface. Once the Host Mode serial port has completed its read or write operation, then it will negate (toggle "High") this output pin.
182	GPIO3/$\overline{CS0}$	O	**General Purpose Output pin/Chip Select Output pin:** The exact role of this output pin depends upon whether the LIU Controller block is operating in the Hardware or Host Mode. **Hardware Mode: GPIO3** This pin is a general purpose output pin that is controlled by the contents of bit-field 3, within the Line Control Register (Address = 0x00, 0x02). **Host Mode: $\overline{CS0}$** This pin is a chip select output pin that is asserted (toggles "Low") following a write operation to the LIU Access Register 1, associated with framer 0. This output pin is intended to be tied to the chip select input of an LIU (or other peripheral device) that is configurable via a Microprocessor Serial Interface. Once the "Host" Mode serial port has completed its read or write operation, then it will negate (toggle "High") this output pin.
183	GPIO2/SClk	O	**General Purpose Output/Serial Clock Output:** The exact role of this output pin depends upon whether the LIU Controller block is operating in the Hardware or Host Mode. **Hardware Mode: GPIO2** This pin is a general purpose output pin that is controlled by the contents of bit-field 2, within the Line Control Register (Address = 00h, 02h). **Host Mode: SClk** This pin functions as the Serial Clock output signal (SCLK), when the LIU Controller Block is configured to operate in the Host Mode.
184	GPIO1/SDI	O	**General Purpose Output pin/"Serial Data In" Output pin:** The exact role of this output pin depends upon whether the LIU Controller block is operating in the Hardware or Host Mode. **Hardware Mode: GPIO1** This pin is a general purpose output pin that is controlled by the contents of bit field 1, within the Line Control Register (Address = 00h, 02h) **Host Mode: SDI** This pin functions as the Serial Data Input (SDI) output pin (to the Microprocessor Serial Interface).
185	GPO0/SDO	I/O	**General Purpose Output pin/"Serial Data Out" Input pin:** The exact role of this pin depends upon whether the LIU Controller block is operating in the Hardware or Host Mode. **Hardware Mode: GPIO0** This pin is a general purpose output pin that is controlled by the contents of bit-field 0, within the Line Control Register (Address = 00h, 02h). **Host Mode: SDO** This Input pin functions as the Serial Data Output (SDO) input pin (into the LIU Controller Block).

JTAG

PIN #	NAME	TYPE	DESCRIPTION
(Framer_n Channel indicated by _n)			
25	TCK		**Test clock:** Boundary Scan clock input. ***Note****: This input pin should be pulled “Low” for normal operation*
26	TMS		**Test Mode Select:** Boundary Scan Mode Select input. ***Note****: This input pin should be pulled “Low” for normal operation*
27	TDI		**Test Data In:** Boundary Scan Test data input ***Note****: This input pin should be pulled “Low” for normal operation*
28	TDO		**Test Data Out:** Boundary Scan Test data output
129	Test Mode		**Factory Test Mode Pine** ***Note:*** *User should tie this pin to ground*

MICROPROCESSOR SECTION

PIN #	NAME	TYPE	DESCRIPTION
(Framer_n Channel indicated by _n)			
60 65 69 71 74 76 79 81	DATA0 DATA1 DATA2 DATA3 DATA4 DATA5 DATA6 DATA7	I/O	**Bidirectional Microprocessor Data Bus Bit 0--LSB** **Bidirectional Microprocessor Data Bus Bit 1** **Bidirectional Microprocessor Data Bus Bit 2** **Bidirectional Microprocessor Data Bus Bit 3** **Bidirectional Microprocessor Data Bus Bit 4** **Bidirectional Microprocessor Data Bus Bit 5** **Bidirectional Microprocessor Data Bus Bit 6** **Bidirectional Microprocessor Data Bus Bit 7--MSB**
63 68	$\overline{\text{REQ}}$0 $\overline{\text{REQ}}$1	O	**DMA Cycle Request Output—DMA Controller 0 (Write):** The XRT84V24 asserts this output pin (toggles it “Low”) when at least one of the Transmit HDLC buffers are empty and can receive one more HDLC message. The XRT84V24 negates this output pin (toggles it “High”) when the HDLC buffer can no longer receive another HDLC message. **DMA Cycle Request Output—DMA Controller 1 (Read):** The XRT84V24 asserts this output pin (toggles it “Low”) when one of the Receive HDLC buffer contains a complete HDLC message that needs to be read by the µC/µP. The XRT84V24 negates this output pin (toggles it “High”) when the Receive HDLC buffers are depleted.
70	$\overline{\text{INT}}$	O	**Interrupt Request Output:** The Quad E1 Framer will assert this “active low” output (toggles it “Low”), to the local µP, anytime it requires interrupt service.
73	CLK	I	**Microprocessor Clock Input:** This clock signal is the Microprocessor Interface System clock. This clock signal is used for synchronous/burst/DMA data transfer. The maximum frequency of this clock signal is 33MHz.

<table>
<tr><th>PIN #</th><th>NAME</th><th>TYPE</th><th>DESCRIPTION</th></tr>
<tr><td>75

78
80</td><td>µPTYPE0

µPTYPE1
µPTYPE2</td><td>I</td><td>Microprocessor Type Input: Bit 0 (LSB):
This input pin, along with µPTYPE1 and µPTYPE2 permit the user to specify which type of Microprocessor/Microcontroller to be interfaced the XRT84V24.

Microprocessor Type Input: Bit 1
Microprocessor Type Input: Bit 2
<table>
<tr><th>uPType2</th><th>uPType1</th><th>uPType0</th><th>MICROPROCESSOR TYPE</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>68HC11, 8051, 80C188</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>MOTOROLA 68K</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>INTEL X86</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>INTEL I960</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>IDT3051/51</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>IBM POWER PC 403</td></tr>
</table></td></tr>
<tr><td>83</td><td>RDY_DTACK</td><td>O</td><td>Ready/Data Transfer Acknowledge Output:
The exact behavior of this pin depends upon which Microprocessor the XRT84V24 is configured to interface to:
Intel Type Microprocessors
This output pin toggles “Low” when the XRT84V24 is ready to respond to the current PIO (Programmed I/O) or Burst Transaction.
Motorola Type Microprocessors
This output pin toggles “Low” when the XRT84V24 has completed the current bus cycle.</td></tr>
<tr><td>84
86
89
93
97
102</td><td>AO
A1
A2
A3
A4
A5</td><td>I</td><td>Microprocessor Interface Address Bus Input Bit 0 -- (LSB)
Microprocessor Interface Address Bus Input Bit 1
Microprocessor Interface Address Bus Input Bit 2
Microprocessor Interface Address Bus Input Bit 3
Microprocessor Interface Address Bus Input Bit 4
Microprocessor Interface Address Bus Input Bit 5 -- (MSB)</td></tr>
<tr><td>85</td><td>DBEn</td><td>I</td><td>Data Bus Enable Input pin.</td></tr>
<tr><td>88</td><td>ALE_AS</td><td>I</td><td>Address Strobe/Address_Latch Enable Input</td></tr>
<tr><td>90</td><td>CS</td><td>I</td><td>Microprocessor Interface—Chip Select Input:
The Microprocessor/Microcontroller must assert this input pin (toggle it “Low”) in order to exchange data with the Quad E1 Framer IC.
Note: For the 68K MPU, this signal is generated by address decode and address strobe.</td></tr>
<tr><td>95</td><td>RD</td><td>I</td><td>Microprocessor Interface—Read Strobe Input:
The exact behavior of this pin depends upon the type of Microprocessor/Microcontroller the XRT84V24 Quad E1 Framer has been configured to interface to, as defined by the µPTYPE[2:0] pins.
Note: See pin 75 (µPType0) for the µP selection table.</td></tr>
</table>

PIN #	NAME	TYPE	DESCRIPTION
100	$\overline{\text{WR}}$	I	**Microprocessor Interface—Write Strobe Input** **Low:** Indicates current bus cycle is a write cycle: Intel 51, 188, MIPS350x **High:** Indicates present bus cycle is a write cycle: Intel x86, i960 **Low:** Indicates current bus cycle is a read cycle: Intel x86, i960 **High:** Indicates present bus cycle is a read cycle: Motorola, Power PC 403 **Low:** Also used as write strobe in DMA transfer
54 58	$\overline{\text{ACK0}}$ $\overline{\text{ACK1}}$	I I	**DMA Cycle Acknowledge Input—DMA Controller 0 (Write):** The external DMA Controller will assert this input pin "High" when the following two conditions are met: a. After the DMA Controller, within the XRT84V24 has asserted (toggled "Low"), the Req_0 output signal. b. When the external DMA Controller is ready to transfer data from external memory to the selected Transmit HDLC buffer. At this point, the DMA transfer between the external memory and the selected Transmit HDLC buffer may begin. After completion of the DMA cycle, the external DMA Controller will negate this input pin after the DMA Controller, within the XRT84V24 has negated the REQ_0 output pin. The external DMA Controller must do this in order to acknowledge the end of the DMA cycle. **DMA Cycle Acknowledge Input—DMA Controller 1 (Read):** The external DMA Controller asserts this input pin "High" when the following two conditions are met: a. After the DMA Controller, within the XRT84V24 has asserted (toggled "Low"), the Req_1 output signal. b. When the external DMA Controller is ready to transfer data from the selected Receive HDLC buffer to external memory. At this point, the DMA transfer between the selected Receive HDLC buffer and the external memory may begin. After completion of the DMA cycle, the external DMA Controller will negate this input pin after the DMA Controller, within the XRT84V24 has negated the REQ_1 output pin. The external DMA Controller will do this in order to acknowledge the end of the DMA cycle.
55	BLAST	I	**Last Cycle of Burst Indicator Input:** The Microprocessor asserts this pin when it is performing its last read or write cycle, within a burst operation.
131	$\overline{\text{Reset}}$/TRST	I	**Reset Input:** Active "Low" This input also resets the JTAG and disables output drivers.

POWER, GROUND AND NO CONNECTS

PIN #	NAME	TYPE	DESCRIPTION
(Framer_n Channel indicated by _n)			
20 39 67 77 87 118 137 178 186	VDD	****	Power Supply pin
11 29 49 57 62 72 82 92 98 108 128 146 164 171 193 200	GND	****	Ground Pin
1 53 103 104 156 208	NC		Not Connected

ELECTRICAL CHARACTERISTICS

ABSOLUTE MAXIMUMS

Power Supply.. -0.5V to +3.465V	Power Dissipation PBGA Package........................... 2W
Storage Temperature ....-65°C to 150°C	Input Logic Signal Voltage (Any Pin) ....-0.5V to + 5.5V
Operating Temperature Range.................-40°C to 85°C	ESD Protection..>2000V
Supply Voltage .... GND-0.5V to +VDD + 0.5V	Input Current (Any Pin) .... ± 100mA

DC ELECTRICAL CHARACTERISTICS

Test Conditions: TA = 25°C, VDD = 3.3V ± 5% unless otherwise specified

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNITS	CONDITIONS
I_{DD}	Power Supply Current		100		mA	All Channels on
I_{LL}	Data Bus Tri-State Bus Leakage Current	-10		+10	µA	
V_{IL}	Input Low voltage			0.8	V	
V_{IH}	Input High Voltage	2.0		VDD	V	
V_{OL}	Output Low Voltage	0.0		0.4	V	I_{OL} = -1.6mA
V_{OH}	Output High Voltage	2.4		VDD	V	I_{OH} = 40µA
I_{OC}	Open Drain Output Leakage Current				µA	
I_{IH}	Input High Voltage Current	-10		10	µA	V_{IH} = VDD
I_{IL}	Input Low Voltage Current	-10		10	µA	V_{IL} = GND

AC ELECTRICAL CHARACTERISTICS

Test Conditions: TA = 25(C, VCC = 3.3V ± 5% unless otherwise specified

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNITS	CONDITIONS
Transmit LIU Interface Timing (see Figure 3 and Figure 4)						
t_{30}	Output delay from Rising edge of TxLineClk to rising edge of TxPOS or TxNEG output signal. (Framer is configured to output data on TxPOS and TxNEG on rising edge of TxLineClk	0	5.0	10.0	ns	
t_{31}	Output delay from Falling edge of TxLineClk to rising edge of TxPOS or TxNEG (Framer is configured to output data via TxPOS and TxNEG on falling edge of TxLineClk)	1		4	ns	
$f_{TxLineClk}$	Frequency of TxLineClk clock signal		2.042		MHz	E1 Applications
t_{32}	Period of TxLineClk	480	488	494	ns	E1 Applications

AC ELECTRICAL CHARACTERISTICS

Test Conditions: TA = 25(C, VCC = 3.3V ± 5% unless otherwise specified						
SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNITS	CONDITIONS
Receive LIU Interface Timing (see Figure 5 and Figure 6)						
t_{38}	RxPOS or RxNEG set-up time to rising edge of RxLineClk. (Framer is configured to sample data on RxPOS and RxNEG input pins, on the rising edge of RxLineClk)	3			ns	
t_{39}	RxPOS or RxNEG hold time, from rising edge of RxLineClk (Framer is configured to sample data on RxPOS and RxNEG input pins, on the rising edge of RxLineClk)	5			ns	
t_{40}	RxPOS or RxNEG set-up time to falling edge of RxLineClk. (Framer is configured to sample data on RxPOS and RxNEG input pins, on the falling edge of RxLineClk)	15			ns	
t_{41}	RxPOS or RxNEG hold time, from falling edge of RxLineClk (Framer is configured to sample data on RxPOS and RxNEG input pins, on the falling edge of RxLineClk)	5			ns	

FIGURE 3. TRANSMIT LIU INTERFACE TIMING - TXPOS & TXNEG UPDATED ON RISING EDGE OF TXLINECLK

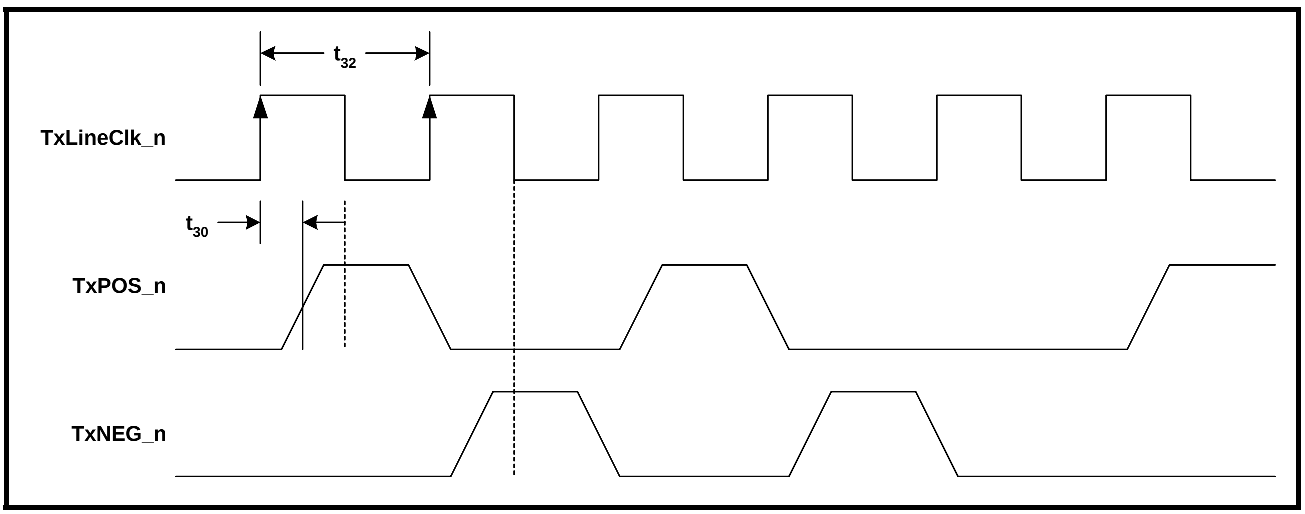


FIGURE 4. TRANSMIT LIU INTERFACE TIMING - TXPOS & TXNEG UPDATED ON FALLING EDGE OF TXLINECLK

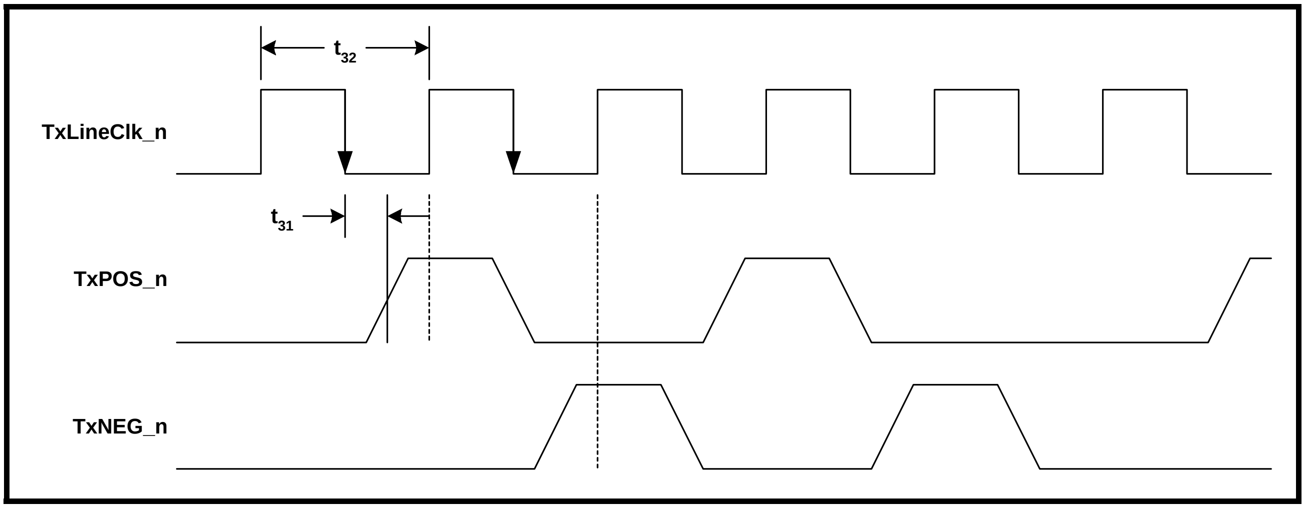


FIGURE 5. RECEIVER LIU INTERFACE TIMING - RXPOS & RXNEG UPDATED ON RISING EDGE OF RXLINECLK

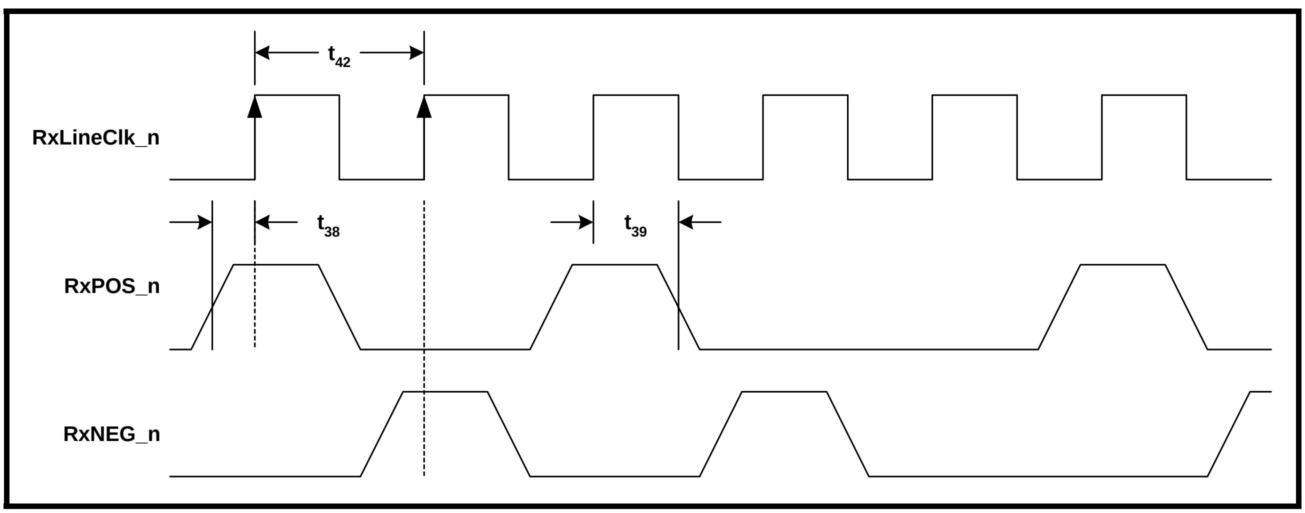
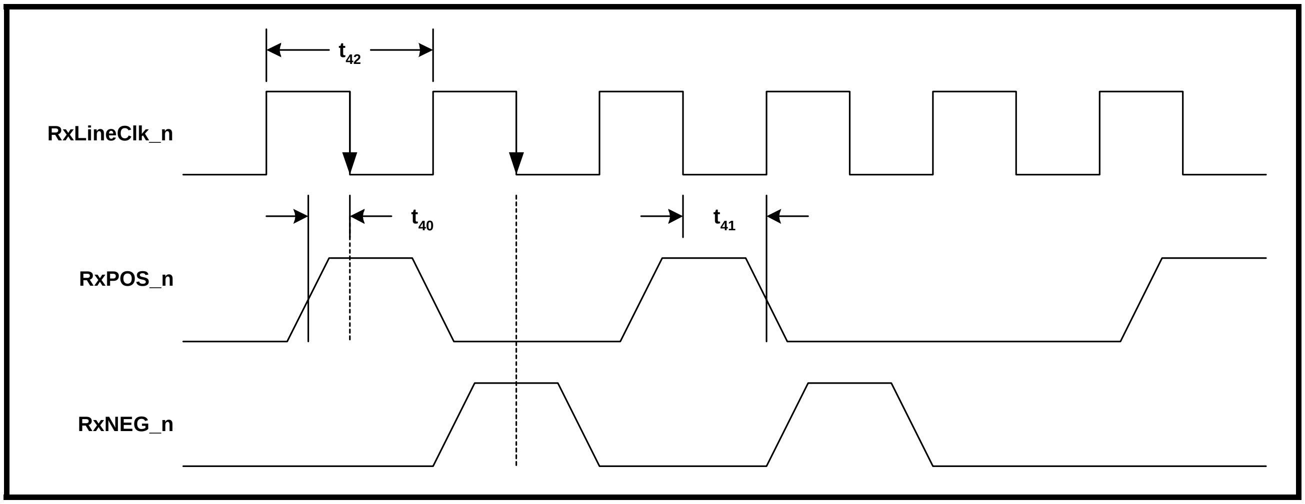


FIGURE 6. RECEIVER LIU INTERFACE TIMING - RXPOS & RXNEG UPDATED ON FALLING EDGE OF RXLINECLK



AC ELECTRICAL CHARACTERISTICS

Test Conditions: TA = 25°C, VDD = 3.3V ± 5% unless otherwise specified						
SYMBOL	**PARAMETER**	**MIN.**	**TYP.**	**MAX.**	**UNITS**	**CONDITIONS**
Microprocessor Interface - Intel (See Figure 7)						
t_{64}	A11 - A0 Setup Time to ALE_AS Low	4			ns	
t_{65}	A11 - A0 Hold Time from ALE_AS Low.	2			ns	
Intel Type Read Operations (See Figure 7 and Figure 9)						
t_{66}	$\overline{RD}$_DS, $\overline{WR}$_R/$\overline{W}$ Pulse Width	60			ns	
t_{67}	Data Valid from $\overline{RD}$_DS Low.	6		11	ns	
t_{68}	Data Bus Floating from $\overline{RD}$_DS High	7		12	ns	
t_{69}	ALE to $\overline{RD}$ Time	4			ns	
t_{701}	$\overline{RD}$ Time to NOT READY (e.g., $\overline{RDY}$_$\overline{DTCK}$ toggling "Low")			6	ns	
t_{70}	$\overline{RD}$ to READY Time (e.g., $\overline{RDY}$_$\overline{DTCK}$ toggling "High")	15		70	ns	
t_{76}	Minimum Time between Read Burst Access (e.g., the rising edge of $\overline{RD}$ to falling edge of $\overline{RD}$)	30			ns	
Intel Type Write Operations (Figure 8 and Figure 10)						
t_{71}	Data Setup Time to $\overline{WR}$_R/$\overline{W}$ High	4			ns	
t_{72}	Data Hold Time from $\overline{WR}$_R/$\overline{W}$ High	2			ns	
t_{73}	High Time between Reads and/or Writes	30			ns	
t_{74}	ALE to $\overline{WR}$ Time	4			ns	

AC ELECTRICAL CHARACTERISTICS

Test Conditions: TA = 25°C, VDD = 3.3V ± 5% unless otherwise specified						
Symbol	**Parameter**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
t_{77}	Minimum Time between Write Burst Access (e.g., the rising edge of $\overline{WR}$ to the falling edge of $\overline{WR}$)	30			ns	
t_{770}	$\overline{CS}$ Assertion to falling edge of $\overline{WR}$_R/$\overline{W}$	20			ns	
Microprocessor Interface - Motorola Read Operations (See Figure 11)						
t_{78}	A11 - A0 Setup Time to falling edge of ALE_AS	5			ns	
t_{79}	Rising edge of $\overline{RD}$_DS to rising edge of $\overline{RDY}$_$\overline{DTCK}$ delay	0			ns	
t_{80}	Rising edge of $\overline{RDY}$_$\overline{DTCK}$ to tri-state of D[7:0]	0			ns	
Microprocessor Interface - Motorola Read & Write Operations (See Figure 11 and Figure 12)						
t_{78}	A8 - A0 Setup Time to falling edge of ALE_AS	5			ns	
t_{81}	D[7:0] Set-up time to falling edge of $\overline{RD}$_DS	10			ns	
t_{82}	Rising edge of $\overline{RD}$_DS to rising edge of $\overline{RDY}$_$\overline{DTCK}$ delay	0			ns	
Reset Pulse Width - Both Motorola and Intel Operations (See Figure 13)						
t_{90}	$\overline{Reset}$ pulse width	200			ns	

FIGURE 7. MICROPROCESSOR INTERFACE TIMING - INTEL TYPE PROGRAMMED I/O READ OPERATIONS

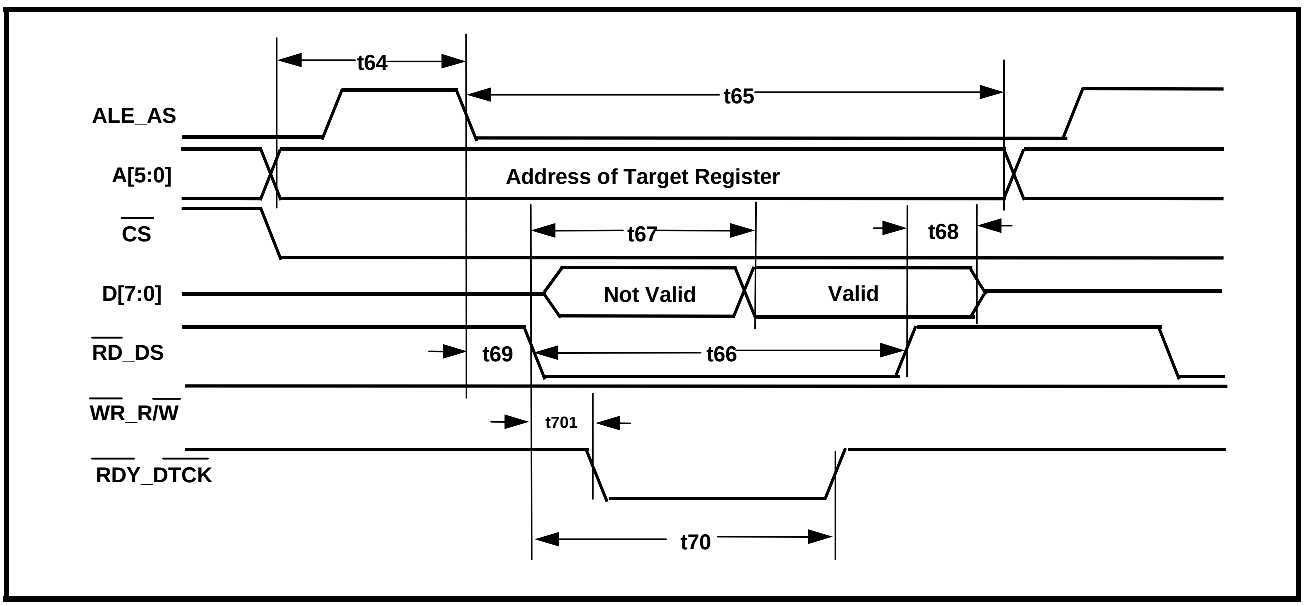


FIGURE 8. MICROPROCESSOR INTERFACE TIMING - INTEL TYPE PROGRAMMED I/O WRITE OPERATIONS

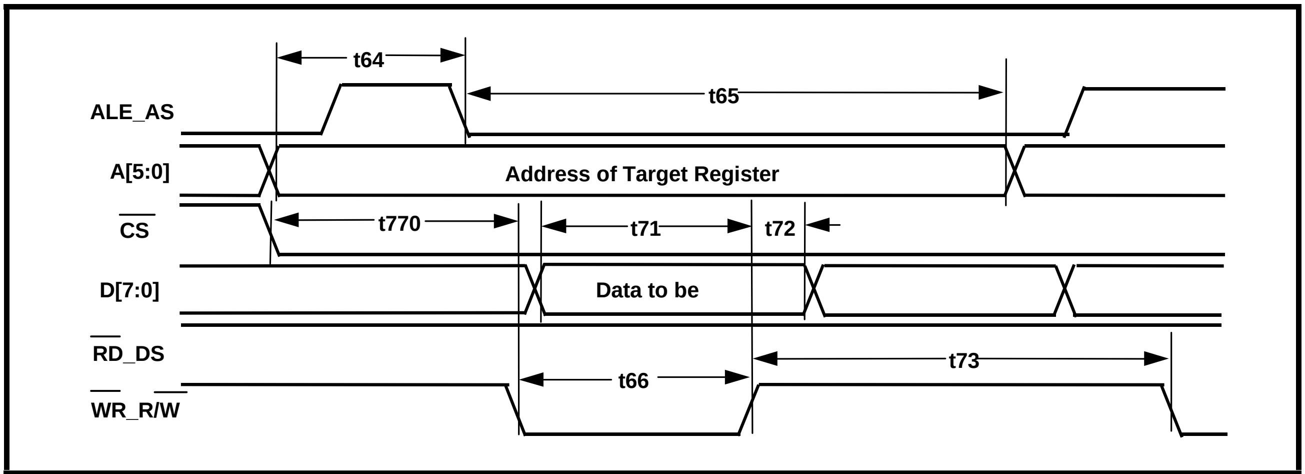


FIGURE 9. MICROPROCESSOR INTERFACE TIMING - INTEL TYPE READ BURST ACCESS OPERATION

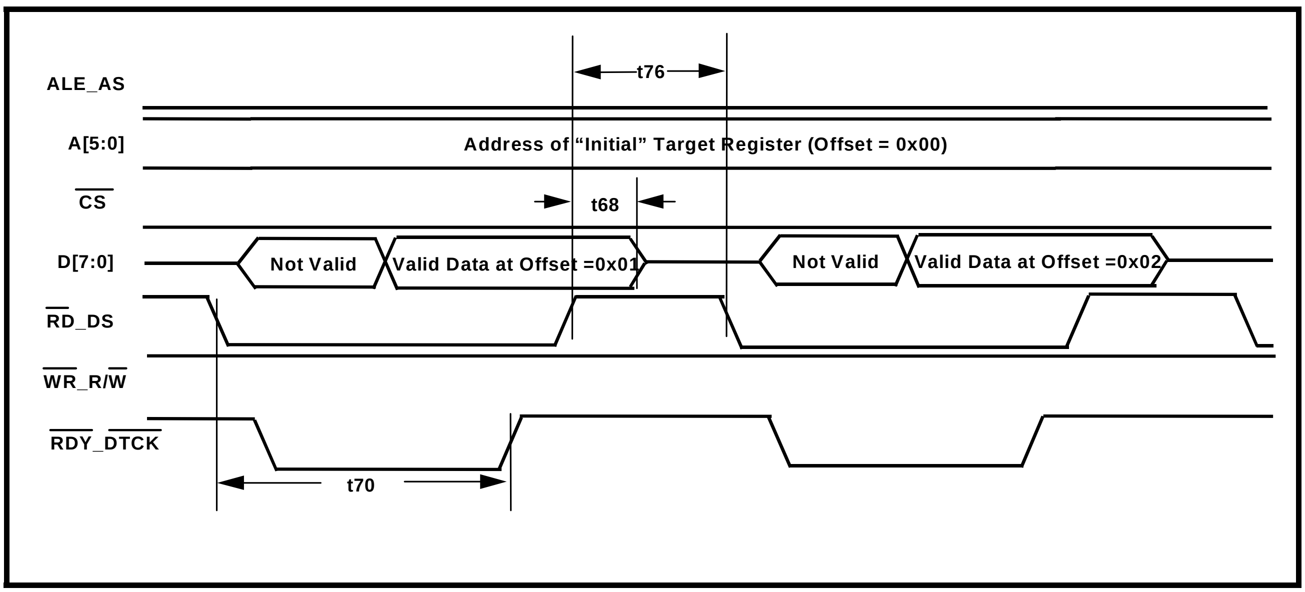


FIGURE 10. MICROPROCESSOR INTERFACE TIMING - INTEL TYPE WRITE BURST ACCESS OPERATION

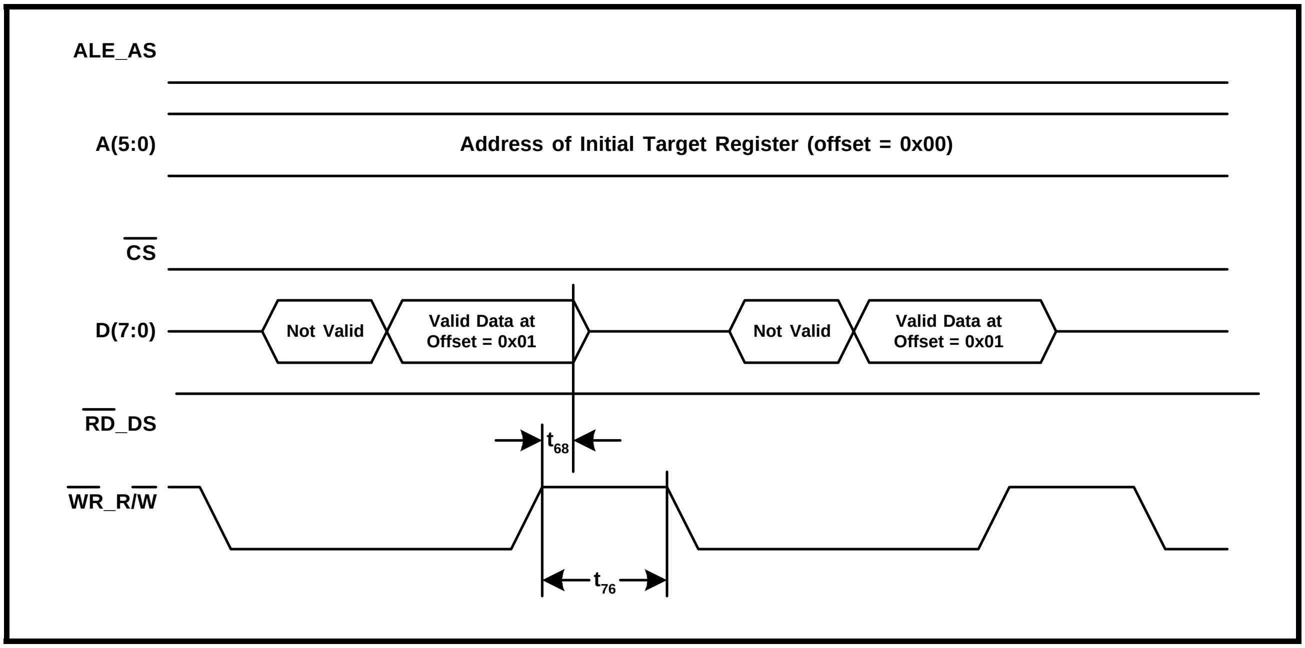


FIGURE 11. MICROPROCESSOR INTERFACE TIMING - MOTOROLA TYPE PROGRAMMED I/O READ OPERATION

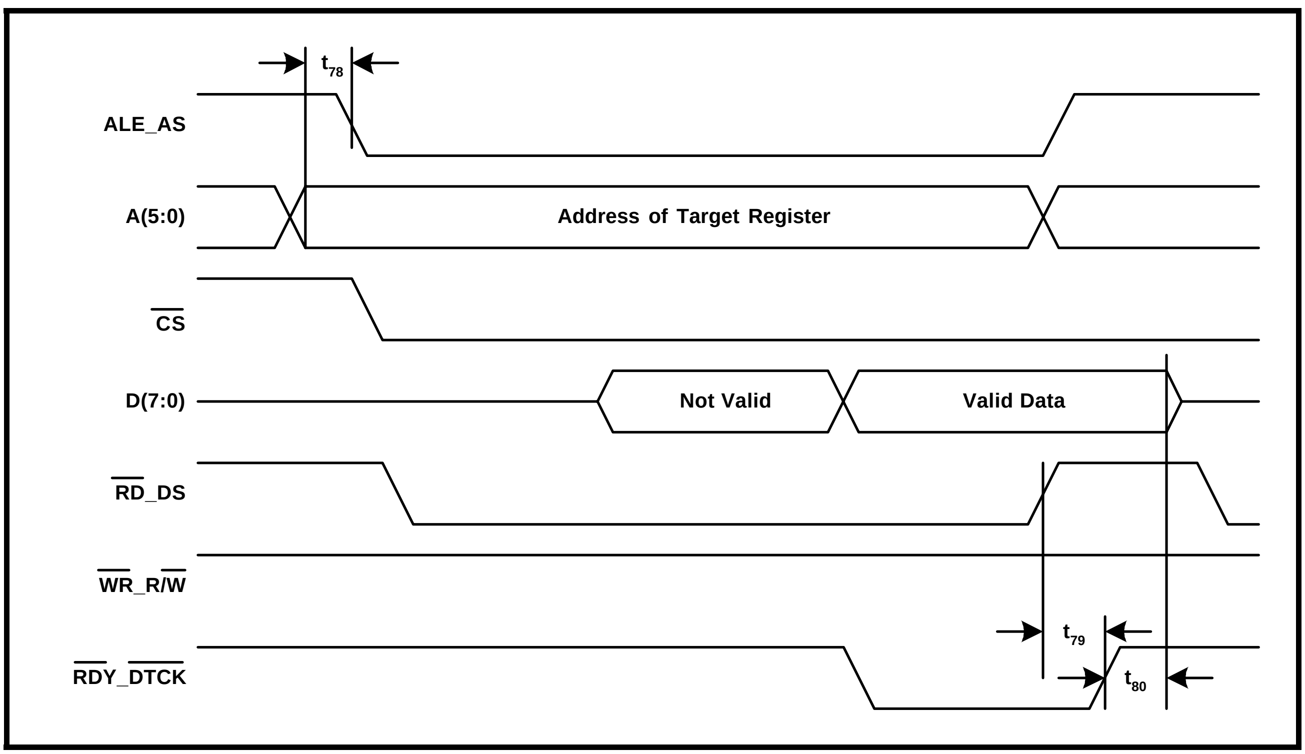


FIGURE 12. MICROPROCESSOR INTERFACE TIMING - MOTOROLA TYPE PROGRAMMED I/O WRITE OPERATION

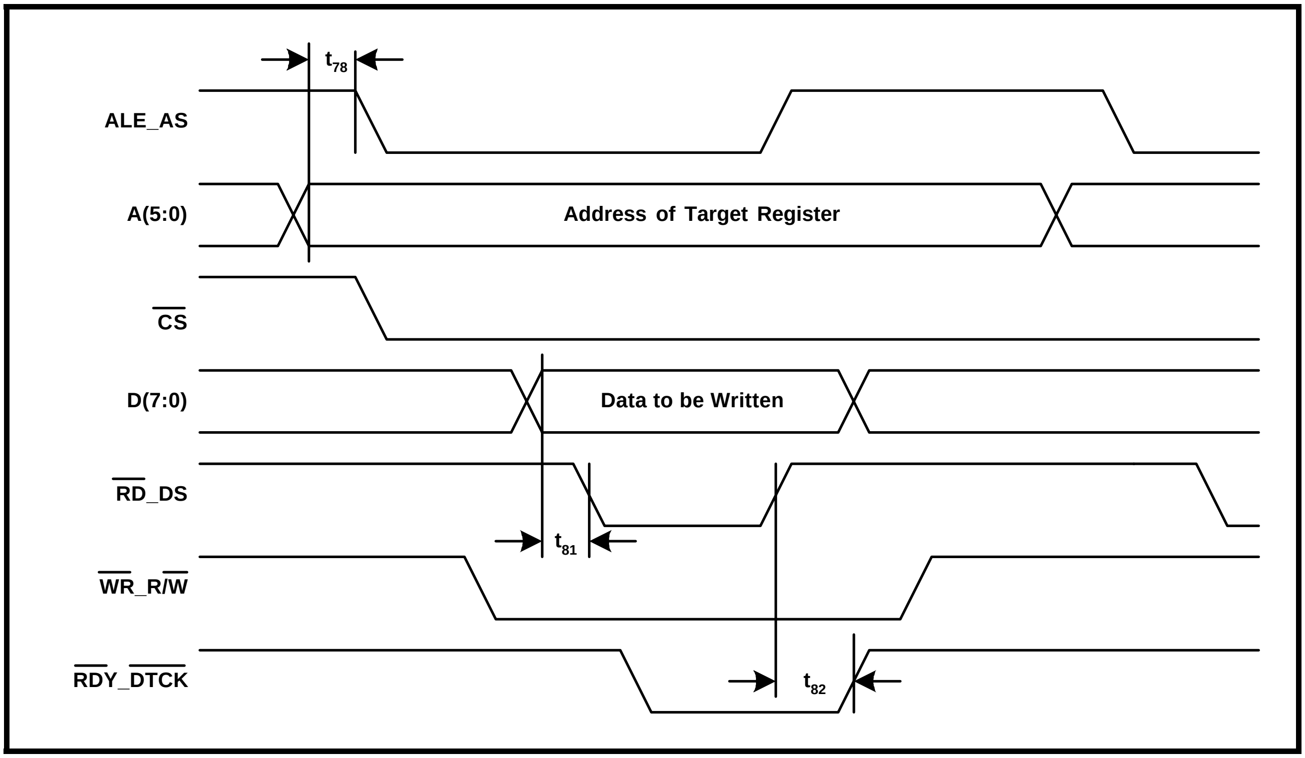
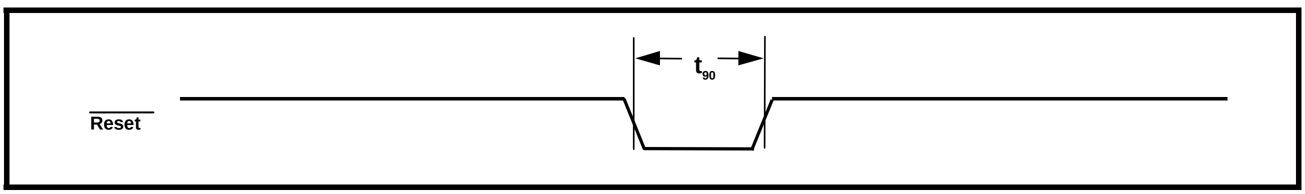


FIGURE 13. MICROPROCESSOR INTERFACE TIMING - RESET PULSE WIDTH



SYSTEM DESCRIPTION

1.0 SYSTEM DESCRIPTION

The XRT84V24 Quad E1 Framer IC consists of four fully functional bidirectional communication channels. Each of these channels operate at 2.048Mbps and are each independent of each other. Each of these channels consists of the following functional blocks.

- Transmit Payload Data Input Interface
- Transmit E1 Framer and Slip Buffer
- Transmit E1 LIU Encoder and Interface
- Transmit HDLC Controller
- Transmit E1 Overhead Data Input Interface
- Receive E1 Decoder and LIU Interface
- Receive E1 Overhead Data Input Interface
- Receive E1 Framer and Slip Buffer
- Receive HDLC Controller
- Receive E1 Payload Data Output Interface
- Receive E1 Overhead Output Interface

These four channels also share the following common resources:

- Microprocessor Interface
- LIU Controller block
- Two 96 Byte “Transmit HDLC Buffers
- Two 96 Byte “Receive HDLC Buffers

Each of these functional blocks will be discussed in detail, in this data sheet. A functional block diagram of the XRT84V24 Quad E1 Framer IC is presented in Figure 14.

The XRT84V24 Quad E1 Framer is designed to support framing and the processing of facility data link messages over four independent E1 lines. Each of the four framers, within this chip, contains the ability to transmit (e.g., map timeslot or clear-channel data into an outbound E1 frame) and receive (e.g., acquire frame alignment and extract timeslot or clear-channel data from the incoming E1 frames) E1 frames in a full-duplex manner.

FOUR FULLY INDEPENDENT RECEIVE SECTIONS:

- Accept either dual-rail or single-rail PCM inputs from an E1 Line Interface Unit IC.
- Decodes the inbound HDB3 or AMI encoded data
- Acquires and maintains FAS Frame Alignment.
- Acquires and maintains CAS Multiframe Alignment, when enabled.
- Acquires and maintains CRC Multiframe Alignment, when enabled.
- Detects and Declares Loss of Signal (LOS).
- Detects and Declares Loss of Framing (LOF) Alignment.
- Detects and Declares Loss of CAS Multiframe Alignment
- Detects and Declares Loss of CRC Multiframe Alignment.
- Detects and Flags Line Code Violations
- Supports line and path performance monitoring according to ITU-T Recommendations. Accumulators are provided for:
  - Line Code Violations
  - Frame Alignment Bit Errors
  - Far-End Block Errors
  - CRC-4 Block Errors
- Indicates the reception of Remote Yellow Alarm (e.g., FAS Yellow Alarm) and Multiframe Yellow Alarm (e.g., CRC Multiframe Yellow Alarm).
- Indicates the reception of AIS and Timeslot-16 AIS.
- Provides an HDLC/LAPD Controller for terminating a data link. Supports polled, interrupt-driven, or DMA servicing of the HDLC Interface.
- Optionally extracts the data link from timeslot 16 (64kbps), which may be used to receive Common-Channel signaling, or from any combination of the national bits in timeslot 0 of non-FAS frames.
- Includes a “Receive Overhead Output Interface” to permit “Terminal Data Link” Equipment direct access to the “National Bits” carrying Data Link information.
- Supports fractional E1 extraction.
- Provides a two-frame elastic (or slip) buffers to handle timing differences between the remote terminal equipment and local terminal equipment clock domains.

THE FOUR FULLY INDEPENDENT TRANSMIT SECTIONS:

- Formats data to create an ITU-T G.704 compliant 2.048Mbps signal.
- Optionally inserts CAS Multiframe Alignment signal.
- Optionally inserts CRC Multiframe Alignment structure including the optional transmission of Far-End Block Errors.
- Provides CAS insertion, programmable Idle Code substitution, digital milliwatt code substitution, and data inversion or a per timeslot basis.

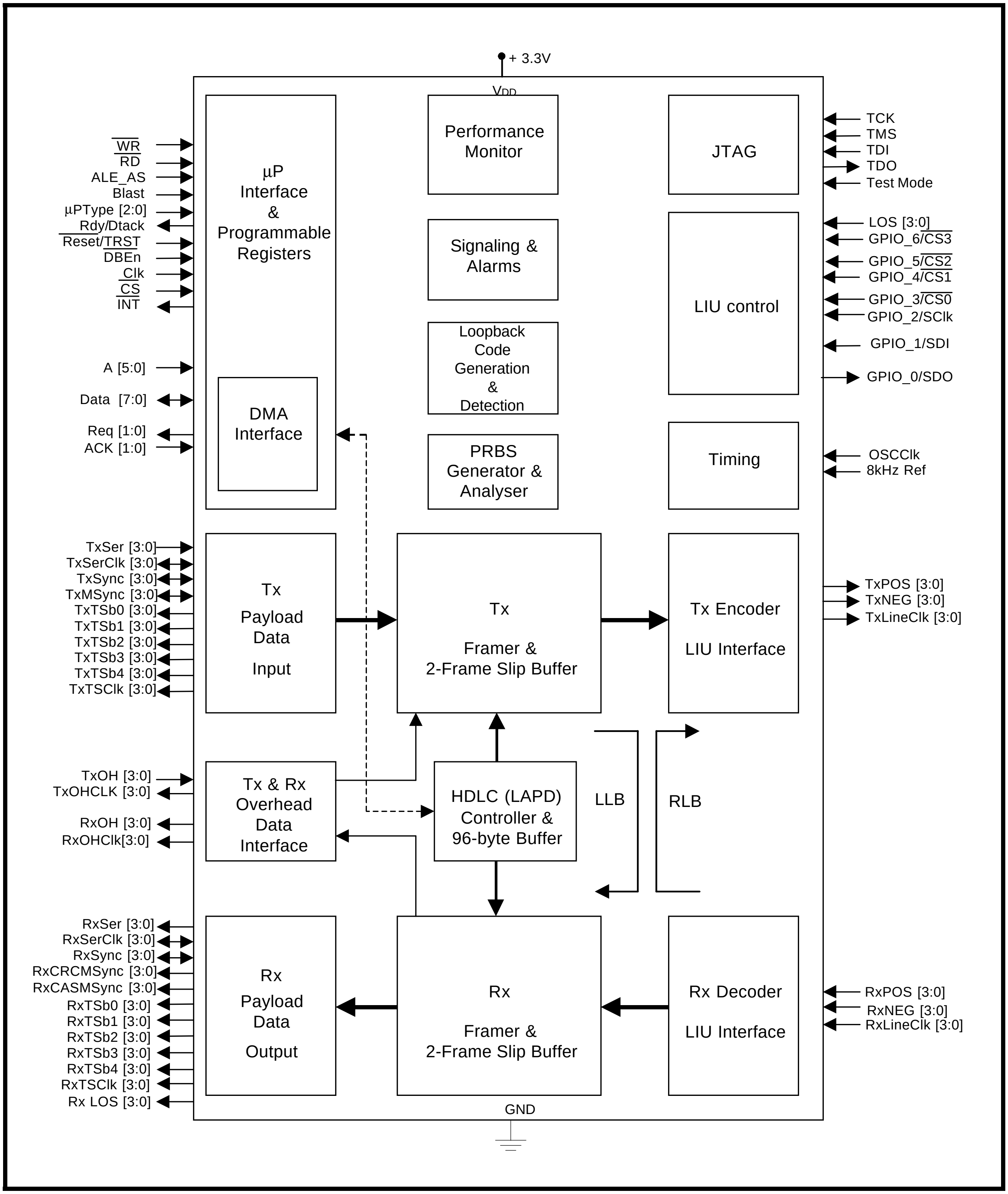
- Provides trunk conditioning which forces programmable trouble code.
- Provides trunk conditioning which forces programmable trouble code substitution and signaling conditions on all timeslots or on selected timeslots, substitution and signaling conditioning on all timeslots or on selected timeslots.
- Supports transmission of AIS, AIS-16 (Timeslot 16 AIS), Remote Yellow Alarm Signal or Remote Multiframe Yellow Alarm Signal.
- Provides an HDLC/LAP-D Interface for transmitting and receiving Data Link information. Supports polled, interrupt-driven or DMA servicing of the HDLC Interface.
- Optionally inserts the data link into timeslot 16 (64kbps) which may be used to transmit Common Channel Signaling, or into any combination of the National Bits in Timeslot 0 of the non-FAS frames (4kbps to 20kbps).
- Includes a “Transmit Overhead Data Input Interface” to permit “Terminal Data Link” equipment direct access to the National Bits carrying Data Link information.
- Supports fractional E1 channel insertion.
- Supports HDB3 or AMI Line Codes.
- Provides Dual-Rail or Single-Rail Output signals.

The XRT84V24 Quad E1 Framer IC also contains the following resources which are shared among the 4 Transmit and Receive Sections.

MICROPROCESSOR INTERFACE

- Easily interfaces to many different types of Microprocessors/Microcontrollers.
  - Intel
  - Motorola
  - MIPS
  - Power PC
- Supports Polled and Interrupt-Driven environments.
- Permits Programmed I/O, Burst and DMA Access.
- Two 96 byte “Transmit HDLC Buffers”, which are shared among the Transmit HDLC Controllers (within each of the 4 Transmit Sections).
- Two 96 byte “Receive HDLC Buffers”, which are shared among the Receive HDLC Controllers (within each of the 4 Receive Sections)

FIGURE 14. BLOCK DIAGRAM OF THE XRT84V24



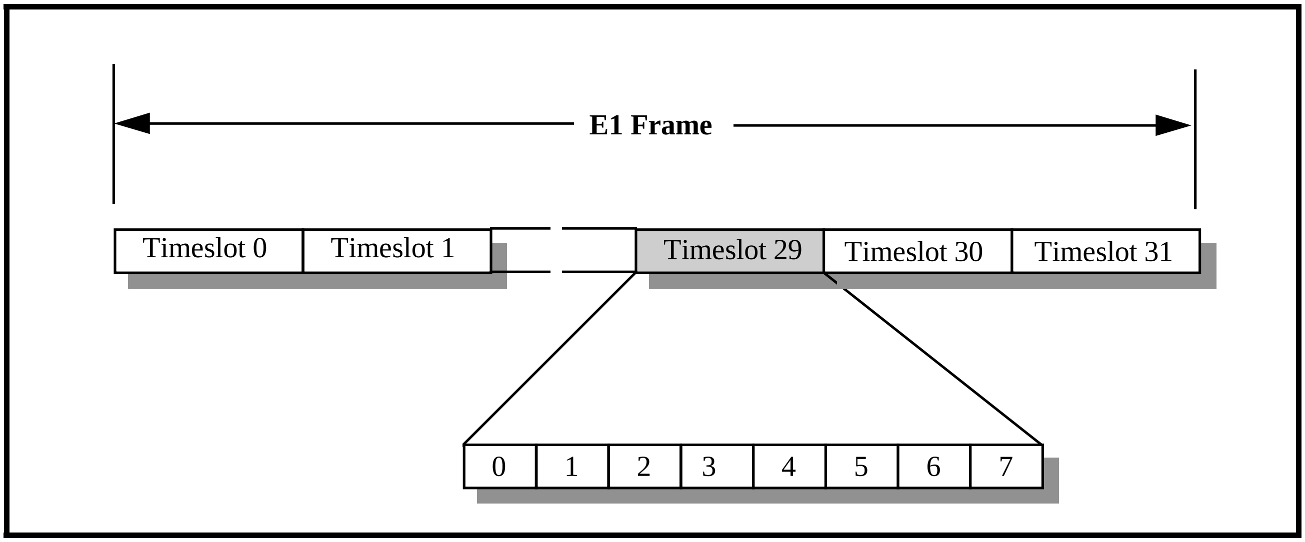
2.0 THE E1 FRAMING STRUCTURE

To discuss the functionality of the Transmit E1 Framer, a brief description of the E1 framing structure is useful.

2.1 The Single E1 Frame

A single E1 frame consists of 256 bits, which is created 8000 times a second; thereby yielding a bit-rate of 2.048Mbps. The 256 bits, within each E1 frame, is grouped into 32 octets (or timeslots). These timeslots are numbered from 0 to 31. Figure 15 presents a simple diagram of a single E1 frame.

Figure 15. Simple Diagram of a Single E1 Frame



A single E1 frame consists of 32 timeslots. However, not all of these timeslots are available to transmit voice or user data. For instance, timeslot 0 is always reserved for system use; and timeslot 16 is sometimes used (reserved) by the system. Hence, within each E1 frame, either 30 or 31 of the 32 timeslots are available for transporting user (or voice) data. The role of timeslot 0 and timeslot 16 will be described in greater detail in the next sections.

Timeslot 0

In general, there are two types of E1 frames.

- FAS (Frame Alignment Signaling) frames
- non-FAS frames

In any E1 data stream, the E1 frame type will alternate between the "FAS" and "non-FAS" frames.

The exact role of the "timeslot 0" octet depends upon which type of E1 frame it is residing in. In general, the "timeslot 0" octet within the "FAS" E1 frame contain a framing alignment pattern and therefore supports framing. The "timeslot 0" octet within the "non-FAS" E1 frame contains bits that support signaling or data link message transmission.

Timeslot 0 octets within FAS frames

The bit-format of a "timeslot 0" octet, within a FAS frame, is presented in the figure below.

Table 1: Bit Format of Timeslot 0 octet within a "FAS" E1 Frame

Bit	7	6	5	4	3	2	1	0
Value	0	0	1	1	0	1	1	SI
Function	FAS Pattern							International Bit
Description-Operation	**Frame Alignment Signaling (FAS) Pattern** The "fixed" framing pattern (e.g., 0, 0, 1, 1, 0, 1, 1) will be used by the Receive E1 Framer, at the Remote terminal for frame synchronization/alignment purposes.							**International Bit** In practice, the Si bit, within the "FAS" E1 Frame carries the results of a CRC-4 calculation, which is discussed in greater detail within the section discussing the CRC Multi-Frame Structure.

The table above indicates that the “FAS” frame “timeslot 0” octet consists of a single “International Bit”, (within bit-field 0) Si, followed by a fixed 7-bit pattern (within bit-fields 1 through 7)

BIT 0—SI (INTERNATIONAL BIT)

The Si bit, within the “FAS” E1 Frame typically carries the results of a CRC-4 calculation, which is discussed in greater detail within the section discussing the CRC Multi-Frame Structure. The “fixed” framing pattern (e.g., 0, 0, 1, 1, 0, 1, 1) will be used by the Receive E1 Framer, at the Remote terminal for frame synchronization/alignment purposes. Section _ discusses how the Receive E1 Framer uses these bits.

Timeslot 0 octets within “non-FAS” frames

The bit-format of a “timeslot 0” octet within a “non-FAS” frame is presented in the table below.

TABLE 2: BIT FORMAT OF TIMESLOT 0 OCTET WITHIN A “NON-FAS” E1 FRAME

BIT	7	6	5	4	3	2	1	0
Value	Sa8	Sa7	Sa6	Sa5	Sa4	A	1	Si
Function6	National bits					Yellow Alarm	Fixed Value	International Bit
Description-Operation	**National Bits** These bit-fields can be used to carry data link information from the Local transmitting terminal to the Remote receiving terminal. Since the National bits only exist in the “non-FAS” frames, they offer a maximum signaling data link bandwidth of 20kbps.					**FAS Frame Yellow Alarm Bit** This bit-field is used to transmit a “Yellow” alarm to the Remote Terminal. This bit-field is set to “0” during normal conditions, and is set to “1” whenever the Receive E1 Framer detects an LOS (Loss of Signal) or LOF (Loss of Framing) condition in the incoming E1 frame data.	**Fixed at “1”** Bit-field “1” contains a fixed value “1”. This bit-field will be used for “FAS framing synchronization/ alignment purposes by the Remote Receive E1 Framer.	**International Bit** The Si bit, within the “non-FAS” E1 Frame typically carries a specific value that will be used by the Receive E1 Framer, for CRC Multi-frame alignment purposes.

The table above indicates the “non-FAS” frame “timeslot 0” octet consists of a single international bit, Si, within bit-field 0.

BIT 0—SI (INTERNATIONAL BIT)

The Si bit, within the “non-FAS” E1 Frame carries a specific value that will be used by the Receive E1 Framer, for CRC Multi-frame alignment purposes. Section 7 discusses the exact role of the Si bit-field within the “non-FAS” frames.

BIT 1—FIXED AT “1”

Bit-field “1” contains a fixed value “1”. This bit-field will be used for “FAS framing synchronization/alignment purposes by the Remote Receive E1 Framer. Section _ discusses how the Receive E1 Framer uses this bit-field.

BIT 2—A (FAS FRAME YELLOW ALARM BIT)

This bit-field is used to transmit a “Yellow” alarm to the Remote Terminal. This bit-field is set to “0” during normal conditions, and is set to “1” whenever the Receive E1 Framer detects an LOS (Loss of Signal) or LOF (Loss of Framing) condition in the incoming E1 frame data.

BIT 3 THROUGH 7—SA4–SA8 (NATIONAL BITS)

These bit-fields can be used to carry data link information from the Local transmitting terminal to the Remote receiving terminal. Since the National bits only exist in the “non-FAS” frames, they offer a maximum signaling data link bandwidth of 20kbps.

2.2 THE E1 MULTI-FRAME STRUCTURES

The XRT84V24 Quad E1 Framer supports two kinds of E1 Multi-frame structures:

- CRC Multi-frame
- CAS Multi-frame

Each of these Multi-Frame structures are described below

2.2.1 The CRC Multi-frame Structure

A CRC Multi-frame consists of 16 consecutive E1 frames, with the first of these frames being a "FAS" frame. From a "Frame Alignment" point of view, the "timeslot 0" octets of each of these E1 frames (within the Multi-frame) are the most important 16 octets within the Multi-frame. Table 3 presents the bit-format for all "timeslot 0" octets within a 16 frame CRC Multi-frame.

TABLE 3: BIT FORMAT OF ALL "TIMESLOT 0" OCTETS WITHIN A CRC MULTI-FRAME

SMF	FRAME NUMBER	BIT 0	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7
1	0	C1	0	0	1	1	0	1	0
	1	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	2	C2	0	0	1	1	0	1	1
	3	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	4	C3	0	0	1	1	0	1	1
	5	1	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	6	C4	0	0	1	1	0	1	1
	7	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
2	8	C1	0	0	1	1	0	1	1
	9	1	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	10	C2	0	0	1	1	0	1	1
	11	1	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	12	C3	0	0	1	1	0	1	1
	13	E	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	14	C4	0	0	1	1	0	1	1
	15	E	1	A	Sa4	Sa5	Sa6	Sa7	Sa8

Table 3 indicates that the CRC Multi-frame can be divided into 2 sub Multi-Frames. Sub-Multi-Frame 1 is designated as "SMF1" and Sub-Multi-Frame 2 is designated as "SMF2".

MF1 consists of E1 frames 0 through 7. Hence, SMF1 consists of 4 "FAS" frames and 4 "non-FAS" frames.

There are two interesting things to note, in Table 3. First, all of the bit-field 0 positions, within each of the "FAS" frames are designated as "C1", "C2", "C3" and "C4". These four bit-fields contain the CRC-4 values which has been computed over the previous SMF. Hence, while the Transmit E1 Framer is assembling a given SMF, it will compute the CRC-4 value for that SMF and will insert these results into the C1 through C4 bit-fields within the very next SMF. These CRC-4 values will ultimately be used by the "Remote" Receive E3 Framer, for error-detection purposes.

***NOTE:** This framing structure is referred to as a "CRC Multi-Frame" because it permits the remote receiving terminal to locate (and in turn, verify) the CRC-4 bit-fields.*

The second interesting thing to note regarding Table 3 is that the bit-field 0 positions, within each of the "non-FAS" frames are of a fixed six (6) bit pattern: 0, 0, 1, 0, 1, 1; along with two bits, each designated at "E". This six bit pattern is referred to as the "CRC Multi-Frame" alignment pattern. This six-bit pattern will ultimately be used by the "Remote" Receive E1 Framer for "CRC Multi-Frame" synchronization/alignment. Section 7.1.1.2 presents a detailed discussion on how the Receive E1 Framer uses this 6-bit CRC Multi-frame alignment pattern for frame synchronization/alignment. The "E" bits are used to indicate that the Local Receive E1 framer has detected errored sub-Multi-Frames.

2.2.2 CAS Multi-Frames and Channel Associated Signaling

CAS Multi-Frames are only relevant if the user is using CAS or "Channel Associated Signaling". If the user is implementing Common Channel Signaling then the CAS Multi-Frame is not available. The exact role of CAS Multi-Frames is discussed in some detail in Section 7.1.1.1 where Channel Associated Signaling is discussed.

2.2.2.1 Channel Associated Signaling

If the user operates an E1 channel in Channel Associated Signaling (CAS) mode, then the "timeslot 16" octets within each E1 frame will be reserved for signaling. Such signaling would convey information such as "On-Hook", "Off-Hook" conditions, call set-up, control, etc. In CAS, this type of signaling data that is associated with a particular voice channel, will be carried within timeslot 16 of a particular E1 frame; within a CAS Multi-Frame.

The CAS is carried in a Multi-Frame structure which consists of 16 consecutive E1 frames. The framing/

byte format of a CAS Multi-Frame is presented below in Figure 16.

FIGURE 16. FRAME/BYTE FORMAT OF THE CAS MULTI-FRAME STRUCTURE

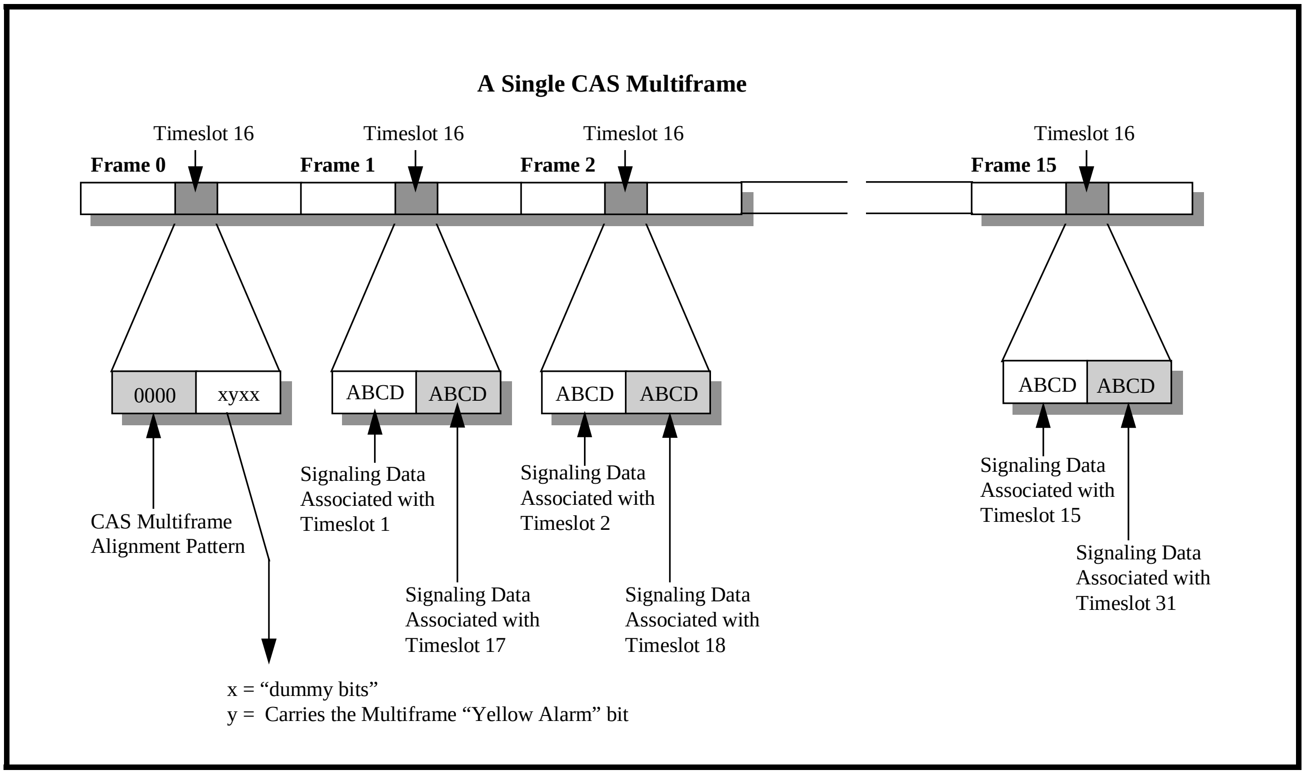


Figure 16 indicates that "timeslot 16", within "frame 1" of the CAS Multi-Frame, contains 4 bits of signaling data for voice channel 1 and 4 bits of signaling data for "voice channel 17". Likewise, "timeslot 16" within "frame 2" contains 4 bits of signaling data for "voice channel 2" and 4 bits of signaling data for "voice channel 18"; and so on. "Timeslot 16", within frame 0 is a special octet that is used for two purposes.

1. To convey CAS Multi-Frame alignment information, and
2. To convey "Multi-Frame" alarm information to the "Remote" Terminal.

The bit-format of timeslot 16, within frame 0 of a CAS Multi-Frame is **0000 xyxx**

The upper nibble of this octet contains all zeros and is used to identify itself as the CAS Multi-Frame alignment signal. If CAS is used, then the user is advised to insure that none of the other "timeslot 16" octets contain the value "0000". The lower nibble of this octet contains the expression "xyxx". In this case, the "x" bits are the spare bits and should be set to "0" if not used. The "y" bit is used to indicate a "Multi-Frame" alarm condition to the Remote terminal. During normal operation, this bit-field is cleared to "0". However, if the "Local" Receive E1 Framer detects a problem with the incoming Multi-Frames, then the Local Transmit E1 Framer will set this bit-field (within the next "outbound" CAS Multi-Frame) to "1".

NOTE: *The "Local" Transmit E1 Framer will continue to set the "y" bit to "1" for the duration that the "Local" Receive E1 Framer detects this problem.*

2.2.2.2 Common Channel Signaling (CCS)

Common Channel Signaling is an alternative form of signaling, from Channel Associated Signaling. In CCS, whatever signaling data which is transported via the "outbound" E1 data stream, carries information that applies to all of the voice channels as a set (e.g., timeslots 1 through 15 and 17 through 31) in the E1 frame. There are numerous other variations of Common Channel Signaling that are available. Some of these are listed below.

- 31 Voice Channels, with the common channel signaling being transported via the National Bits.
- 30 Voice Channels (with the common channel signaling data being transported via the National Bits) and CAS data being transported via timeslot 16.
- 30 Voice Channels, with the Common Channel Signaling being processed via timeslot 16. (e.g., Primary Rate ISDN Signaling).

A more detailed discussion of these forms of Common Channel signaling are discussed in Section 7.

3.0 MICROPROCESSOR INTERFACE BLOCK

The Microprocessor Interface section supports communication between the local microprocessor (µP) and the Framer. The Microprocessor Interface supports the following features.

- Communicates through a 6 bit address bus (4 bit for one framer) and an 8 bit data bus.
- Supports DMA read/write data interface
- Supports burst transfers
- Supports Programmed I/O read and write, wait cycle extended with READY/$\overline{\text{DTCK}}$

The Microprocessor Interface section supports the following operations:

- Channel Selection
- Writing configuration data into the Framer on-chip (addressable) registers
- Writing outbound PMDL (Path Maintenance Data Link) messages into the Transmit LAPD Message buffer of the Framer
- Generation of Interrupt Requests to the µP
- Servicing Interrupt Requests from the Framer
- Monitoring the system's health by periodically reading the on-chip Performance Monitor registers
- Reading inbound PMDL Messages from the Receive LAPD Message Buffer of the Framer

Each of these operations (between the local microprocessor and the Framer IC) is discussed in detail, throughout this data sheet.

The Quad E1 Framer supports the following microprocessors/microcontrollers with a minimum amount of glue logic.

- Intel 8051, 80C188, x86, i960
- Motorola 68HC11, 68K
- MIPS 3051/52
- PowerPC 403

The type of microprocessor/microcontroller to interface to the Quad E1 Framer is specified by tying the µPTYPE[2:0] pins to the appropriate level. Table 1, lists the values for µPTYPE[2:0] and the corresponding µP/µC types.

TABLE 4: µC/µP SELECTION TABLE

µPTYPE[2:0] INPUT LEVELS	CORRESPONDING µC/µP
000	68HC11, 8051, 80C188
001	Motorola 68000 Family
010	Intel x86 Family
011	Intel I960
100	IDT3051/52 (MIPS)
101	IBM PowerPC 403

The behavior of some of the pins, associated with the Microprocessor Interface, depends upon the value that the user has applied to the PType[2:0] input pins. The next sections present a detailed discussion on the role of each of these pins, and how to configure the Quad E1 Framer to interface to each of these types of Microprocessors.

The Quad E1 Framer connects to the Microcontroller as if it were external memory. The microcontroller can read or write to two different storage elements in the E1 Framer:

- Flip-flop types of registers
- RAMs

The configuration of the Quad E1 Framer, including the enabling/disabling of interrupts, is selected by setting values in various control registers. The registers can be read as well as written. The Quad E1 Framer can be designed into both polled and interrupt-driven systems. All detection of change of state of alarm conditions, data link events, error events, or counter overflows can be programmed to cause interrupts.

The Microcontroller Interface Block within the Quad E1 Framer supports three types of data transfer schemes:

- Programmable Input/Output (PIO)
- Burst Transfer
- DMA (Direct Memory Access)

Each of these data transfer methods are also discussed in the next sections.

3.1 CHANNEL SELECTION WITHIN THE FRAMER

The XRT84V24 Quad E1 Framer consists of four independent banks of configuration registers. Each of these banks are identical and correspond to each of the four channels within the XRT84V24. The XRT84V24 permits selection of and access to, any one of these Configuration Register Banks, via the two (2) Most Significant Address Pins, A4 and A5. The relationship between the states of A4 and A5, and the corresponding "Configuration Register" bank, is shown below.

TABLE 5: CHANNEL SELECTION

A5	A4	CONFIGURATION REGISTER BANK
0	0	Channel 0
0	1	Channel 1
1	0	Channel 2
1	1	Channel 3

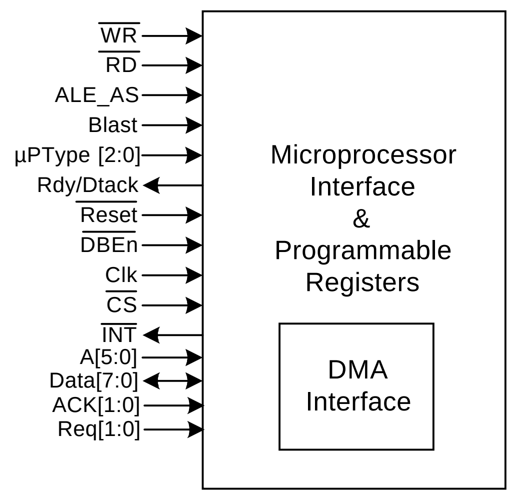
The remaining Address Bus pins [A3 through A0] are used to select the individual configuration registers (within the selected configuration register bank) for Read/Write access.

Each of the four (4) Configuration Register Banks, within the E1 Framer has an identical set of configuration registers. However, address pins A4 and A5 impose the following address location offset, for each of the Configuration Register Bank within the address space of the XRT84V24.

CONFIGURATION REGISTER BANK - CHANNEL NUMBER	ADDRESS OFFSET (WITHIN THE XRT84V24 ADDRESS SPACE)
0	0x000
1	0x200
2	0x400
3	0x600

Figure 17 presents a simple block diagram of the Microprocessor Interface Section, within the Framer.

FIGURE 17. SIMPLIFIED BLOCK DIAGRAM OF THE MICROPROCESSOR INTERFACE BLOCK



3.2 THE MICROPROCESSOR INTERFACE BLOCK SIGNAL

The Quad Framer may be configured into different operating modes and have its performance monitored by software through a standard (local housekeeping) microprocessor, using data, address and control signals.

The local µP configures the Framer (into a desired operating mode) by writing data into specific addressable, on-chip Read/Write registers, or on-chip RAM. The microprocessor interface provides the signals which are required for a general purpose microprocessor to read or write data into these registers. The Microprocessor Interface also supports polled and interrupt driven environments. These interface signals are described below in Table 6, Table 7, and Table 8. The microprocessor interface can be configured to operate in the Motorola Mode, the Intel mode, as well as other modes. When the Microprocessor Interface is operating in the Motorola mode, some of the control signals function in a manner required by the Motorola 68000 family of microprocessors. Likewise, when the Microprocessor Interface is operating in the Intel Mode, then these Control Signals function in a manner as required by the Intel 80xx family of microprocessors.

Table 6 lists and describes those Microprocessor Interface signals whose role is constant across the two modes. Table 7 describes the role of some of these signals when the Microprocessor Interface is operating in the **Intel Mode**. Likewise, Table 8 describes the role of these signals when the Microprocessor Interface is operating in the **Motorola Mode**.

TABLE 6: XRT84V24 MICROPROCESSOR INTERFACE SIGNALS THAT EXHIBIT CONSTANT ROLES IN BOTH THE INTEL AND MOTOROLA MODES

PIN NAME	TYPE	DESCRIPTION
µPTYPEe[2:0]	I	**Microprocessor Interface Mode Select Input pins** These three pins are used to specify the "Microprocessor Mode" that the Microprocessor Interface will operate in. The relationship between the state of these three input pins, and the corresponding "Microprocessor Mode" is presented in Table 1.
D[7:0]	I/O	**Bi-Directional Data Bus for register "Read" or "Write" Operations.**
A[5:0]	I	**Six-Bit Address Bus Inputs** The XRT84V24 Quad E1 Framer Microprocessor Interface uses a Multiplexed Address bus. This address bus is provided to permit the user to select an on-chip register or buffer location for "Read/Write" access.
$\overline{\text{CS}}$	I	**Chip Select Input** This "active-low" signal selects the Microprocessor Interface of the XRT84V24 Quad E1 Framer IC and enables "Read/Write" operations with the on-chip registers/buffer locations.
$\overline{\text{Int}}$	O	**Interrupt Request Output** This "active-low" output signal will inform the local Microprocessor that the Quad E1 Framer IC has an interrupt condition that needs servicing.

TABLE 7: INTEL MODE: MICROPROCESSOR INTERFACE SIGNALS

XRT84V24 PIN NAME	INTEL EQUIVALENT PIN	TYPE	DESCRIPTION
ALE_AS	ALE	I	**Address-Latch Enable:** This “active-high” signal is used to latch the contents on the address bus, A[5:0]. The contents of the Address Bus are latched into the A[5:0] inputs on the falling edge of ALE_AS. Additionally, this signal can be used to indicate the start of a burst cycle.
$\overline{\text{RD}}$_DS	RD*	I	**Read Signal:** This “active-low” input functions as the read signal from the local µP. When this signal goes "Low", the UNI Microprocessor Interface will place the contents of the addressed register on the Data Bus pins (D[7:0]). The Data Bus will be "tri-stated" once this input signal returns "High".
$\overline{\text{WR}}$_R/$\overline{\text{W}}$	WR*	I	**Write Signal:** This "active-low" input functions as the write signal from the local µP. The contents of the Data Bus (D[7:0]) will be written into the addressed register (via A[5:0]), on the rising edge of this signal.
RDY_$\overline{\text{DTCK}}$	READY*	O	**Ready Output:** This "active-low" signal is provided by the UNI device, and indicates that the current read or write cycle is to be extended until this signal is asserted. The local µP will typically insert WAIT states until this signal is asserted. This output will toggle "Low" when the device is ready for the next Read or Write cycle.

TABLE 8: MOTOROLA MODE: MICROPROCESSOR INTERFACE SIGNALS

XRT84V24 PIN NAME	MOTOROLA EQUIVALENT PIN	TYPE	DESCRIPTION
ALE_AS	AS*	I	**Address Strobe:** This "active-low" signal is used to latch the contents on the address bus input pins: A[5:0] into the Microprocessor Interface circuitry. The contents of the Address Bus are latched into the UNI device on the rising edge of the ALE_AS signal. This signal can also be used to indicate the start of a burst cycle.
$\overline{RD}$_DS	DS*	I	**Data Strobe:** This signal latches the contents of the bi-directional data bus pins into the Addressed Register (within the UNI) during a Write Cycle.
$\overline{WR}$_R/$\overline{W}$	R/W*	I	**Read/Write Input:** When this pin is "High", it indicates a Read Cycle. When this pin is "Low", it indicates a Write cycle.
RDY_$\overline{DTCK}$	DTACK*	O	**Data Transfer Acknowledge:** The Quad Framer asserts $\overline{DTCK}$ in order to inform the CPU that the present READ or WRITE cycle is nearly complete. The 68000 family of CPUs requires this signal from its peripheral devices, in order to quickly and properly complete a READ or WRITE cycle.

3.3 INTERFACING THE XRT84V24 FRAMER TO THE LOCAL µC/µP VIA THE MICROPROCESSOR INTERFACE BLOCK

The Microprocessor Interface block within the Framer is very flexible and provides the following options to the user.

- Interface the Framer to a µC/µP over an 8-bit wide bi-directional data bus.
- Interface the Framer to an Intel-type or Motorola-type µC/µP.
- Transfer data (between the Framer IC and the µC/µP) via the Programmed I/O or Burst Mode

Each of the options are discussed in detail below. Section 3.3.1 will discussed the issues associated with interfacing the Framer to a µC/µP over an 8-bit bi-directional data bus. Afterwards, Section 3.3.2 will discuss Data Access (e.g., Programmed I/O and Burst) Mode when interfaced to both Motorola-type and Intel-type µC/µP.

3.3.1 Interfacing the Framer to the Microprocessor over an 8 bit wide bi-directional Data Bus

The Framer Microprocessor Interface permits the user to interface it to a µC/µP over an 8-bit wide bi-directional data bus.

3.3.1.1 Interfacing the Framer to the µC/µP over an 8-bit wide bi-directional data bus.

In general, interfacing the Framer to an 8-bit µC/µP is quite straight-forward. This is because most of the registers, within the Framer, are 8-bits wide. Further, in this mode, the µC/µP can read or write data into both even and odd numbered addresses within the Framer address space.

Reading Performance Monitor (PMON) Registers

A possible complication that the user should be aware of (while operating in the 8-bit mode) occurs whenever the µC/µP needs to read the contents of one of the PMON (Performance Monitor) registers.

The Framer consists of the following PMON Registers.

- E1 receive line code (bipolar) violation counter
- E1 receive framing alignment error counter
- E1 receive severely errored frame counter
- E1 receive CRC-4 block error counter
- E1 receive far-end block error counter
- E1 receive slip counter
- E1 receive loss of frame counter
- E1 receive change of frame alignment counter
- E1 receive synchronization bit error counter
- LAPD frame check sequence error counter

Unlike most of the registers within the Framer, the PMON registers are 16-bit registers (or 16-bits wide). Table 13 lists each of these PMON registers as consisting of two 8-bit registers. One of these 8-bit register is labeled MSB (or Most Significant Byte) and the other register is labeled LSB (or Least Significant Byte). When an 8-bit PMON Register is concatenated with its companion 8-bit PMON Register, one obtains the full 16-bit expression within that PMON Register.

The consequence of having these 16-bit registers is that an 8-bit µC/µP will have to perform two consecutive read operations in order to read in the full 16-bit expression contained within a given PMON register. In addition, these PMON Registers are Reset-Upon-

Read registers. More specifically, these PMON Register are Reset-Upon-Read in the sense that, the entire 16-bit contents, within a given PMON Register is reset, as soon as an 8-bit µC/µP reads in either byte of this two-byte (e.g., 16 bit) expression.

Example;

Consider that an 8-bit µC/µP needs to read in the PMON LCV Event Count Register. In order to accomplish this task, the 8-bit µC/µP needs to read in the contents of PMON LCV Event Count Register - MSB (located at Address = 0x50) and the contents of the PMON LCV Event Count Register - LSB (located at Address = 0x51). These two eight-bit registers, when concatenated together, make up the PMON LCV Event Count Register.

If the 8-bit µC/µP reads in the PMON LCV Event Count-LSB register first, then the entire PMON LCV Event Count register will be reset to 0x0000. As a consequence, if the 8-bit µC/µP attempts to read in the PMON LCV Event Count-MSB register in the very next read cycle, it will read in the value 0x00.

The PMON Holding Register

In order to resolve this Reset-Upon-Read problem, the Framer includes a special register, which permits 8-bit µC/µP to read in the full 16-bit contents of these PMON registers. This special register is called the PMON Holding Register and is located at 0x6c within the Framer Address space.

The operation of the PMON Holding register is as follows. Whenever an 8-bit µC/µP reads in one of the bytes (of the 2-byte PMON register), the contents of the unread (e.g., other) byte will be stored in the PMON Holding Register. Therefore, the 8-bit µC/µP must then read in the contents of the PMON Holding Register in the very next read operation.

In Summary: Whenever an 8-bit* µC/µP *needs to read a PMON Register, it must execute the following steps.

Step 1: Read in the contents of a given 8-bit PMON Register (it does not matter whether the µC/µP reads in the MSB or the LSB register).

Step 2: Read in the contents of the PMON Holding Register (located at Address = 0x6c). This register will contain the contents of the other byte.

3.3.2 Data Access Modes

As mentioned earlier, the Microprocessor Interface block supports data transfer between the Framer and the µC/µP (e.g., Read and Write operations) via two modes: the Programmed I/O and the Burst Modes. Each of these Data Access Modes are discussed in detail below.

3.3.2.1 Programmed I/O

Programmed I/O is basically a “handshaking” type of asynchronous bus access, which provides relatively slow single read and write data transfers. The Microprocessor must supply an address value to the Address Bus input pins A[5:0] with each “read” and “write” cycle. Because of the “Indirect Addressing” scheme each PIO reads and write access requires two accesses, as illustrated below.

In the first access, the Microprocessor is specifying two things:

1. Which of the four framer register sets it intends to access.
2. Which group of registers within the “selected” framer’s register sets, the Microprocessor wants to access.

As a slave, the E1 is the target of access generated by a bus master, in our case, the CPU. Slave accesses are accepted by the slave control state machine, then passed to related functional logic. Address is buffered and decoded to address relevant destination. Data is also latch in both write and read directions. PIO operations are enabled by the Chip Select (CS) input signal. E1 framer PIO interface supports pipelined (buffered) writes to increase bus throughput. All internal registers and accessible memory are addressable through 6 bits of address bus.

3.3.2.2 Data Access using Programmed I/O

Programmed I/O is the conventional manner in which a microprocessor exchanges data with a peripheral device. However, it is also the slowest method of data exchange between the Framer and the µC/µP.

The next two sections present detailed information on Programmed I/O Access, when the Framer is operating in the Intel Mode or in the Motorola Mode.

3.3.2.2.1 Intel Mode Programmed I/O Access

If the Framer is interfaced to an Intel-type µC/µP (e.g., the 80x86 family, etc.), then it should be configured to operate in the Intel mode. Intel-type Read and Write operations are described below.

3.3.2.2.1.1 Intel Mode Read Cycle

Whenever an Intel-type µC/µP wishes to read the contents of a register or some location within the Receive LAPD Message buffer or the Receive OAM Cell Buffer, (within the Framer), it should do the following.

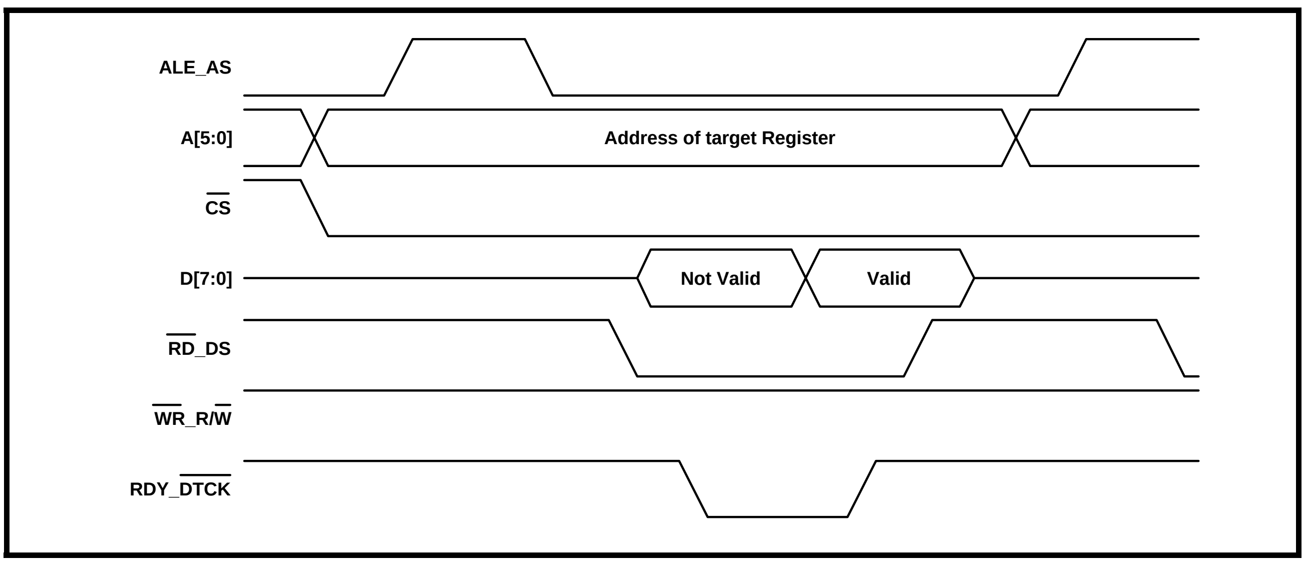
1. Place the address of the target register or buffer location (within the Framer) on the Address Bus input pins A[5:0].
2. While the µC/µP is placing this address value on the Address Bus, the Address Decoding circuitry (within the user's system) should assert the $\overline{CS}$

(Chip Select) pin of the Framer, by toggling it "Low". This action enables further communication between the µC/µP and the Framer Microprocessor Interface block.

3. Toggle the ALE_AS (Address Latch Enable) input pin "High". This step enables the Address Bus input drivers, within the Microprocessor Interface block of the Framer.
4. After allowing the data on the Address Bus pins to settle (by waiting the appropriate Address Data Setup time), the µC/µP should toggle the ALE_AS pin "Low". This step causes the Framer to latch the contents of the Address Bus into its internal circuitry. At this point, the address of the register or buffer locations (within the Framer), has now been selected.
5. Next, the µC/µP should indicate that this current bus cycle is a Read Operation by toggling the RD_DS (Read Strobe) input pin "Low". This action also enables the bi-directional data bus output drivers of the Framer. At this point, the bi-directional data bus output drivers will proceed to drive the contents of the latched addressed register (or buffer location) onto the bi-directional data bus, D[7:0].
6. Immediately after the µC/µP toggles the Read Strobe signal "Low", the Framer will toggle the RDY_DTCK output pin "Low". The Framer does this in order to inform the µC/µP that the data (to be read from the data bus) is NOT READY to be latched into the µC/µP.
7. After some settling time, the data on the bi-directional data bus will stabilize and can be read by the µC/µP. The Framer will indicate that this data can be read by toggling the **RDY**_DTCK (READY) signal "High".
8. After the µC/µP detects the RDY_DTCK signal (from the Framer), it can terminate the Read Cycle by toggling the RD_DS (Read Strobe) input pin "High".

Figure 18 presents a timing diagram which illustrates the behavior of the Microprocessor Interface signals, during an Intel-type Programmed I/O Read Operation.

FIGURE 18. INTEL MICROPROCESSOR INTERFACE SIGNALS DURING PROGRAMMED I/O READ OPERATION



3.3.2.2.1.2 The Intel Mode Write Cycle

Whenever an Intel-type µC/µP wishes to write a byte or word of data into a register or buffer location, within the Framer, it should do the following.

1. Assert the ALE_AS (Address Latch Enable) input pin by toggling it "High". When the µC/µP asserts the ALE_AS input pin, it enables the Address Bus Input Drivers within the Framer chip.
2. Place the address of the target register or buffer location (within the Framer), on the Address Bus input pins, A[5:0].
3. While the µC/µP is placing this address value onto the Address Bus, the Address Decoding circuitry (within the user's system) should assert the CS input pin of the Framer by toggling it "Low". This step enables further communication between the µC/µP and the Framer Microprocessor Interface block.
4. After allowing the data on the Address Bus pins to settle (by waiting the appropriate Address Setup time), the µC/µP should toggle the ALE_AS input pin "Low". This step causes the

Framer to latch the contents of the Address Bus into its internal circuitry. At this point, the address of the register or buffer location (within the Framer), has now been selected.

5. Next, the µC/µP should indicate that this current bus cycle is a Write Operation by toggling the WR_R/W (Write Strobe) input pin "Low". This action also enables the bi-directional data bus input drivers of the Framer.

6. The µC/µP should then place the byte or word that it intends to write into the target register, on the bi-directional data bus, D[7:0].

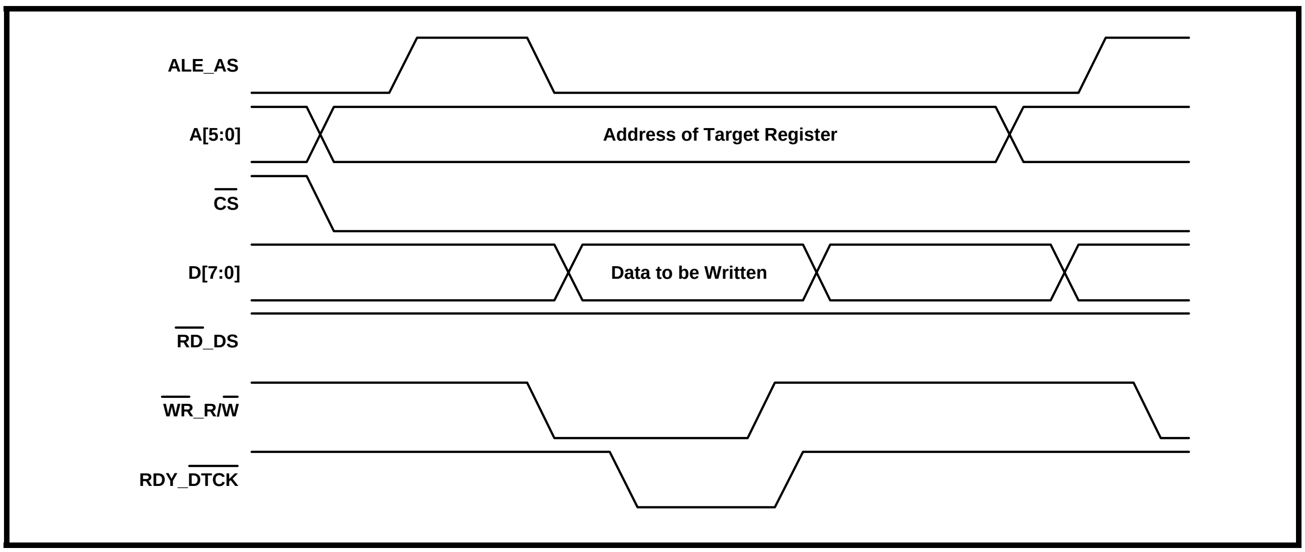
7. After waiting the appropriate amount of time for the data (on the bi-directional data bus) to settle, the µC/µP should toggle the WR_R/W (Write Strobe) input pin "High". This action accomplishes two things:

a. It latches the contents of the bi-directional data bus into the Framer Microprocessor Interface block.

b. It terminates the write cycle.

Figure 19 presents a timing diagram which illustrates the behavior of the Microprocessor Interface signals, during an Intel-type Programmed I/O Write Operation.

FIGURE 19. INTEL MICROPROCESSOR INTERFACE SIGNALS, DURING PROGRAMMED I/O WRITE OPERATION



3.3.2.2.2 Motorola Mode Programmed I/O Access

If the Framer is interfaced to a Motorola-type µC/µP (e.g., the MC680X0 family, etc.), it should be configured to operate in the Motorola mode. Motorola-type Programmed I/O Read and Write operations are described below.

3.3.2.2.2.1 Motorola Mode Read Cycle

Whenever a Motorola-type µC/µP wishes to read the contents of a register or some location within the Receive LAPD Message or Receive OAM Cell Buffer, (within the Framer) it should do the following.

1. Assert the ALE_AS (Address-Strobe) input pin by toggling it low. This step enables the Address Bus input drivers, within the Microprocessor Interface Block of the Framer.

2. Place the address of the target register (or buffer location) within the Framer, on the Address Bus input pins, A[5:0].

3. At the same time, the Address Decoding circuitry (within the user's system) should assert the CS (Chip Select) input pin of the Framer, by toggling it "Low". This action enables further communication between the µC/µP and the Framer Microprocessor Interface block.

4. After allowing the data on the Address Bus pins to settle (by waiting the appropriate Address Setup time), the µC/µP should toggle the ALE_AS input pin "High". This step causes the Framer to latch the contents of the Address Bus into its internal circuitry. At this point, the address of the register or buffer location (within the Framer) has been selected.

5. Further, the µC/µP should indicate that this cycle is a Read cycle by setting the WR_R/W (R/W*) input pin "High".

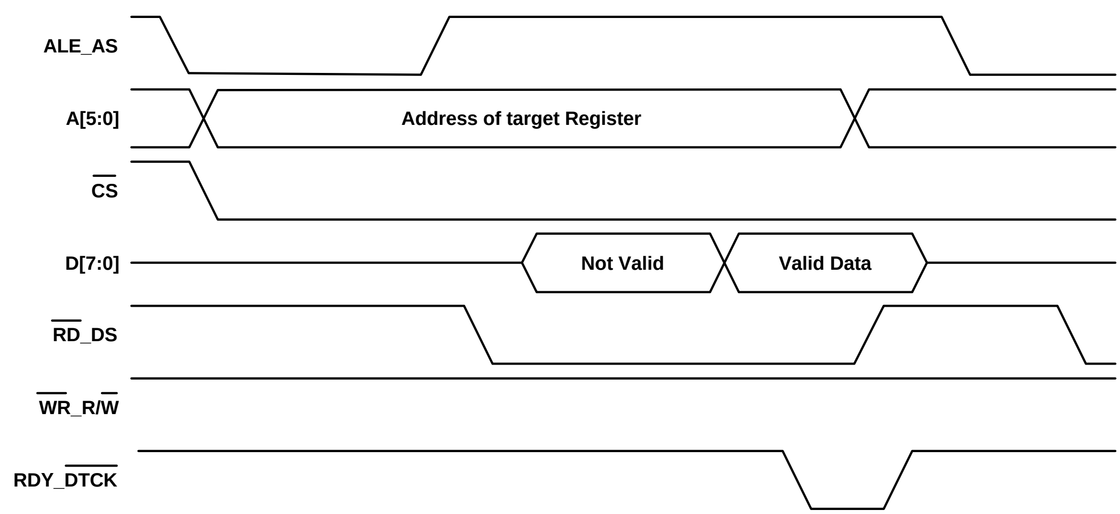
6. Next the µC/µP should initiate the current bus cycle by toggling the RD_DS (Data Strobe) input pin "Low". This step enables the bi-directional data bus output drivers, within the Framer. At this point, the bi-directional data bus output drivers will proceed to drive the contents of the Address register onto the bi-directional data bus, D[7:0].

7. After some settling time, the data on the bi-directional data bus will stabilize and can be read by the µC/µP. The Framer will indicate that this data can be read by asserting the RDY_DTCK (DTACK) signal.

8. After the µC/µP detects the RDY_DTCK signal (from the Framer) it will terminate the Read Cycle by toggling the RD_DS (Data Strobe) input pin "High".

Figure 20 presents a timing diagram which illustrates the behavior of the Microprocessor Interface signals during a Motorola-type Programmed I/O Read Operation.

FIGURE 20. MOTOROLA MICROPROCESSOR INTERFACE SIGNALS, DURING A PROGRAMMED I/O READ OPERATION



3.3.2.2.2.2 Motorola Mode Write Cycle

Whenever a Motorola-type µC/µP wishes to write a byte or word of data into a register or buffer location, within the Framer, it should do the following.

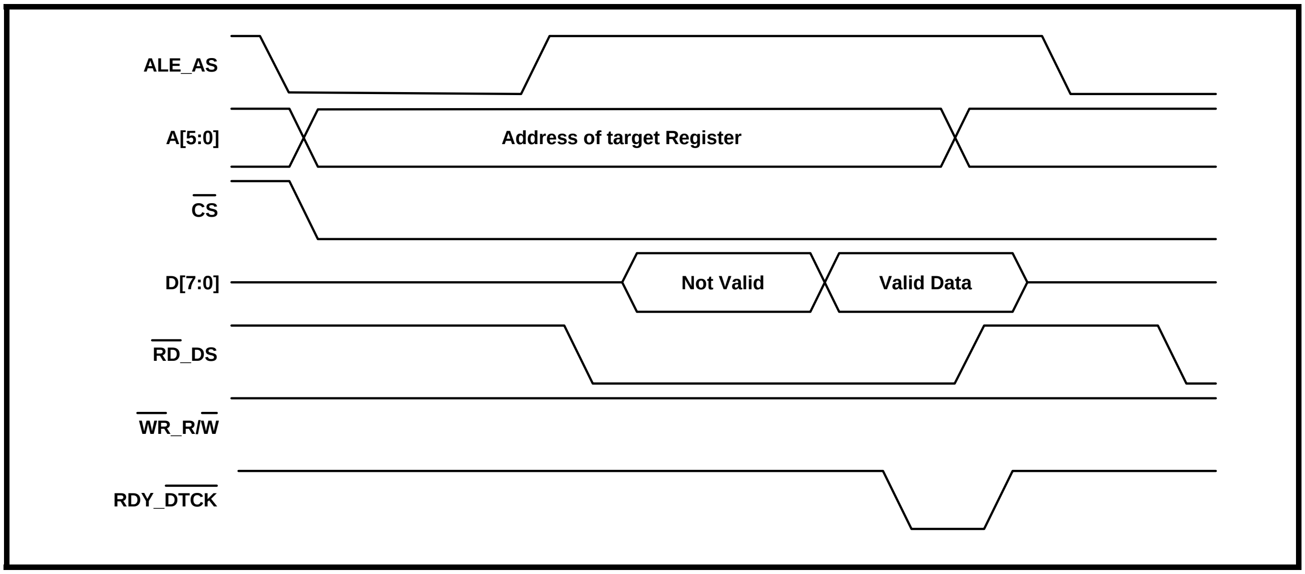
1. Assert the ALE_AS (Address Select) input pin by toggling it "Low". This step enables the Address Bus input drivers (within the Framer chip).
2. Place the address of the target register or buffer location (within the Framer), on the Address Bus input pins, A[5:0].
3. While the µC/µP is placing this address value onto the Address Bus, the Address-Decoding circuitry (within the user's system) should assert the CS (Chip Select) input pins of the Framer by toggling it "Low". This step enables further communication between the µC/µP and the Framer Microprocessor Interface block.
4. After allowing the data on the Address Bus pins to settle (by waiting the appropriate Address Setup time), the µC/µP should toggle the ALE_AS input pin "High". This step causes the Framer to latch the contents of the Address Bus into its own circuitry. At this point, the Address of the register or buffer location (within the Framer), has now been selected.
5. Further, the µC/µP should indicate that this current bus cycle is a Write operation by toggling the WR_R/W (R/W*) input pin "Low".
6. The µC/µP should then place the byte or word that it intends to write into the target register, on the bi-directional data bus, D[7:0].
7. Next, the µC/µP should initiate the bus cycle by toggling the RD_DS (Data Strobe) input pin "Low". When the Framer senses that the WR_R/W (R/W*) input pin is "High" and that the RD_DS (Data Strobe) input pin has toggled "Low", it will enable the input drivers of the bi-directional data bus, D[7:0].
8. After waiting the appropriate time, for this newly placed data to settle on the bi-directional data bus (e.g., the Data Setup time) the Framer will assert the RDY_DTCK output signal.
9. After the µC/µP detects the RDY_DTCK signal (from the Framer), the µC/µP should toggle the RD_DS input pin "High". This action accomplishes two things.

a. It latches the contents of the bi-directional data bus into the Microprocessor Interface block.

b. It terminates the Write cycle.

Figure 21 presents a timing diagram which illustrates the behavior of the Microprocessor Interface signals, during a Motorola-type Programmed I/O Write Operation.

FIGURE 21. MOTOROLA MICROPROCESSOR INTERFACE SIGNAL DURING PROGRAMMED I/O WRITE OPERATION



3.3.2.3 Burst Mode I/O for Data Access

Burst Mode I/O access is a much faster way to transfer data between the µC/µP and the Microprocessor Interface (of the Framer), than Programmed I/O. The reason why Burst Mode I/O is faster is explained below.

Data is placed upon the Address Bus input pins A[5:0] only for the very first access, within a given burst access. The remaining read or write operations (within this burst access) do not require the placement of the Address Data on the Address Data Bus. As a consequence, the user does not have to wait through the Address Setup and Hold times for each of these Read/Write operation, within the Burst Access.

It is important to note that there are some limitations associated with Burst Mode I/O Operations.

1. All cycles within the Burst Access, must be either all Read or all Write cycles. No mixing of Read and Write cycles is permitted.
2. A Burst Access can only be used when Read or Write operations are to be employed over a contiguous range of address locations, within the Framer.
3. The very first Read or Write cycle, within a burst access, must start at the lowest address value, of the range of addresses to be accessed. Subsequent operations will automatically be incremented to the very next higher address value.

Examples of Burst Mode I/O operations are presented below for read and write operations, with both Intel-type and Motorola-type µC/µP.

3.3.2.3.1 Burst I/O Access: Intel Mode

If the XRT84V24 Framer is interfaced to an Intel-type µC/µP (e.g., the 80x86 family, etc.), then it should be configured to operate in the Intel mode (by tying the MOTO pin to ground). Intel-type Read and Write Burst I/O Access operations are described below.

3.3.2.3.1.1 Intel-Mode Read Burst Access

When an Intel-type µC/µP wants to read the contents of numerous registers or buffer locations over a contiguous range of addresses, then it should do the following.

a. Perform the initial read operation of the burst access.

b. Perform the remaining read operations of the burst access.

c. Terminate the burst access operation.

Each of these operations within the burst access are described below.

3.3.2.3.1.1.1 Initial Read Operation: Intel mode

The initial read operation of an Intel-type read burst access is accomplished by executing a Programmed I/O Read Cycle as summarized below.

A.0 Execute a Single Ordinary (Programmed I/O) Read Cycle, as described in steps A.1 through A.7 below.

A.1 Place the address of the initial-target register or buffer location (within the Framer) on the Address Bus input pins A[5:0].

A.2 While the µC/µP is placing this address value onto the Address Bus, the Address Decoding circuitry (within the user's system) should assert the CS input pin of the Framer, by toggling it "Low". This step enables further communication between the µC/µP and the Framer Microprocessor Interface block.

A.3 Assert the ALE_AS (Address Latch Enable) pin by toggling it "High". This step enables the Address Bus input drivers, within the Microprocessor Interface block of the Framer.

A.4 After allowing the data on the Address Bus pins to settle (by waiting the appropriate Address Data Setup time), the µC/µP should then toggle the ALE_AS pin "Low". This step latches the contents, on the Address Bus pins, A[5:0], into the Framer Microprocessor Interface block. At this point, the initial address of the burst access has now been selected.

NOTE: *The ALE_AS input pin should remain "Low" for the remainder of this Burst Access operation.*

A.5 Next, the µC/µP should indicate that this current bus cycle is a Read Operation by toggling the RD_DS (Read Strobe) input pin "Low". This action also enables the bi-directional data bus output drivers of the Framer. At this point, the bi-directional data bus output drivers will proceed to drive the contents of the addressed register onto the bi-directional data bus, D[7:0].

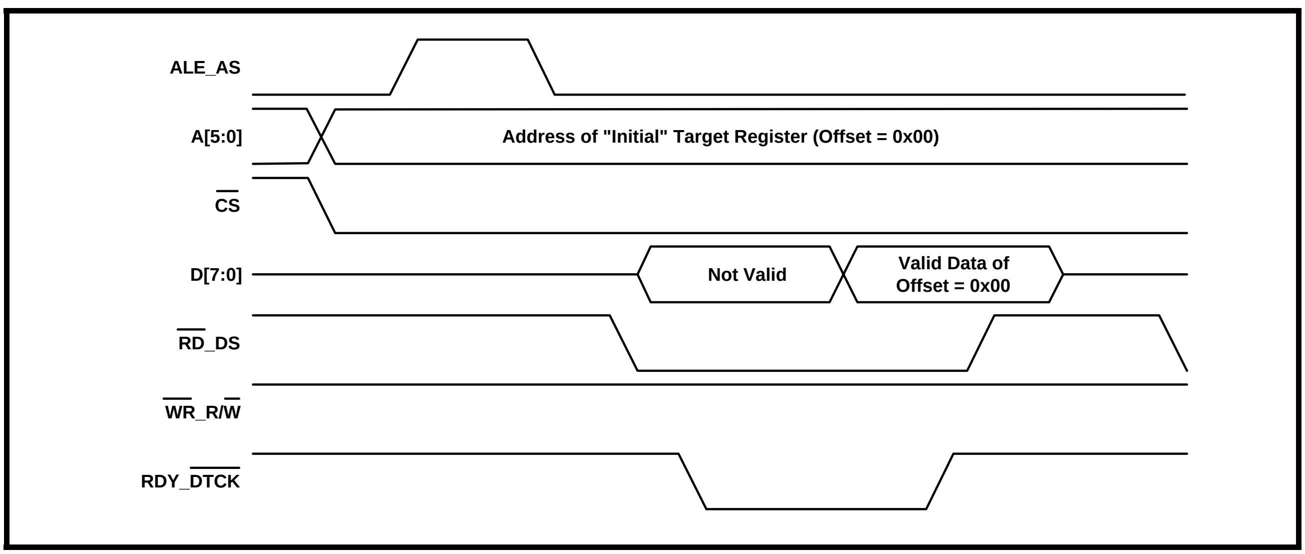
A.6 Immediately after the µC/µP toggles the Read Strobe signal "Low", the Framer will toggle the RDY_DTCK (READY) output pin "Low". The Framer does this in order to inform the µC/µP that the data (to be read from the data bus) is NOT READY to be latched into the µC/µP.

A.7 After some settling time, the data on the bi-directional data bus will stabilize and can be read by the µC/µP. The Framer will indicate that this data is ready to be read, by toggling the RDY_DTCK (Ready) signal "High".

A.8 After the µC/µP detects the RDY_DTCK signal (from the Framer), it can then will terminate the Read cycle by toggling the RD_DS (Read Strobe) input pin "High".

Figure 22 presents an illustration of the behavior of the Microprocessor Interface Signals, during the initial Read Operation, within a Burst I/O Cycle for an Intel-type µC/µP.

FIGURE 22. INTEL MICROPROCESSOR INTERFACE SIGNALS, DURING THE INITIAL READ OPERATION OF A BURST CYCLE



At the completion of this initial read cycle, the µC/µP has read in the contents of the first register or buffer location (within the Framer) for this particular burst I/O access operation. In order to illustrate how this burst access operation works, the byte (or word) of data, that is being read in Figure 22, has been labeled Valid Data at Offset = 0x00. This label indicates that the µC/µP is reading the very first register (or buffer location) in this burst access operation.

3.3.2.3.1.1.2 Subsequent Read Operations

The procedure that the µC/µP must use to perform the remaining read cycles, within this Burst Access operation, is presented below.

B.0 Execute each subsequent Read Cycles, as described in steps 1 through 3 below.

B.1 Without toggling the ALE_AS input pin (e.g., keeping it "Low"), toggle the RD_DS input pin "Low". This step accomplishes the following.

- **a.** The Framer will internally increments the latched address value (within the Microprocessor Interface circuitry).
- **b.** The output drivers of the bi-directional data bus, D[7:0] are enabled. At some time later, the register or buffer location corresponding to the incremented latched address value will be driven onto the bi-directional data bus.

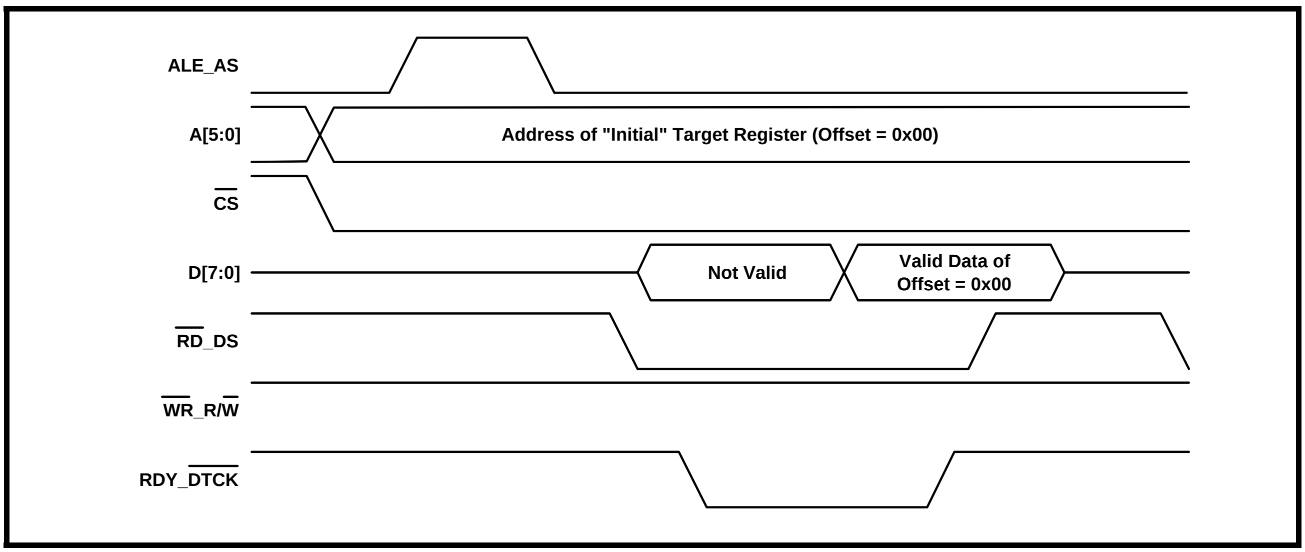
B.2 Immediately after the Read Strobe pin toggles "Low" the Framer will toggle the RDY_DTCK (READY) output pin "Low" to indicate its NOT READY status.

B.3 After some settling time, the data on the bi-directional data bus will stabilize and can be read by the µC/µP. The Framer will indicate that this data is ready to be read by toggling the RDY_DTCK (READY) signal "High".

B.4 After the µC/µP detects the RDY_DTCK signal (from the Framer), it can terminate the Read cycle by toggling the RD_DS (Read Strobe) input pin "High".

For subsequent read operations, within this burst cycle, the µC/µP simply repeats steps 1 through 3, as illustrated in Figure 23.

FIGURE 23. INTEL MICROPROCESSOR INTERFACE SIGNALS, DURING SUBSEQUENT READ OPERATIONS WITHIN THE BURST I/O CYCLE



In addition to the behavior of the Microprocessor Interface signals, Figure 23 also illustrates other points regarding the Burst Access Operation.

- **a.** The Framer internally increments the address value, from the original latched value shown in Figure 22. This is illustrated by the data, appearing on the data bus, (for the first read access) being labeled Valid Data at Offset = 0x01 and that for the second read access being labeled Valid Data at Offset = 0x02.
- **b.** The Framer performs this address incrementing process even though there are no changes in the Address Bus Data, A[5:0].

3.3.2.3.1.1.3 Terminating the Burst Access Operation

The Burst Access Operation will be terminated upon the rising edge of the ALE_AS input signal. At this point the Framer will cease to internally increment the latched address value. Further, the µC/µP is now free to execute either a Programmed I/O access or to start another Burst Access Operation with the Framer.

3.3.2.3.1.2 Write Burst Access: Intel-Mode

When an Intel-type µC/µP wishes to write data into a contiguous range of addresses, then it should do the following.

- **a.** Perform the initial write operation of the burst access.
- **b.** Perform the remaining write operations, of the burst access.
- **c.** Terminate the burst access operation.

Each of these operations within the burst access are described below.

3.3.2.3.1.2.1 Initial Write Operation

The initial write operation of an Intel-type Write Burst Access is accomplished by executing a Programmed I/O write cycle as summarized below.

A.0 Execute a Single Ordinary (Programmed I/O) Write cycle, as described in Steps A.1 through A.7 below.

A.1 Place the address of the initial target register (or buffer location) within the Framer, on the Address Bus pins, A[5:0].

A.2 At the same time, the Address-Decoding circuitry (within the user's system) should assert the CS (Chip Select) input pin of the Framer, by toggling it "Low". This step enables further communication between the µC/µP and the Framer Microprocessor Interface block.

A.3 Assert the ALE_AS (Address Latch Enable) input pin "High". This step enables the Address Bus input drivers, within the Microprocessor Interface Block of the Framer.

A.4 After allowing the data on the Address Bus pins to settle (by waiting the appropriate Address Setup time), the µC/µP should then toggle the ALE_AS input pin "Low". This step latches the contents, on the Address Bus pins, A[5:0], into the XRT84V24 Framer Microprocessor Interface block. At this point, the initial address of the burst access has now been selected.

NOTE: The ALE_AS input pin should remain "Low" for the remainder of this Burst I/O Access operation.

A.5 Next, the µC/µP should indicate that this current bus cycle is a Write operation by keeping the RD_DS pin "High" and toggling the WR_R/W (Write Strobe) pin "Low". This action also enables the bi-directional data bus input drivers of the Framer.

A.6 The µC/µP places the byte (or word) that it intends to write into the target register on the bi-directional data bus, D[7:0].

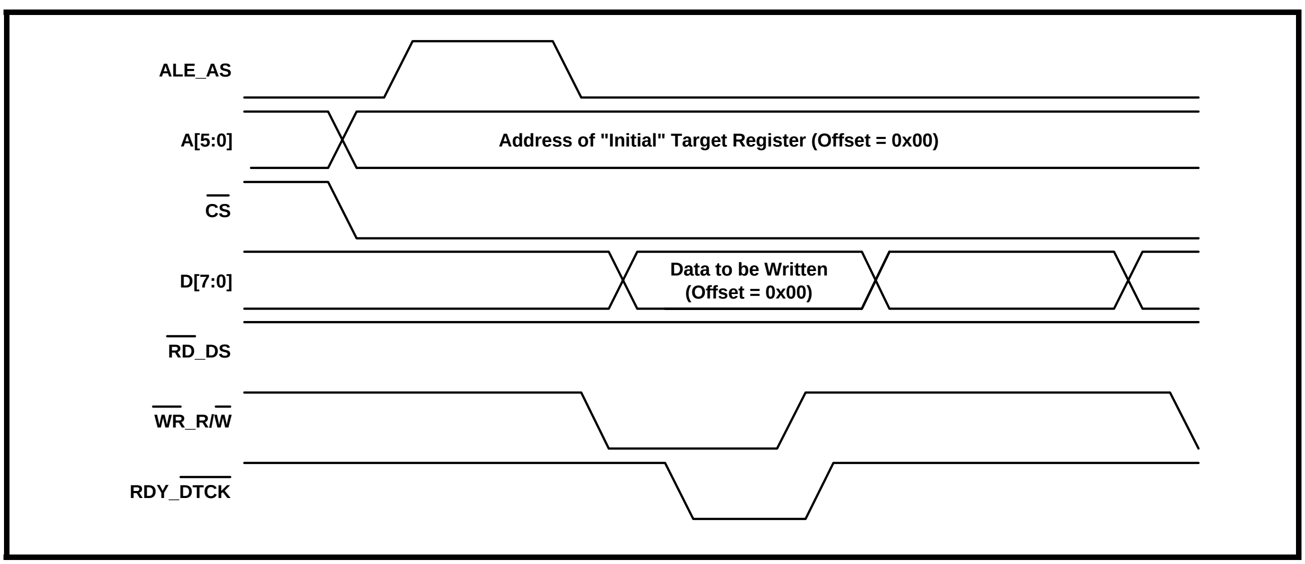
A.7 After waiting the appropriate amount of time, for the data (on the bi-directional data bus) to settle, the µC/µP should toggle the WR_R/W (Write Strobe) input pin "High". This action accomplishes two things.

a. It latches the contents of the bi-directional data bus into the Framer Microprocessor Interface Block.

b. It terminates the write cycle.

Figure 24 presents a timing diagram which illustrates the behavior of the Microprocessor Interface signals, during the initial write operation within a Burst Access, for an Intel-type µC/µP.

FIGURE 24. INTEL MICROPROCESSOR INTERFACE SIGNALS, DURING THE INITIAL WRITE OPERATION OF A BURST CYCLE



At the completion of this initial write cycle, the µC/µP has written a byte or word into the first register or buffer location (within the Framer) for this particular burst access operation. In order to illustrate this point, the byte (or word) of data, that is being written in Figure 24 has been labeled Data to be Written (Offset = 0x00).

3.3.2.3.1.2.2 The Subsequent Write Operations

The procedure that the µC/µP must use to perform the remaining write cycles, within this burst access operation, is presented below.

B.0 Execute each subsequent write cycle, as described in steps B.1 through B.3.

B.1 Without toggling the ALE_AS input pin (e.g., keeping it "Low"), apply the value of the next byte or word (to be written into the Framer) to the bi-directional data bus pins, D[7:0].

B.2 Toggle the $\overline{\text{WR}}$_R/$\overline{\text{W}}$ (Write Strobe) input pin "Low". This step accomplishes two things.

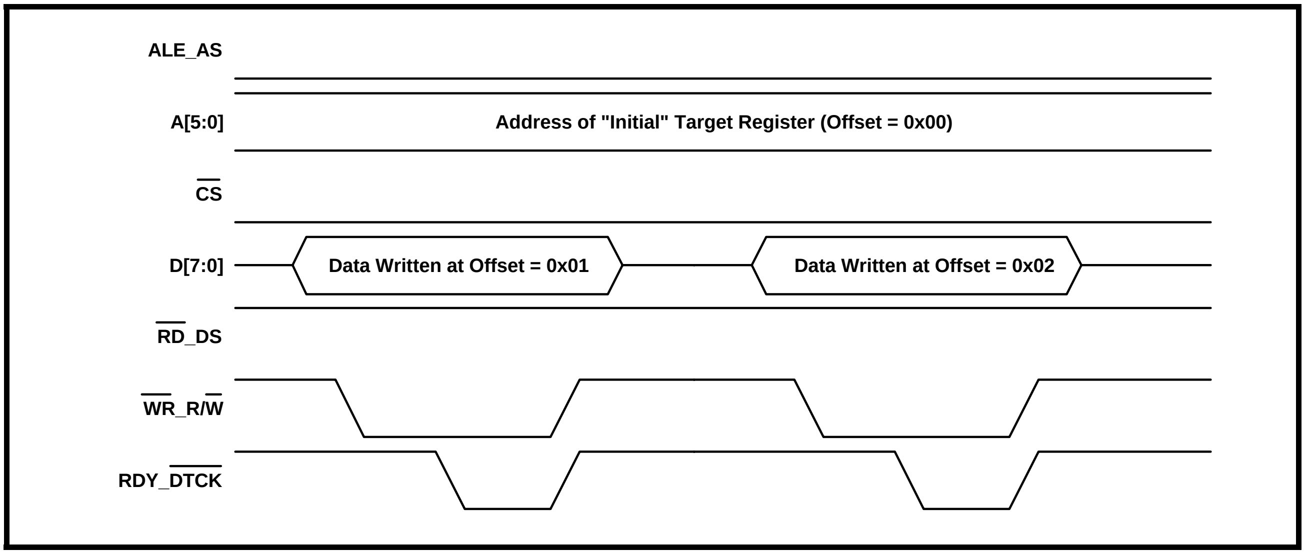
a. It enables the input drivers of the bi-directional data bus.

b. It causes the Framer to internally increment the value of the latched address.

B.3 After waiting the appropriate amount of settling time the data, in the internal data bus, will stabilize and is ready to be latched into the Framer Microprocessor Interface block. At this point, the µC/µP should latch the data into the Framer by toggling the $\overline{\text{WR}}$_R/$\overline{\text{W}}$ input pin "High".

For subsequent write operations, within this burst I/O access, the µC/µP simply repeats steps B.1 through B.3, as illustrated in Figure 25.

FIGURE 25. MICROPROCESSOR INTERFACE SIGNALS, DURING SUBSEQUENT WRITE OPERATIONS WITHIN THE BURST I/O CYCLE



3.3.2.3.1.2.3 Terminating the Burst I/O Access

Burst Access Operation will be terminated upon the rising edge of the ALE_AS input signal. At this point the Framer will cease to internally increment the latched address value. Further, the µC/µP is now free to execute either a Programmed I/O access or to start another Burst Access Operation with the XRT84V24 Framer.

3.3.2.3.2 Burst I/O Access: Motorola Mode

If the XRT84V24 Framer is interfaced to a Motorola-type µC/µP (e.g., the MC680x0 family, etc.), then it should be configured to operate in the Motorola mode (by tying the MOTO pin to VCC). Motorola-type Read and Write Burst I/O Access operations are described below.

3.3.2.3.2.1 Read Burst I/O Access Operation: Motorola-Mode

Whenever a Motorola-type µC/µP wishes to read the contents of numerous registers or buffer locations over a contiguous range of addresses, then it should do the following.

a. Perform the initial Read operation of the burst access.

b. Perform the remaining read operations in the burst access.

c. Terminate the burst access operation.

Each of these operations, within the Burst Access are discussed below.

3.3.2.3.2.1.1 Initial Read Operation: Motorola Mode

The initial read operation of a Motorola-type read burst access is accomplished by executing a Programmed I/O Read cycle, as summarized below.

A.0 Execute a Single Ordinary (Programmed I/O) Read Cycle, as described in steps A.1 through A.8 below.

A.1 Assert the ALE_AS (AS) input pin by toggling it "Low". This step enables the Address Bus input drivers (within the Framer) within the Framer Microprocessor Interface Block.

A.2 Place the address of the initial target register or buffer location (within the Framer), on the Address Bus input pins, A[5:0].

A.3 At the same time, the Address-Decoding circuitry (within the user's system) should assert the CS (Chip Select) input pins of the Framer by toggling it "Low". This action enables further communication between the µC/µP and the Framer Microprocessor Interface block.

A.4 After allowing the data on the Address Bus pins to settle (by waiting the appropriate Address Setup time), the µC/µP should toggle the ALE_AS input pin "High". This step causes the Framer to latch the contents of the Address Bus into its internal circuitry. At this point, the initial address of the burst access has now been selected.

A.5 Further, the µC/µP should indicate that this cycle is a Read cycle by setting the WR_R/W (R/W) input pin "High".

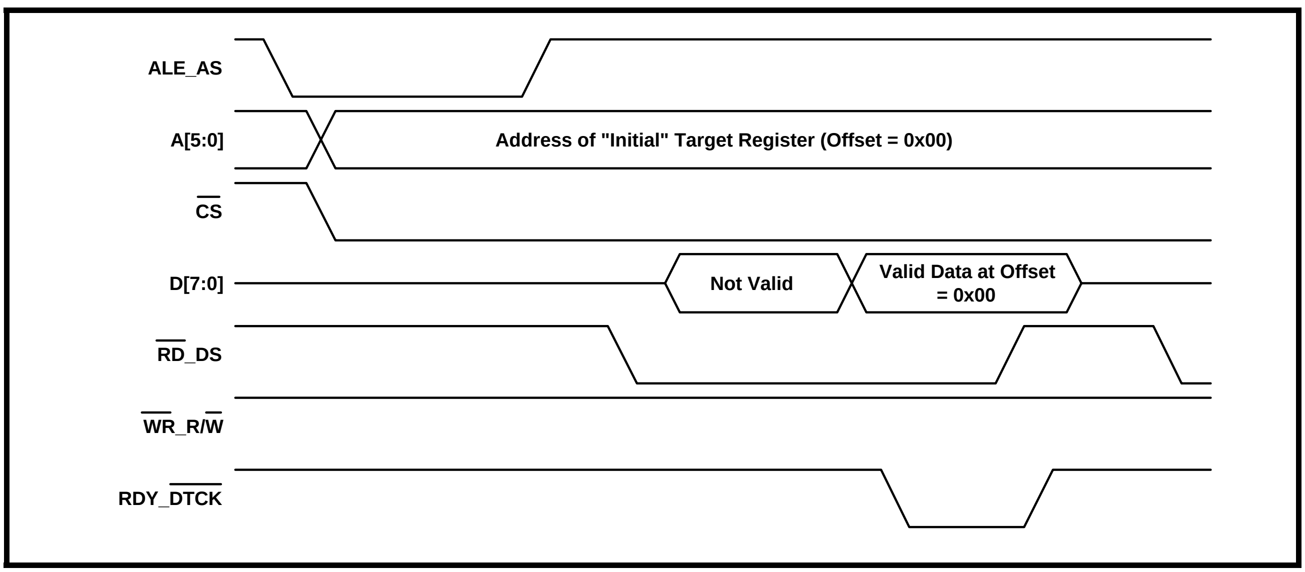
A.6 Next the µC/µP should initiate the current bus cycle by toggling the RD_DS (Data Strobe) input pin "Low". This step will enable the bi-directional data bus output drivers, within the Framer. At this point, the bi-directional data bus output drivers will proceed to driver the contents of the Address register onto the bi-directional data bus.

A.7 After some settling time, the data on the bi-directional data bus will stabilize and can be read by the µC/µP. The Framer will indicate that this data can be read by asserting the RDY_DTCK (DTACK) signal.

A.8 After the µC/µP detects the RDY_DTCK signal (from the Framer) it will terminate the Read Cycle by toggling the RD_DS (Data Strobe) input pin "High".

Figure 26 presents an illustration of the behavior of the Microprocessor Interface Signals during the initial Read Operation, within a Burst I/O Cycle, for a Motorola-type µC/µP.

FIGURE 26. MOTOROLA MICROPROCESSOR INTERFACE SIGNALS, DURING THE INITIAL READ OPERATION OF A BURST CYCLE



At the completion of this initial read cycle, the µC/µP has read in the contents of the first register or buffer location (within the Framer) for this particular burst access operation. In order to illustrate how this burst I/O cycle works, the byte (or word) of data, that is being read in Figure 26 has been labeled Valid Data at Offset = 0x00. This indicates that the µC/µP is reading the very first register (or buffer location) in this burst access.

3.3.2.3.2.1.2 Subsequent Read Operations

The procedure that the µC/µP must use to perform the remaining read cycles, within this Burst Access operation, is presented below.

B.0 Execute each subsequent Read Cycle, as described in steps B.1 through B.3, below.

B.1 Without toggling the ALE_AS input pin (e.g., keeping it "High"), toggle the RD_DS (Data Strobe) input pin "Low". This step accomplishes the following.

a. The Framer internally increments the latched address value (within the Microprocessor Interface circuitry).

b. The output drivers of the bi-directional data bus (D[7:0]) are enabled. At some time later, the register or buffer location corresponding to the incremented latched address value will be driven onto the bi-directional data bus.

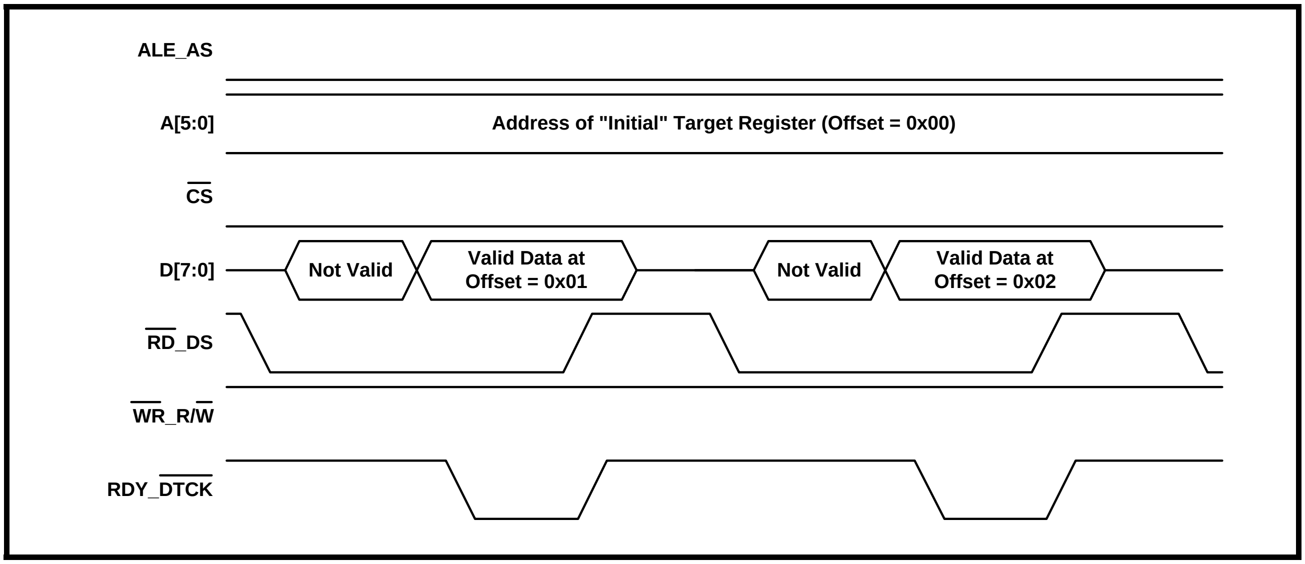
NOTE: *In order to insure that the Framer will interpret this signal as being a Read signal, the* µC/µP *should keep the* WR_R/W *input pin "High".*

B.2 After some settling time, the data on the bi-directional data bus will stabilize and can be read by the µC/µP. The Framer will indicate that this data is ready to be read by asserting the RDY_DTCK (DTACK) signal.

B.3 After the µC/µP detects the RDY_DTCK signal (from the Framer), it terminates the Read cycle by toggling the RD_DS (Data Strobe) input pin "High".

For subsequent read operations, within this burst cycle, the µC/µP simply repeats steps B.1 through B.3, as illustrated in Figure 27.

FIGURE 27. MOTOROLA MICROPROCESSOR INTERFACE SIGNALS, DURING SUBSEQUENT READ OPERATIONS WITHIN THE BURST I/O CYCLE



3.3.2.3.2.1.3 Terminating Burst Access Operation

The Burst I/O Access will be terminated upon the falling edge of the ALE_AS input signal. At this point the Framer will cease to internally increment the latched address value. Further, the µC/µP is now free to execute either a Programmed I/O access or to start another Burst Access Operation with the Framer.

3.3.2.3.2.2 Write Burst Access: Motorola-Mode

Whenever a Motorola-type µC/µP wishes to write the contents of numerous registers or buffer locations over a contiguous range of addresses, then it should do the following.

a. Perform the initial write operation of the burst access.

b. Perform the remaining write operations, of the burst access.

c. Terminate the burst access operation.

Each of these operations within the burst access are described below.

3.3.2.3.2.2.1 Initial Write Operation

The initial write operation of a Motorola-type Write Burst Access is accomplished by executing a Programmed I/O Write Cycle as summarized below.

A.0 Execute a Single Ordinary (Programmed I/O) Write cycle, as described in Steps A.1 through A.7 below.

A.1 Assert the ALE_AS (Address Strobe) input pin by toggling it "Low". This step enables the Address Bus input drivers (within the Framer).

A.2 Place the address of the initial target register or buffer location (within the Framer), on the Address Bus input pins, A[5:0].

A.3 At the same time, the Address-Decoding circuitry (within the user's system) should assert the CS input pin of the Framer by toggling it "Low". This step enables further communication between the µC/µP and the Framer Microprocessor Interface block.

A.4 After allowing the data on the Address Bus pins to settle (by waiting the appropriate Address Setup time), the µC/µP should toggle the ALE_AS input pin "High". This step causes the Framer to latch the contents of the Address Bus into its own circuitry. At this point, the initial address of the burst access has now been selected.

A.5 Further, the µC/µP should indicate that this current bus cycle is a Write operation by toggling the WR_R/W (R/W) input pin "Low".

A.6 The µC/µP should then place the byte or word that it intends to write into the target register, on the bi-directional data bus, D[7:0].

A.7 Next, the µC/µP should initiate the bus cycle by toggling the RD_DS (Data Strobe) input pin "Low". When the XRT84V24 Framer senses that the WR_R/W input pin is "Low", and that the RD_DS input pin has toggled "Low" it will enable the input drivers of the bi-directional data bus, D[7:0].

A.8 After waiting the appropriate amount of time, for this newly placed data to settle on the bi-directional data bus (e.g., the Data Setup time) the Framer will assert the RDY_DTCK (DTACK) output signal.

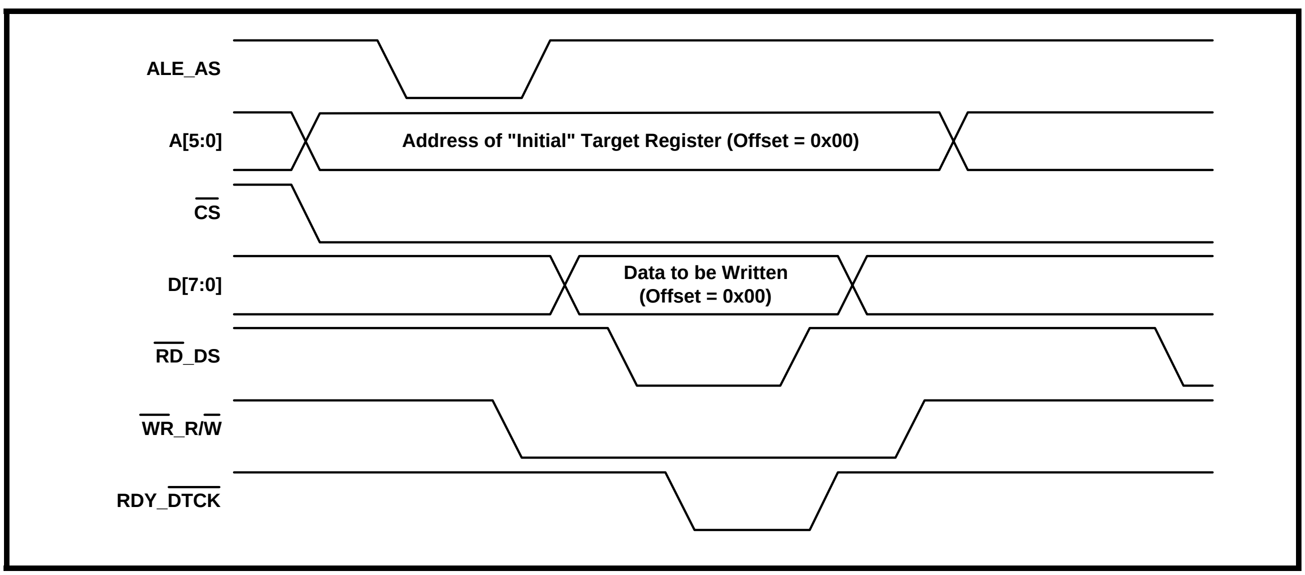
A.9 After the µP/µC detects the RDY_DTCK signal (from the Framer) it should toggle the RD_DS input pin "High". This action accomplishes two things:

a. It latches the contents of the bi-directional data bus into the Framer Microprocessor Interface block.

b. It terminates the Write cycle.

Figure 28 presents a timing diagram which illustrates the behavior of the Microprocessor Interface signals, during the Initial write operation within a Burst Access, for a Motorola-type µC/µP.

FIGURE 28. MOTOROLA MICROPROCESSOR INTERFACE SIGNALS, DURING THE INITIAL WRITE OPERATION OF A BURST CYCLE



At the completion of this initial write cycle, the µC/µP has written a byte or word into the first register or buffer location (within the Framer) for this particular burst I/O access. In order to illustrate how this burst I/O cycle works, the byte (or word) of data, that is being written in Figure 28 has been labeled Data to be Written (Offset = 0x00).

3.3.2.3.2.2.2 The Subsequent Write Operations

The procedure that the µC/µP must use to perform the remaining write cycles, within this burst access operation, is presented below.

B.0 Execute each subsequent write cycle, as described in steps B.1 through B.3.

B.1 Without toggling the ALE_AS input pin (e.g., keeping it "Low"), apply the value of the next byte or word (to be written into the Framer) to the bi-directional data bus pins, D[7:0].

B.2 Toggle the $\overline{WR}$_R/$\overline{W}$ (Write Strobe) input pin "Low". This step accomplishes two things.

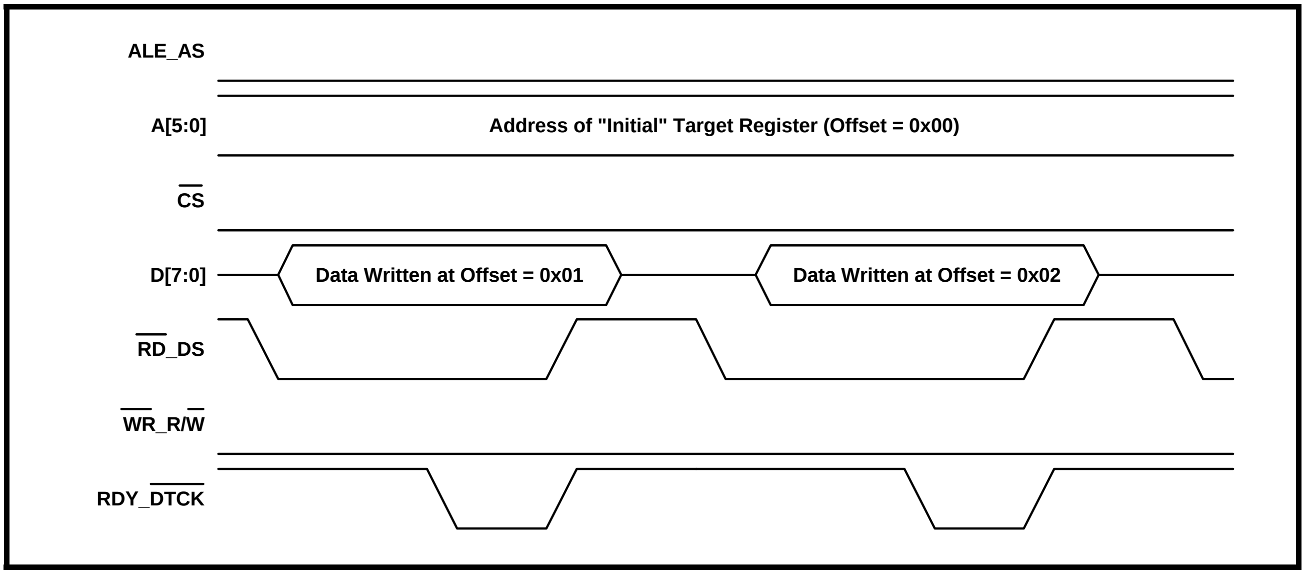
a. It enables the input drivers of the bi-directional data bus.

b. It causes the Framer to internally increment the value of the latched address.

B.3 After waiting the appropriate amount of settling time the data, in the internal data bus, will stabilize and is ready to be latched into the Framer Microprocessor Interface block. At this point, the µC/µP should latch the data into the Framer by toggling the $\overline{WR}$_R/$\overline{W}$ input pin "High".

For subsequent write operations, within this burst I/O access, the µC/µP simply repeats steps B.1 through B.3, as illustrated in Figure 29.

FIGURE 29. MOTOROLA MICROPROCESSOR INTERFACE SIGNALS, DURING SUBSEQUENT WRITE OPERATIONS WITH THE BURST I/O CYCLE



3.3.2.3.2.2.3 Terminating the Burst I/O Access

The Burst I/O Access will be terminated upon the falling edge of the ALE_AS input signal. At this point the Framer will cease to internally increment the latched address value. Further, the µC/µP is now free to execute either a Programmed I/O access or to start another Burst I/O Access with the Framer.

3.4 DMA READ/WRITE OPERATIONS

The XRT84V24 Quad E1 Framer contains two DMA Controller Interfaces which provide support for all four framers within the chip. The purpose of the two DMA Controllers is to facilitate the rapid block transfer of data between an external memory location and the on-chip HDLC buffers via the Microprocessor Interface.

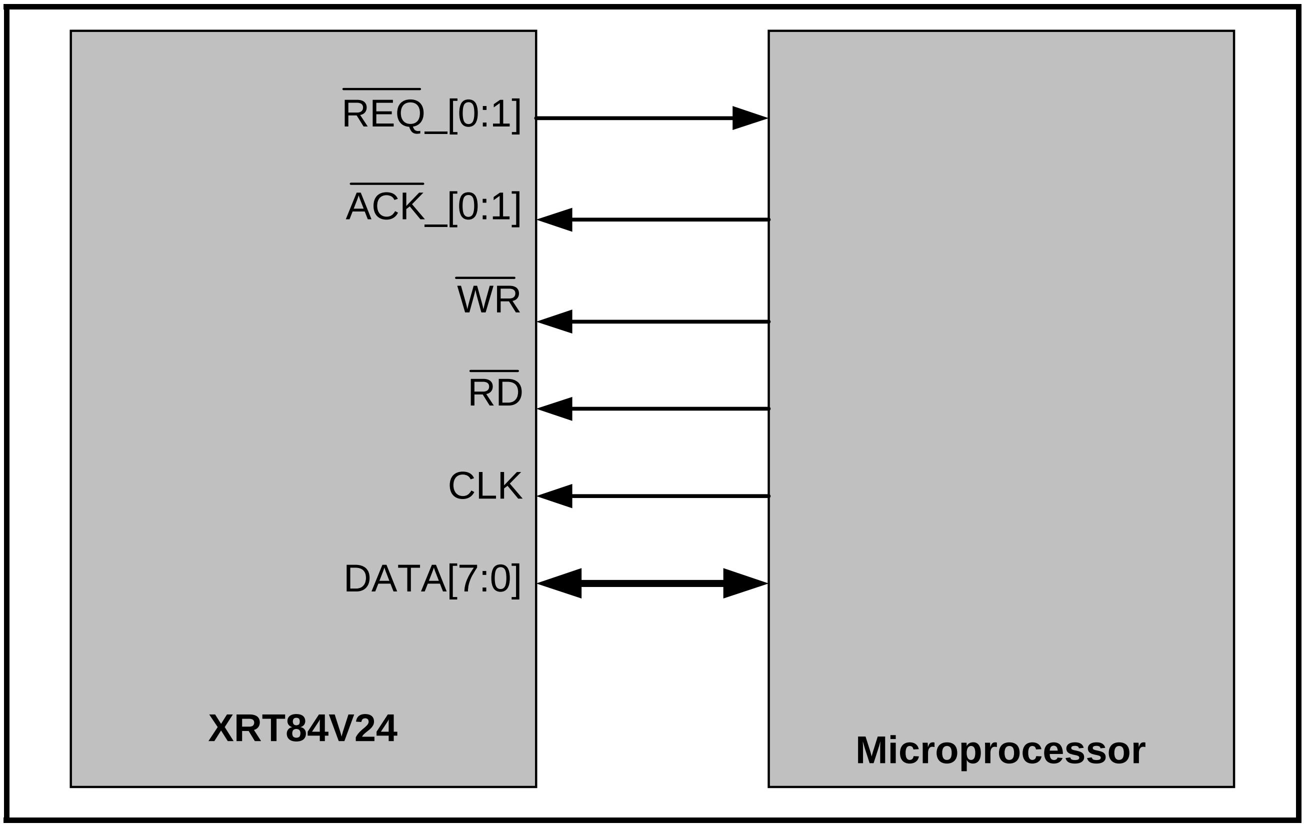
DMA-0 WRITE DMA INTERFACE

DMA 0 Controller Interface handles data transfer between external memory and the “selected” Transmit HDLC Buffer.

The DMA cycle starts when the XRT84V24 asserts the $\overline{REQ0}$ output pin. The “external” DMA Controller then responds by asserting the ACK0 input pin. The contents of the Microprocessor Interface bi-directional data bus are latched into the XRT84V24 each time the pWRL (Write Strobe) input pin is strobed “low”.

The XRT84V24 ends the DMA cycle by negating the DMA request input ($\overline{REQ0}$) while $\overline{WR}$ is still active. The external DMA Controller acknowledges the end of DMA Transfer by driving the ACK0 input pin “high”.

FIGURE 30. DMA MODE FOR THE XRT84V24 AND A MICROPROCESSOR



3.5 MEMORY AND REGISTER MAP

This section presents a complete list of the E1 Framer external memory address map and the internal memory map. In addition, the allocations of the three internal storage spaces is depicted.

3.5.1 Memory Mapped I/O Indirect Addressing

The XRT84V24 employs a "complete" indirect addressing approach for the Microprocessor Interface; in order to support multiple channel implementations, maintaining rich user-controlled features, minimizing the total pin count and providing future scalability without sacrificing performance for microcontroller access. Six address bits are used with the 2 MSB (most significant bits) identifying each of the four E1 framers channels and the 4 LSBs to address the indirect mapping registers. Table 9 through Table 14 indicates the indirect address memory mapping.

TABLE 9: INDIRECT ADDRESS MEMORY MAP

INDIRECT ADDRESS	CONTENTS
Framer Number 0	
00	Channel 0—Control Register/Indirect Address Register
01	Channel 0—Control Register/Indirect Data Register
02	Channel 0—Channel Control/Indirect Address Register
03	Channel 0—Channel Control/Indirect Data Register
04	Channel 0—Receive Signaling Array/Indirect Address Register
05	Channel 0—Receive Signaling Array/Indirect Data Register
06	Channel 0—LAPD Buffer 0/Indirect Data Register
07	Channel 0—LAPD Buffer 1/Indirect Data Register
08	Channel 0—Performance Monitor/Indirect Address Register

TABLE 9: INDIRECT ADDRESS MEMORY MAP (CONTINUED)

INDIRECT ADDRESS	CONTENTS
09	Channel 0—Performance Monitor/Indirect Data Register
0A	Channel 0—Interrupt Indirect Address Register
0B	Channel 0—Interrupt Indirect Data Register
0C–0F	Reserved
Framer Number 1	
10	Channel 1—Control Register/Indirect Address Register
11	Channel 1—Control Register/Indirect Data Register
12	Channel 1—Channel Control/Indirect Address Register
13	Channel 1—Channel Control/Indirect Data Register
14	Channel 1—Receive Signaling Array/Indirect Address Register
15	Channel 1—Receive Signaling Array/Indirect Data Register
16	Channel 1—LAPD Buffer 0/Indirect Data Register
17	Channel 1—LAPD Buffer 1/Indirect Data Register
18	Channel 1—Performance Monitor/Indirect Address Register
19	Channel 1—Performance Monitor/Indirect Data Register
1A	Channel 1—Interrupt Indirect Address Register
1B	Channel 1—Interrupt Indirect Data Register
1C–1F	Reserved
Framer Number 2	
20	Channel 2—Control Register/Indirect Address Register
21	Channel 2—Control Register/Indirect Data Register
22	Channel 2—Channel Control/Indirect Address Register
23	Channel 2—Channel Control/Indirect Data Register
24	Channel 2—Receive Signaling Array/Indirect Address Register
25	Channel 2—Receive Signaling Array/Indirect Data Register
26	Channel 2—LAPD Buffer 0/Indirect Data Register
27	Channel 2—LAPD Buffer 1/Indirect Data Register
28	Channel 2—Performance Monitor/Indirect Address Register
29	Channel 2—Performance Monitor/Indirect Data Register
2A	Channel 2—Interrupt Indirect Address Register
2B	Channel 2—Interrupt Indirect Data Register
2C–2F	Reserved
Framer Number 3	
30	Channel 3—Control Register/Indirect Address Register
31	Channel 3—Control Register/Indirect Data Register
32	Channel 3—Channel Control/Indirect Address Register
33	Channel 3—Channel Control/Indirect Data Register
34	Channel 3—Receive Signaling Array/Indirect Address Register
35	Channel 3—Receive Signaling Array/Indirect Data Register
36	Channel 3—LAPD Buffer 0/Indirect Data Register
37	Channel 3—LAPD Buffer 1/Indirect Data Register

TABLE 9: INDIRECT ADDRESS MEMORY MAP (CONTINUED)

INDIRECT ADDRESS	CONTENTS
38	Channel 3—Performance Monitor/Indirect Address Register
39	Channel 3—Performance Monitor/Indirect Data Register
3A	Channel 3—Interrupt Indirect Address Register
3B	Channel 3—Interrupt Indirect Data Register
3C—3F	Reserved

TABLE 10: CONTROL REGISTER ADDRESS MAP (INDIRECT ADDRESS = X0H, FOR ADDRESS AND X1H, FOR DATA)

ADDRESS	CONTENT	REGISTER TYPE
00h	Clock Select Register	R/W
01h	Line Interface Control Register	R/W
02h	Line Control Register	R/W
03h	LIU Access Register 1	R/W
04h	LIU Access Register 2	R/W
05h	LIU Poll Register 1	R/W
06h	LIU Poll Register 2	R/W
07h	Framing Select Register	R/W
08h	Alarm Generation Register	R/W
09h	Synchronization Mux Register	R/W
0Ah	Transmit Signaling & Data Link Select Register	R/W
0Bh	Framing Control Register	R/W
0Ch	Receive Signaling & Data Link Select Register	R/W
0Dh–10h	Signaling Change Register 0–4	RUR
11h	Receive National Bits Registers	RO
12h	Receive Extra Bits Registers	RO
13h	Data Link Control Register	R/W
14h	Transmit Data Link Byte Count Register	R/W
15h	Receive Data Link Byte Count Register	RO
16h	Slip Buffer Control Register	R/W
17h	FIFO Latency Register	R/W
18h	DMA 0 Write Configuration Register	R/W
19h	DMA 1 Read Configuration Register	R/W
1Ah	Interrupt Control Register	R/W
80h	Transmit HDLC Buffer 0 Indirect Address Register	R/W
81h	Transmit HDLC Buffer 0 Indirect Data Register	R/W
82h	Transmit HDLC Buffer 1 Indirect Address Register	R/W
83h	Transmit HDLC Buffer 1 Indirect Data Register	R/W
84h	Receive HDLC Buffer 0 Indirect Address Register	R/W
85h	Receive HDLC Buffer 0 Indirect Data Register	R/W
86h	Receive HDLC Buffer 1 Indirect Address Register	R/W
87h	Receive HDLC Buffer 1 Indirect Data Register	R/W

TABLE 10: CONTROL REGISTER ADDRESS MAP (INDIRECT ADDRESS = X0H, FOR ADDRESS AND X1H, FOR DATA) (CONTINUED)

ADDRESS	CONTENT	REGISTER TYPE
FDh	Diag Group Register	
FEh	Part Number Register	
FFh	Version Number Register	

TABLE 11: TRANSMIT CHANNEL CONTROL REGISTER ADDRESS MAP (INDIRECT ADDRESS = X2H FOR ADDRESS AND X3H FOR DATA)

ADDRESS	CONTENT	TYPE
00	Transmit Channel Control Register 0	R/W
01	Transmit Channel Control Register 1	R/W
02	Transmit Channel Control Register 2	R/W
03	Transmit Channel Control Register 3	R/W
04	Transmit Channel Control Register 4	R/W
05	Transmit Channel Control Register 5	R/W
06	Transmit Channel Control Register 6	R/W
07	Transmit Channel Control Register 7	R/W
08	Transmit Channel Control Register 8	R/W
09	Transmit Channel Control Register 9	R/W
0A	Transmit Channel Control Register 10	R/W
0B	Transmit Channel Control Register 11	R/W
0C	Transmit Channel Control Register 12	R/W
0D	Transmit Channel Control Register 13	R/W
0E	Transmit Channel Control Register 14	R/W
0F	Transmit Channel Control Register 15	R/W
10	Transmit Channel Control Register 16	R/W
11	Transmit Channel Control Register 17	R/W
12	Transmit Channel Control Register 18	R/W
13	Transmit Channel Control Register 19	R/W
14	Transmit Channel Control Register 20	R/W
15	Transmit Channel Control Register 21	R/W
16	Transmit Channel Control Register 22	R/W
17	Transmit Channel Control Register 23	R/W
18	Transmit Channel Control Register 24	R/W
19	Transmit Channel Control Register 25	R/W
1A	Transmit Channel Control Register 26	R/W
1B	Transmit Channel Control Register 27	R/W
1C	Transmit Channel Control Register 28	R/W
1D	Transmit Channel Control Register 29	R/W
1E	Transmit Channel Control Register 30	R/W
1F	Transmit Channel Control Register 31	R/W
20	User Code Register 0	R/W

TABLE 11: TRANSMIT CHANNEL CONTROL REGISTER ADDRESS MAP (INDIRECT ADDRESS = X2H FOR ADDRESS AND X3H FOR DATA) (CONTINUED)

ADDRESS	CONTENT	TYPE
21	User Code Register 1	R/W
22	User Code Register 2	R/W
23	User Code Register 3	R/W
24	User Code Register 4	R/W
25	User Code Register 5	R/W
26	User Code Register 6	R/W
27	User Code Register 7	R/W
28	User Code Register 8	R/W
29	User Code Register 9	R/W
2A	User Code Register 10	R/W
2B	User Code Register 11	R/W
2C	User Code Register 12	R/W
2D	User Code Register 13	R/W
2E	User Code Register 14	R/W
2F	User Code Register 15	R/W
30	User Code Register 16	R/W
31	User Code Register 17	R/W
32	User Code Register 18	R/W
33	User Code Register 19	R/W
34	User Code Register 20	R/W
35	User Code Register 21	R/W
36	User Code Register 22	R/W
37	User Code Register 23	R/W
38	User Code Register 24	R/W
39	User Code Register 25	R/W
3A	User Code Register 26	R/W
3B	User Code Register 27	R/W
3C	User Code Register 28	R/W
3D	User Code Register 29	R/W
3E	User Code Register 30	R/W
3F	User Code Register 31	R/W

TABLE 12: RECEIVE SIGNALLING REGISTER ARRAY—ADDRESS MAP (INDIRECT ADDRESS = X4H, FOR ADDRESS AND X5H, FOR DATA)

ADDRESS	CONTENT	REGISTER TYPE
00	Receive Signalling Register Array—0	R/W
01	Receive Signalling Register Array—1	R/W
02	Receive Signalling Register Array—2	R/W
03	Receive Signalling Register Array—3	R/W
04	Receive Signalling Register Array—4	R/W

TABLE 12: RECEIVE SIGNALLING REGISTER ARRAY—ADDRESS MAP (INDIRECT ADDRESS = X4H, FOR ADDRESS AND X5H, FOR DATA) (CONTINUED)

ADDRESS	CONTENT	REGISTER TYPE
05	Receive Signalling Register Array—5	R/W
06	Receive Signalling Register Array—6	R/W
07	Receive Signalling Register Array—7	R/W
08	Receive Signalling Register Array—8	R/W
09	Receive Signalling Register Array—9	R/W
0A	Receive Signalling Register Array—10	R/W
0B	Receive Signalling Register Array—11	R/W
0C	Receive Signalling Register Array—12	R/W
0D	Receive Signalling Register Array—13	R/W
0E	Receive Signalling Register Array—14	R/W
0F	Receive Signalling Register Array—15	R/W
10	Receive Signalling Register Array—16	R/W
11	Receive Signalling Register Array—17	R/W
12	Receive Signalling Register Array—18	R/W
13	Receive Signalling Register Array—19	R/W
14	Receive Signalling Register Array—20	R/W
15	Receive Signalling Register Array—21	R/W
16	Receive Signalling Register Array—22	R/W
17	Receive Signalling Register Array—23	R/W
18	Receive Signalling Register Array—24	R/W
19	Receive Signalling Register Array—25	R/W
1A	Receive Signalling Register Array—26	R/W
1B	Receive Signalling Register Array—27	R/W
1C	Receive Signalling Register Array—28	R/W
1D	Receive Signalling Register Array—29	R/W
1E	Receive Signalling Register Array—30	R/W
1F	Receive Signalling Register Array—31	R/W

TABLE 13: PERFORMANCE MONITOR ADDRESS MAP, INDIRECT ADDRESS = X8H (FOR ADDRESS) AND X9H (FOR DATA)

ADDRESS	CONTENT	TYPE
00h	E1 Receive Line Code Violation Counter (LSB)	RUR
01h	E1 Receive Line Code Violation Counter (MSB)	RUR
02h	E1 Receive Frame Alignment Error Counter (LSB)	RUR
03h	E1 Receive Frame Alignment Error Counter (MSB)	RUR
04h	E1 Receive Severely Errored Frame Counter	RUR
05h	E1 Receive CRC-4 Block Error Counter (LSB)	RUR
06h	E1 Receive CRC-4 Block Error Counter (MSB)	RUR
07h	E1 Receive Far End Block Error Counter (LSB)	RUR
08h	E1 Receive Far End Block Error Counter (MSB)	RUR

TABLE 13: PERFORMANCE MONITOR ADDRESS MAP, INDIRECT ADDRESS = X8H (FOR ADDRESS) AND X9H (FOR DATA)

ADDRESS	CONTENT	TYPE
09h	E1 Receive Slip Counter	RUR
0Ah	E1 Receive Loss of Frame Counter	RUR
0Bh	E1 Receive Change of Frame Alignment Counter	RUR
0Ch	E1 Frame Check Sequence Error Counter	RUR

TABLE 14: INTERRUPT REGISTERS ADDRESS = XAH (FOR ADDRESS) AND XBH (FOR DATA)

ADDRESS	CONTENT	REGISTER TYPE
00h	Block Interrupt Status Register	RUR - RO
01h	Block Interrupt Enable Register	R/W
02h	Alarm and Error Status Register	RUR - RO
03h	Alarm and Error Enable Register	R/W
04h	E1 Framer Interrupt Status Register	RUR
05h	E1 Framer Interrupt Enable Register	R/W
06h	Data Link Interrupt Status Register	RUR
07h	Data Link Interrupt Enable Register	R/W
08h	Slip Buffer Interrupt Status Register	R/W
09h	Slip Buffer Interrupt Enable Register	RO - RUR

3.6 DESCRIPTION OF THE CONTROL REGISTERS

3.6.1 List of Registers

All even numbered registers get mapped onto the microprocessor data bus higher byte D15-D8.

All Odd numbered registers get mapped onto the microprocessor data bus lower byte D7-D0.

Even Numbered Register								Odd Numbered Register							
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Register Summary List

REG. #	FUNCTION
0	Clock Select Register: Table 15
1	Line Interface Control Register: Table 16
2	Line Control Register: Table 17
3	LIU Access Register 1: Table 18
4	LIU Access Register 2: Table 19
5	LIU Poll Register 1: Table 20
6	LIU Poll Register 2: Table 21
7	Framing Select Register: Table 22
8	Alarm Generation Register: Table 23
9	Synchronization MUX Register: Table 24
10	Transmit Signaling and Data Link Select Register: Table 25
11	Framing Control Register: Table 26
12	Receive Signaling & Data Link Register: Table 27
13	Signaling Change Register 0: Table 28
14	Signaling Change Register 1: Table 29
15	Signaling Change Register 2: Table 30
16	Signaling Change Register 3: Table 31
17	Receive National Bits Register: Table 32
18	Receive Extra Bits Register: Table 33
19	Data Link Control Register: Table 34
20	Transmit Data Link Byte Count Register: Table 35
21	Receive Data Link Byte Count Register: Table 36
22	Slip Buffer Control Register: Table 37
23	FIFO Latency Register: Table 38
24	DMA 0 (Write) Configuration Register: Table 39
25	DMA 1 (Read) Configuration Register: Table 40
26	Interrupt Control Register: Table 41
27	Transmit Channel Control Register 0: Table 42
28	Transmit Channel Control Register 1: Table 43
29	Transmit Channel Control Register -2: Table 44
30	Transmit Control Register - 3: Table 45
31	Transmit Control Register -4: Table 46
32	Transmit Control Register - 5: Table 47

REG. #	FUNCTION
33	Transmit Control Register - 6: Table 48
34	Transmit Control Register - 7: Table 49
35	Transmit Channel Control Register - 8: Table 50
36	Transmit Channel Control Register - 9: Table 51
37	Transmit Channel Control Register - 10: Table 52
38	Transmit Channel Control Register - 11: Table 53
39	Transmit Channel Control Register - 12: Table 54
40	Transmit Channel Control Register - 13: Table 55
41	Transmit Channel Control Register - 14: Table 56
42	Transmit Channel Control Register - 15: Table 57
43	Transmit Channel Control Register - 16: Table 58
44	Transmit Control Channel Register - 17: Table 59
45	Transmit Channel Control Register - 18: Table 60
46	Transmit Channel Control Register - 19: Table 61
47	Transmit Channel Control Register - 20: Table 62
48	Transmit Channel Control Register - 21: Table 63
49	Transmit Channel Control Register - 22: Table 64
50	Transmit Channel Control Register - 23: Table 65
51	Transmit Channel Control Register - 24: Table 66
52	Transmit Channel Control Register - 25: Table 67
53	Transmit Channel Control Register - 26: Table 68
54	Transmit Channel Control Register - 27: Table 69
55	Transmit Channel Control Register - 28: Table 70
56	Transmit Channel Control Register - 29: Table 71
57	Transmit Channel Control Register - 30: Table 72
58	Transmit Channel Control Register - 31: Table 73
59	User Code Register 0: Table 74
60	User Code Register 1: Table 75
61	User Code Register 2: Table 76
62	User Code Register 3: Table 77
63	User Code Register 4: Table 78
64	User Code Register 5: Table 79
65	User Code Register 6: Table 80
66	User Code Register 7: Table 81
67	User Code Register 8: Table 82

REG. #	FUNCTION
68	User Code Register 9: Table 83
69	User Code Register 10: Table 84
70	User Code Register 11: Table 85
71	User Code Register 12: Table 86
72	User Code Register 13: Table 87
73	User Code Register 14: Table 88
74	User Code Register 15: Table 89
75	User Code Register 16: Table 90
76	User Code Register 17: Table 91
77	User Code Register 18: Table 92
78	User Register 19: Table 93
79	User Code Register 20: Table 94
80	User Code Register 21: Table 95
81	User Code Register 22: Table 96
82	User Code Register 23: Table 97
83	User Code Register 24: Table 98
84	User Code Register 25: Table 99
85	User Code Register 26: Table 100
86	User Code Register 27: Table 101
87	User Code Register 28: Table 102
88	User Code Register 29: Table 103
89	User Code Register 30: Table 104
90	User Code Register 31: Table 105
91	Receive Signaling Array Register 0: Table 106
92	Receive Signaling Array Register 1: Table 107
93	Receive Signaling Array Register 2: Table 108
94	Receive Signaling Array Register 3: Table 109
95	Receive Signaling Array Register 4: Table 110
96	Receive Signaling Array Register 5: Table 111
97	Receive Signaling Array Register 6: Table 112
98	Receive Signaling Array Register 7: Table 113
99	Receive Signaling Array Register 8: Table 114
100	Receive Signaling Array Register 9: Table 115
101	Receive Signaling Array Register 10: Table 116
102	Receive Signaling Array Register 11: Table 117

REG. #	FUNCTION
103	Receive Signaling Array Register 12: Table 118
104	Receive Signaling Array Register 13: Table 119
105	Receive Signaling Array Register 14: Table 120
106	Receive Signaling Array Register 15: Table 121
107	Receive Signaling Array Register 16: Table 122
108	Receive Signaling Array Register 17: Table 123
109	Receive Signaling Array Register 18: Table 124
110	Receive Signaling Array Register 19: Table 125
111	Receive Signaling Array Register 20: Table 126
112	Receive Signaling Array Register 21: Table 127
113	Receive Signaling Array Register 22: Table 128
114	Receive Signaling Array Register 23: Table 129
115	Receive Signaling Array Register 24: Table 130
116	Receive Signaling Array Register 25: Table 131
117	Receive Signaling Array Register 26: Table 132
118	Receive Signaling Array Register 27: Table 133
119	Receive Signaling Array Register 28: Table 134
120	Receive Signaling Array Register 29: Table 135
121	Receive Signaling Array Register 30: Table 136
122	Receive Signaling Array Register 31: Table 137
123	PMON E1 Receive Line Code (bipolar) Violation Counter: Table 138
124	PMON E1 Receive Line Code (bipolar) Violation Counter: Table 139
125	PMON E1 Receive Framing Alignment Bit Error Counter: Table 140
126	PMON E1 Receive Framing Alignment Bit Error Counter: Table 141
127	PMON E1 Receive Severely Errored Frame Counter: Table 142
128	PMON E1 Receive CRC-4 Block Error Counter - LSB: Table 143
129	PMON E1 Receive CRC-4 Block Error Counter - MSB: Table 144
130	PMON E1 Receive Far End Block Error Counter: Table 145
131	PMON E1 Receive Far-End BLock Error Counter - MSB: Table 146
132	PMON E1 Receive Slip Counter: Table 147

REG. #	FUNCTION
133	PMON E1 Receive Loss of Frame Counter: Table 148
134	PMON E1 Receive Change of Frame Alignment Counter: Table 149
135	PMON LAPD E1 Frame Check Sequence Error Counter: Table 150
136	Block Interrupt Status Register: Table 151
137	Block Interrupt Enable Register: Table 152
138	Alarm & Error Interrupt Status Register: Table 153
139	Alarm & Error Interrupt Enable Register: Table 154
140	Framer Interrupt Status Register: Table 155
141	Framer Interrupt Enable Register: Table 156
142	Data Link Status Register: Table 157
146	Data Link Interrupt Enable Register: Table 158
144	Slip Buffer Interrupt Enable Register: Table 159
145	Slip Buffer Interrupt STatus Register: Table 160

3.6.2 Description of the Control Registers

Table 15: Clock Select Register

Register 0 | **Clock Select Register** | **Hex Address: x0,00**

Bit	Function	Type	Default	Description-Operation
7	Unused	RO	0	
6	Reserved	R/W	0	
5	8kHz	R/W	0	This Read/Write Bit-Field allows the user to configure the transmit sections of all four framer blocks to synchronize their frame alignment with the signal applied to the 8kREF input pin. Setting this bit-field to a “1” enables this feature for all four channels. ***Note:*** *This bit-field is ignored if TxSerClk_n or RxLineClk_n is configured to be the timing reference for the transmit section.*
4	CLDET	R/W	0	**Clock Loss Detect Enable/Disable Select** Enables a protection feature for the Framer whenever the Recovered Received Line Clock (RxLineClk) is used as the timing source for the transmit section of the framer. If the Clock Loss Detection protection feature is enabled and the Recovered Received Line Clock is used as the timing source, then if the LIU somehow loses clock recovery the Clock Distribution Block will detect this occurrence and automatically begin to use the OSCClk Driven Divided clock as the Transmitter source, until the LIU is able to regain clock recovery.
3	CFS(1)	R/W	0	**Frequency Select** Specifies the frequency of the oscillator clock. 00: The OSCClk input is 16.384 MHz (internally divided by 1) 01: The OSCClk input is 32.768 MHz (internally divided by 2) 10: The OSCClk input is 65.536 MHz (internally divided by 4) 11: Reserved ***Note:*** *This bit-field is ignored if TxSerClk_n or RxLineClk_n is configured to be the timing reference for the transmit section.*
2	CFS(0)	R/W	0	
1	CSS(1)	R/W	0	**Clock Source Select** Specifies the timing source for the Transmit E1 Framer block (associated with this register). 00: RxLineClk - the Recovered Recieved Channel Input Clock is chosen as the timing reference for the transmit section of Framer N (Loop Timing) 01: TxSerClk - The Transmit E1 Serial Data Input Clock is chosen as the timing reference for the timing source for the transmit section of Framer n. 10: OSCClk - the OSCClk-driven divided clock is chosen as the timing reference for the transmit section of Framer n. 11: RxLineClk - The Recovered Received Channel Input Clock is chosen as the timing reference for the transmit section of Framer n.
0	CSS(0)	R/W	1	

TABLE 16: LINE INTERFACE CONTROL REGISTER

REGISTER 1 **LINE INTERFACE CONTROL REGISTER** **HEX ADDRESS:X0,01**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	Unused	RO	0	
6	NRS	R/W	0	**Dual-Rail/Single-Rail Select** This Read/Write bit-field is used to configure a channel to operate in either the Single-Rail or Dual-Rail mode. 0: Selects the Dual-Rail Mode. 1: Selects the Single-Rail Mode
5	LB(1)	R/W	0	**Loopback Selection** These two Read/Write Bit-Fields are used to configure a given channel to operate in any of the following loop-back modes 00: No local loopback 01: Local loopback 10: Remote Line Loopback 11: Reserved
4	LB(0)	R/W	0	
3	TCI	R/W	0	**Transmit Clock Inversion Select** This Read/Write bit-field is used to configure a channel to update TxPOS_n and TxNEG_n output data on either the rising or falling edge of the TxLineClk_n output signal 0: Configures the Transmit LIU Interface block to update TxPOS and TxNEG on the rising edge of TxLineClk. 1: Configures the Transmit LIU Interface block to update TxPOS and TxNEG on the falling edge of TxLineClk.
2	RCI	R/W	0	**Receive Clock Inversion Select** Specifies whether the RxPOS and RxNEG input signals should be sampled (latched into the Receive LIU Interface) on the rising or falling edge of RxLineClk. 0: Configures the Receive LIU Interface to sample RxPOS and RxNEG on the rising edge of RxLineClk. 1: Configures the Receive LIU Interface to sample RxPOS and RxNEG on the falling edge of RxLineCLk
1	Encode AMI/$\overline{\text{HDB3}}$	R/W	0	**Encode AMI/HDB3* Line Code Select** Configures the Transmit LIU Interface block to transmit data via the AMI or HDB3 line codes. 0: Transmit LIU interface block transmits the E1 frame data in the HDB3 line code. 1: Transmit LIU interface block transmits the E1 frame data in the AMI line code.
0	Decode AMI/$\overline{\text{HDB3}}$	R/W	0	**Decode AMI/HDB3* Line Code Select** Enables or disables the HDB3 decoder with in the Receive LIU interface block. 0: Enables the HDB3 decoder 1: Disables the HDB3 decoder ***NOTE:*** *This bit-field is ignored if the Single-Rail Mode is selected (if bit 6 of this register is set to 1)*

TABLE 17: LINE CONTROL REGISTER

REGISTER 2	LINE CONTROL REGISTER	HEX ADDRESS: x00,02

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	MODE	R/W	0	**LIU Controller Block MODE Select** 0: Configures the LIU Controller Block to operate in the HARDWARE mode. 1: Configures the LIU Controller Block to operate in the HOST mode. *Hardware Mode:* Pins GPO_0 through GPO_6 will function as general purpose output pins. Additionally, bits 0 through 6 will function as Read/Write bits that control the logic state of these output pins. *Host Mode*: GPO_0 through GPO_6 pouts will function as a Serial Output Interface capable of communicating with up to 4 LIU devices over a Microprocessor Serial Interface.
6	GPO(6)/CEDGE	R/W	0	Function dependent on whether the LIU Controller Block is in HARDWARE or HOST mode. *Hardware Mode*: Controls the logic state of the GPO[6] output pin. 0: GPO[6] output pin toggles low. 1: GPO[6] output pin toggles high. *Host Mode*: Selects which clock edge of SCLK the LIU Controller block will use to update its output data via the SDI pin. CEDGE=0: SDI output updated on rising edge of SCLK SDO input is sampled on falling edge of SCLK CEDGE=1: SDI output updated on falling edge of SCLK SDO input is sampled on rising edge of SCLK
5	GPO(5)	R/W	0	0: Output pin toggles low 1: Output pin toggles high ***NOTE:*** Active only when the LIU Controller block is operating in Hardware mode.
4	GPO(4)	R/W	0	0: Output pin toggles low 1: Output pin toggles high ***NOTE:*** Active only when the LIU Controller block is operating in Hardware mode.
3	GPO(3)	R/W	0	0: Output pin toggles low 1: Output pin toggles high ***NOTE:*** Active only when the LIU Controller block is operating in Hardware mode.
2	GPO(2)	R/W	0	0: Output pin toggles low 1: Output pin toggles high ***NOTE:*** Active only when the LIU Controller block is operating in Hardware mode.
1	GPO(1)	R/W	0	0: Output pin toggles low 1: Output pin toggles high ***NOTE:*** Active only when the LIU Controller block is operating in Hardware mode.
0	GPO(0)	R/W	0	0: Output pin toggles low 1: Output pin toggles high ***NOTE:*** Active only when the LIU Controller block is operating in Hardware mode.

TABLE 18: LIU ACCESS REGISTER 1

REGISTER 3 **LIU ACCESS REGISTER 1** **HEX ADDRESS:x00,03**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	LAR1(7:0)	R/W	0	

TABLE 19: LIU ACCESS REGISTER 2

REGISTER 4 **LIU ACCESS REGISTER 2** **HEX ADDRESS x00,04**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	LAR2(7:0)	R/W	0	

TABLE 20: LIU POLL REGISTER 1

REGISTER 5 **LIU POLL REGISTER 1** **HEX ADDRESS: x00,05**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	LIU_P1(7:0)	R/W	0	

TABLE 21: LIU POLL REGISTER 2

REGISTER 6 **LIU POLL REGISTER 2** **HEX ADDRESS: x00,06**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	LIU_P2(7:0)	R/W	0	

TABLE 22: FRAMING SELECT REGISTER

REGISTER 7	FRAMING SELECT REGISTER (FSR)	HEX ADDRESS: X00,07

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	Unused	RO	0	
6	CRCDIAG	R/W	0	**CRC Diagnostics Select Enable/Disable** This Read/Write bi-field is used to force an errored CRC pattern in the outbound CRC multiframe to be sent on the transmission line. The transmit section will implement this error by inverting the value of CRC bit (C1) 0: Transmit E1 Framer functions normally (no errors) 1: Transmits errored CRC bit ***NOTE:*** *This bit-field is ignored if CRC multi-Framing is disabled.*
5	CASSEL(1)	R/W	0	**CAS Multiframe Alignment Algorithm Select** Allows the user to select which CAS Multiframe Alignment algorithm to employ. 00: CAS Multiframe Alignment disabled 01: CAS Multiframe Alignment Algorithm 1 enabled 10: CAS Multiframe Alignment Algorithm 2 enabled 11: CAS Multiframe Alignment disabled
4	CASSEL(0)	R/W	0	
3	CRCSEL(1)	R/W	0	**CRC Multiframe Alignment Criteria Select** Allows the user to select which CRC-Multiframe Alignment to employ. 00: CRC Multiframe Alignment disabled 01: CRC Multiframe Alignment enabled. Alignment is declared if at least one valid CRC multiframe alignment signal (0,0,1,0,1,1,E1,E2) is observed within 8ms. 10: CRC Multiframe Alignment enabled. Alignment is declared if at least two valid CRC multiframe alignment signals (0,0,1,0,1,1,E1,E2) are observed within 8ms with the time separating the two alignment signals being multiples of 2ms. 11:CRC Multiframe Alignment enabled. Alignment is declared if at least 3 valid CRC multiframe alignment signals (0,0,1,0,1,1,E1,E2) are observed within 8ms with the time separating the two alignment signals being multiples of 2ms.
2	CRCSEL(0)	R/W	0	
1	CKSEQ_ENB	R/W	0	**Check Sequence Enable-FAS Alignment** Enable/Disable frame check sequence in FAS alignment process. 0: Disables Frame Check Sequence 1: Enables Frame Check Sequence‘
0	FASSEL	R/W	0	**FAS Alignment Algorithm Select** Specifies which algorithm the Receive E1 Framer block uses in its search for FAS Alignment. 0: Algorithm 1 1: Algorithm 2

TABLE 23: ALARM GENERATION REGISTER

REGISTER 8 **ALARM GENERATION REGISTER** **HEX ADDRESS: x00,08**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	Unused	RO	0	
6	LOF	R/W	0	**Loss of Frame Declaration Criteria** This Read/Write bit-field is used to select the LOF or Red Alarm generation criteria the Receive E1 Framer block will employ. 0:Receive E1 Framer declares Red Alarm unless both FAS and multi-frame alignment are achieved. 1: Prevents Receive E1 Framer from declaring Red Alarm condition; FAS Alignment is maintained.
5	YEL(1)	R/W	0	**Yellow Alarm and Multiframe Yellow Alarm Generation** These Read/Write bit-fields are used to select the FAS and Multi-Frame Yellow Alarm declaration and clearance criteria as described below. 00: Disables generation and transmission of yellow alarm 01: Enables automatic transmission of yellow alarms. i. The yellow alarm bit (bit 2 of the non-FAS frames in TSO) is transmitted by echoing the receive FAS alignment status. Logic 1 is transmitted if a loss of FAS alignment occurs. ii. The multiframe yellow alarm bit (bit 6 of frame 0 in TS16) is transmitted by echoing the receive CAS multiframe alignment status. Logic 1 transmitted if a loss of CAS multiframe occurs.
4	YEL(0)	R/W	0	
3	AISG(1)	R/W	0	**AIS Generation Select** These Read/Write bit-fields are used to configure the channel to generate and transmit an AIS pattern, as described below. 00: No AIS Alarm generated 01: Enable unframed AIS alarm generation 10: Enable AIS16 generation 11: Enable framed AIS alarm generation
2	AISG(0)	R/W	0	
1	AISD(1)	R/W	0	**AIS Pattern Detection Select** These Read/Write bit-fields are used to specify the type of AIS pattern that the receive E1 framer block will detect as described below. 00: Disabled 01: Unframed AIS alarm detection 10: AIS16 detection 11: Unframed AIS alarm detection
0	AISD(0)	R/W	0	

TABLE 24: SYNCHRONIZATION MUX REGISTER

REGISTER 9 | SYNCHRONIZATION MUX REGISTER | HEX ADDRESS: x00,09

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-5	Unused	RO	0	
4	SYNC INV	R/W	0	**Sync Inversion Select** Selects the direction of the transmit sync and multisync signals. 0: Syncs are input if the CSS(1:0) bits of CSR equal 01 (TxSerClk input is selected as the timing reference for the Transmit section of the framer); otherwise syncs are outputs 1: Syncs are output if CSS(1:0) bits of CSR equal 01 (TxSerClk input is selected as the timing reference for the Transmit section of the framer); otherwise syncs are inputs
3	DLSRC(1)	R/W	0	**Data Link Source Select** Specifies the source of the Data Link bits that will be inserted in the outbound E1 frames. 00: TxSer_n Input: Transmit Payload data Input port will be source of Data Link bits. 01: TX HDLC Controller: Transmit HDLC Controller will generate either BOS (Bit Oriented Signaling) or MOS (Message Oriented Signaling) messages which will be inserted into the Data Link bit-fields in the outbound E1 frames. 10: TxOH_n Input: Transmit Overhead data Input Port will be the source of the Data Link bits. 11: TxSer_n Input: Transmit Payload data Input port will be the source of the Data Link Bits.
2	DLSRC(0)	R/W	0	
1	CRCSRC	R/W	0	**CRC-4 Bits Source Select** This Read/Write bit-field is used to configure the transmit section of the channel to use either internal generation or the TxSER_n input pin as the source of the CRC-4 bits inserted into the outbound frames. 0: Internally Generated and inserted into E1 data stream internally. 1: Tx_Ser_n Input: Transmit Payload data Input port will be source of CRC-4 bits. **NOTE:** *This bit-field is ignored if CRC Multiframe Alignment is disabled*
0	FSRC	R/W	0	**Framing Alignment Bits Source Select** Specifies source of the Framing Alignment bits, which include FAS alignment bits, multiframe alignment bits, E and A bits. 0: Internally generated and inserted into the outbound E1 frames. 1: TxSer_n Input: Transmit Serial Input port will be source of the FAS bits, CRC Multiframe Alignments and the E and A bits.

TABLE 25: TRANSMIT SIGNALING AND DATA LINK SELECT REGISTER

REGISTER 10	TRANSMIT SIGNALING AND DATA LINK SELECT REGISTER	HEX ADDRESS:x00,0A

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	TxSa8ENB	R/W	0	Specifies if the Sa8 bit-field (bit 7 within timeslot 0 of non-FAS frames) will be involved in the transport of Data Link Information 0: Data Link Interface does not use Sa8 bit-field. Sa8 bit-field within each outbound non-FAS frame will be set to 1. 1: Data Link Interface uses Sa8 bit-field. ***NOTE:** This bit-field is only active when the SIGDL(2:0) bits within this register are set to 00x. This bit-field is ignored in all other case.*
6	TxSa7ENB	R/W	0	Specifies if the Sa7 bit-field (bit 6 within timeslot 0 of non-FAS frames) will be involved in the transport of Data Link Information 0: Data Link Interface does not use Sa7 bit-field. Sa7 bit-field within each outbound non-FAS frame will be set to 1. 1: Data Link Interface uses Sa7 bit-field. ***NOTE:** This bit-field is only active when the SIGDL(2:0) bits within this register are set to 00x. This bit-field is ignored in all other cases.*
5	TxSa6ENB	R/W	0	Specifies if the Sa6 bit-field (bit 5 within timeslot 0 of non-FAS frames) will be involved in the transport of Data Link Information 0: Data Link Interface does not use Sa6 bit-field. Sa6 bit-field within each outbound non-FAS frame will be set to 1. 1: Data Link Interface uses Sa6 bit-field. ***NOTE:** This bit-field is only active when the SIGDL(2:0) bits within this register are set to 00x. This bit-field is ignored in all other case.*
4	TxSa5ENB	R/W	0	Specifies if the Sa5 bit-field (bit 4 within timeslot 0 of non-FAS frames) will be involved in the transport of Data Link Information 0: Data Link Interface does not use Sa5 bit-field. Sa5 bit-field within each outbound non-FAS frame will be set to 1. 1: Data Link Interface uses Sa5 bit-field. ***NOTE:** This bit-field is only active when the SIGDL(2:0) bits within this register are set to 00x. This bit-field is ignored in all other case.*
3	TxSa4ENB	R/W	0	Specifies if the Sa4 bit-field (bit 3 within timeslot 0 of non-FAS frames) will be involved in the transport of Data Link Information 0: Data Link Interface does not use Sa4 bit-field. Sa4 bit-field within each outbound non-FAS frame will be set to 1. 1: Data Link Interface uses Sa4 bit-field. ***NOTE:** This bit-field is only active when the SIGDL(2:0) bits within this register are set to 00x. This bit-field is ignored in all other case.*

TABLE 25: TRANSMIT SIGNALING AND DATA LINK SELECT REGISTER

REGISTER 10 | TRANSMIT SIGNALING AND DATA LINK SELECT REGISTER | HEX ADDRESS:X00,0A

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
2	TxSIGDL(2)	R/W	0	These three Read/Write bits are used to specify the type of data that is to be transported via National Bits in timeslot 0 of the non-FAS frames and in Timeslot 16 in the outbound frames. The relationship between these bit fields and the role/function of the National and Timeslot 16 bits are presented below. *National Bits (Sa4-8)* 000: Data Link Data inserted into National bits 001: Data Link Data inserted into National bits 010: National bits forced to 1, not used to carry data link data 011: None (forced to 1) 1xx: None (forced to 1) *Timeslot 16* 000:PCM Data. Timeslot 16 data taken directly from PCM data input, could include signaling 001: CAS signaling bits A,B,C,D 010: PCM Data. Timeslot 16 data taken directly from PCM data input, could include signaling 011: CAS Signaling Bits A,B,C,D 1xx: HDLC Data Link. Common Channel Signaling enabled and timeslot 16 is taken from the Transmit HDLC Controller.
1	TxSIGDL(1)	R/W	0	
0	TxSIGDL(0)	R/W	0	

TABLE 26: FRAMING CONTROL REGISTER

REGISTER 11	FRAMING CONTROL REGISTER	HEX ADDRESS:x00,0B

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	RYSNC	R/W	0	**Force Re-Synchronization** A 0 to 1 transition in this bit-field forces the Receive E1 Framer to restart the synchronization process. This bit field is automatically cleared (set to 0) after frame synchronization is reached.
6	CASC(1)	R/W	0	**Loss of CAS Multiframe Alignment Criteria Select** These two Read/Write bits are used to select the Loss of CAS Multi-frame Alignment Declaration criteria. The relationship between the state of these two bit fields and the corresponding Loss of CAS Multi-Frame is presented below. 00: Two consecutive CAS Multi-Frames with Multiframe Alignment Signal (MAS) errors 01: Three consecutive CAS Multi-Frames with MAS errors 10: Four consecutive CAS Multi-Frames with MAS errors 11: Eight consecutive CAS Multi-Frames with MAS errors ***NOTE:** These bits are only active if Channel Associated Signaling is used.*
5	CASC(0)	R/W	0	
4	CRCC(1)	R/w	0	**Loss of CRC-4 Multiframe Alignment Criteria Select** Selects criteria for Loss of CRC-4 Multiframe Alignment. 00: Four consecutive CRC Multiframe Alignment signals have been received in error 01: Two consecutive CRC Multiframe Alignment signals have been received in error 10: Eight consecutive CRC Multiframe Alignment signals have been received in error 11: 915 or more CRC-4 errors have been detected in one second. ***NOTE:** These bit-fields are ignored if CRC Multiframe Alignment has been disabled.*
3	CRCC(0)	R/W	0	
2	FASC(2)	R/W	0	**Loss of FAS Alignment Criteria Select** These three Read/Write bits are used to select Loss of FAS Frame Declaration criteria. The relationship between the state of these bits and the corresponding Loss of FAS Frame declaration is presented below. 000: Illegal - do not use 001: 1 errored FAS pattern 010: 2 consecutive errored FAS patterns 011: 3 consecutive errored FAS patterns 100: 4 consecutive errored FAS patterns 101: 5 consecutive errored FAS patterns 110: 6 consecutive errored FAS patterns 111: 7 consecutive errored FAS patterns
1	FASC(1)	R/W	1	
0	FASC(0)	R/W	1	

TABLE 27: RECEIVE SIGNALING & DATA LINK REGISTER

REGISTER 12	RECEIVE SIGNALING & DATA LINK REGISTER	HEX ADDRESS:X00,0C

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	RxSa8ENB	R/W	0	This Read/Write bit is used to specify whether or not data link information will be transported via National Bit Sa8 (bit 7 within timeslot 0 of non-FAS frames) 0: Sa8 does not carry data link information 1: Sa8 carries data link information ***NOTE:** This bit-field is valid only if the TxSIGDL[2:0} = "000" or "001". (The National bits have been configured to carry data link bits).*
6	RxSa7ENB	R/w	0	This Read/Write bit is used to specify whether or not data link information will be transported via National Bit Sa7 (bit 6 within timeslot 0 of non-FAS frames) 0: Sa7 does not carry data link information 1: Sa7 carries data link information ***NOTE:** This bit-field is valid only if the TxSIGDL[2:0} = "000" or "001". (The National bits have been configured to carry data link bits).*
5	RxSa6ENB	R/W	0	This Read/Write bit is used to specify whether or not data link information will be transported via National Bit Sa6 (bit 5 within timeslot 0 of non-FAS frames) 0: Sa6 does not carry data Link information 1: Sa6 carries data link information ***NOTE:** This bit-field is valid only if the TxSIGDL[2:0} = "000" or "001". (The National bits have been configured to carry data link bits).*
4	RxSa5ENB	R/W	0	This Read/Write bit is used to specify whether or not data link information will be transported via National Bit Sa5 (bit 4 within timeslot 0 of non-FAS frames) 0: Sa5 does not carry data link information 1: Sa5 carries data link information ***NOTE:** This bit-field is valid only if the TxSIGDL[2:0} = "000" or "001". (The National bits have been configured to carry data link bits).*
3	RxSa4ENB	R/W	0	This Read/Write bit is used to specify whether or not data link information will be transported via National Bit Sa4 (bit 3 within timeslot 0 of non-FAS frames) 0: Sa4 does not carry data link information 1: Sa4 carries data link information ***NOTE:** This bit-field is valid only if the TxSIGDL[2:0} = "000" or "001". (If the National bits have been configured to carry data link bits).*
2	RxSIGDL(2)	R/W	0	These three Read/Write bits are used to configure the receive section of the channels on how to interpret the National and Timeslot 16 Bits. Specifies how signaling and data link information is received via the E1 Frames. *National Bits (Sa4-8)* 000: Data link data extracted from National bits 001: Data link data extracted from National bits 010: Data link data is not extracted from National bits 011: Data link data is not extracted from National bits 1xx: Data link data is not extracted from National bits *Timeslot 16 Bits* 000:PCM Data. 001: CAS Signal A,B,C,D 010: PCM Data. 011: CAS Signal A,B,C,D 1xx: Data Link (CCS). Timeslot 16 data is extracted by the Receive HDLC controller
1	RxSIGDL(1)	R/W	0	
0	RxSIGDL(0)	R/W	0	

TABLE 28: SIGNALING CHANGE REGISTER 0

REGISTER 13 | **SIGNALING CHANGE REGISTER 0** | **HEX ADDRESS: X00,0D**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	N/A	RO	0	These Reset Upon Read bits indicate whether the signaling data associated with Channels 1-7 has changed since the last read of this register. 0: Signaling data has not changed since last read of register 1: Signaling data has changed since last read of register ***NOTE:*** *This register is only relevant if the Framing Channel is using Channel Associated Signaling*
6	Ch. 1	RUR	0	
5	Ch.2	RUR	0	
4	Ch.3	RUR	0	
3	Ch.4	RUR	0	
2	Ch.5	RUR	0	
1	Ch.6	RUR	0	
0	Ch.7	RUR	0	

TABLE 29: SIGNALING CHANGE REGISTER 1

REGISTER 14 | **SIGNALING CHANGE REGISTER 1** | **HEX ADDRESS: X00,0E**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	Ch.8	RUR	0	These Reset Upon Read bits indicate whether the signaling data associated with Channels 8-15 has changed since the last read of this register. 0: Signaling data has not changed since last read of register 1: Signaling data has changed since last read of register ***NOTE:*** *This register is only relevant if the Framing Channel is using Channel Associated Signaling*
6	Ch.9	RUR	0	
5	Ch.10	RUR	0	
4	Ch.11	RUR	0	
3	Ch.12	RUR	0	
2	Ch.13	RUR	0	
1	Ch.14	RUR	0	
0	Ch.15	RUR	0	

TABLE 30: SIGNALING CHANGE REGISTER 2

REGISTER 15 | **SIGNALING CHANGE REGISTER 2** | **HEX ADDRESS:X00,0F**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	Ch.16	RUR	0	These Reset Upon Read bits indicate whether the signaling data associated with Channels 16-23 has changed since the last read of this register. 0: Signaling data has not changed since last read of register 1: Signaling data has changed since last read of register ***NOTE:*** *This register is only relevant if the Framing Channel is using Channel Associated Signaling*
6	Ch.17	RUR	0	
5	Ch.18	RUR	0	
4	Ch.19	RUR	0	
3	Ch.20	RUR	0	
2	Ch.21	RUR	0	
1	Ch.22	RUR	0	
0	Ch.23	RUR	0	

TABLE 31: SIGNALING CHANGE REGISTER 3

REGISTER 16 | **SIGNALING CHANGE REGISTER 3** | **HEX ADDRESS: x00,10**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	Ch.24	RUR	0	These Reset Upon Read bits indicate whether the signaling data associated with Channels 24-31 has changed since the last read of this register. 0: Signaling data has not changed since last read of register 1: Signaling data has changed since last read of register ***NOTE:*** *This register is only relevant if the Framing Channel is using Channel Associated Signaling*
6	Ch.25	RUR	0	
5	Ch.26	RUR	0	
4	Ch.27	RUR	0	
3	Ch.28	RUR	0	
2	Ch.29	RUR	0	
1	Ch.30	RUR	0	
0	Ch.31	RUR	0	

TABLE 32: RECEIVE NATIONAL BITS REGISTER

REGISTER 17 | **RECEIVE NATIONAL BITS REGISTER** | **HEX ADDRESS:x00,11**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	Si_FAS	RO	x	**Received International Bit - FAS Frame** This Read Only bit-field contains the value of the International Bit in the most recently received FAS frame
6	Si_nonFAS	RO	x	**Received International Bit - Non FAS Frame** This Read Only bit-field contains the value of the International Bit in the most recently received non-FAS frame
5	R_ALARM	RO	x	**Received FAS Yellow Alarm** This Read Only bit-field contains the value in the Remote Alarm bit-field (frame Yellow Alarm) within the non-FAS frame.
4	Sa4	RO	x	**Received National Bits** These Read Only bit-fields contain the values of the National bits within the most recently received non-FAS frame.
3	Sa5	RO	x	
2	Sa6	RO	x	
1	Sa7	RO	x	
0	Sa8	RO	x	

TABLE 33: RECEIVE EXTRA BITS REGISTER

REGISTER 18 | **RECEIVE EXTRA BITS REGISTER** | **HEX ADDRESS:x00,12**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	RO	0	
3	EX1	RO	x	**Extra Bit 1** Corresponds to value in bit 5 within timeslot 16 of frame 0 of the signaling multiframe
2	ALARMFE	RO	x	**CAS Multi-Frame Yellow Alarm** Corresponds to value in bit 6(CAS Multiframe Yellow Alarm) within timeslot 16 of frame 0 of the signaling multiframe. 0: Remote E1 transmitting terminal is not sending CAS Multiframe Yellow Alarm 1: Remote E1 transmitting terminal is sending CAS Multiframe Yellow Alarm
1	EX2	RO	x	**Extra Bit 2** Corresponds to value in Bit 7 within timeslot 16 of frame 0 of the signaling multiframe
0	EX3	RO	x	**Extra Bit 3** Corresponds to value in Bit 8 within timeslot 16 of frame 0 of the signaling multiframe

NOTE: *The value of the bit-fields within this register only have meaning if the framer is using Channel Associated Signaling.*

TABLE 34: DATA LINK CONTROL REGISTER

REGISTER 19	DATA LINK CONTROL REGISTER	HEX ADDRESS: X00,13

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	DIAG	R/W	0	Reserved for diagnostic purposes; set to 0 for normal operation
6	MOSA	R/W	0	**MOS Abort Enable/Disable Select** This Read/Write bit-field is used to configure the transmit HDLC controller to automatically transmit an abort sequence anytime it transitions from the MOS mode to the BOS mode. 0: Transmit HDLC Controller inserts an MOS abort sequence if the MOS message is interrupted 1: Prevents Transmit HDLC Controller from inserting an MOS abort sequence.
5	Rx_FCS_DIS	R/W	0	**Receive FCS Verification Disable** Enables/Disables Receive HDLC Controller's computation and verification of the FCS value in the incoming LAPD message frame 0: Verifies FCS value of each MOS frame. 1: Does not verify FCS value of each MOS frame.
4	AutoRx	R/W	0	**Auto Receive LAPD Message** Configures the Rx HDLC Controller to discard any incoming LAPD Message frame that exactly match which is currently stored in the Rx HDLC buffer. 1: Enables this feature.
3	Tx_ABORT	R/W	0	**Transmit ABORT** Configures the Tx HDLC Controller to transmit an ABORT sequence (string of 7 or more consecutive 1's) to the Remote terminal. 0: Tx HDLC Controller operates normally 1: Tx HDLC Controller inserts an ABORT sequence into the data link channel.
2	Tx_IDLE	R/W	0	**Transmit Idle (Flag Sequence Byte)** Configures the Tx HDLC controller to transmit a string of Flag Sequence octets (0X7E) in the data link channel to the Remote terminal. 0: Tx HDLC Controller resumes transmitting data to the Remote terminal 1: Tx HDLC Controller transmits a string of Flag Sequence bytes. ***NOTE:*** *This bit-field is ignored if the Tx HDLC controller is operating in the BOS Mode - bit-field 0(MOS/BOS) within this register is set to 0.*
1	Tx_FCS_EN	R/W	0	**Transmit LAPD Message with FCS** Configure HDLC Controller to include/not include FCS octets in the outbound LAPD message frames. 0: Does not include FCS octets into the outbound LAPD message frame. 1: Inserts FCS octets into the outbound LAPD message frame. ***NOTE:*** *This bit-field is ignored if the transmit HDLC controller has been configured to operate in the BOS mode.*
0	MOS/$\overline{\text{BOS}}$	R/W	0	**Message Oriented Signaling/Bit Oriented Signaling Select** Specifies whether the TxRx HDLC Controller will be transmitting and receiving LAPD message frames (MOS) or Bit Oriented Signal (BOS) messages. 0: Tx/Rx HDLC Controller transmits and receives BOS messages. 1: Tx/Rx HDLC Controller transmits and receives MOS messages.

TABLE 35: TRANSMIT DATA LINK BYTE COUNT REGISTER

REGISTER 20	TRANSMIT DATA LINK BYTE COUNT REGISTER	HEX ADDRESS:X00,13

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	BUFAVAL//BUFSEL	R/W	0	**Transmit HDLC Buffer Available/Buffer Select** Specifies which of the two Tx HDLC Buffers that the Tx HDLC controller should read from to generate the next outbound HDLC message. 0: transmits message data residing in Tx HDLC Buffer 0, indirect address 81h 1: transmits message data residing in Tx HDLC buffer 1, indirect address 83h ***NOTE:*** *If one of these Tx HDLC buffers contain a message which has yet to be completely read-in and processed for transmission by the Tx HDLC controller, then this bit-field will automatically reflect the value corresponding to the available buffer. Changing this bit-field to the in-use buffer is not permitted.*
6	TDLBC6	R/W	0	**Transmit HDLC Message - Byte Count** Depends on whether an MOS or BOS message is being transmitted to the Remote Terminal Equipment ***If BOS message is being transmitted*:** These bit fields contain the number of repetitions the BOS message must be transmitted before the Tx HDLC controller generates the TxEOT interrupt and halts transmission. If these fields are set to 00000000, then the BOS message will be transmitted for an indefinite number of times. ***If MOS message is being transmitted:*** These bit fields contain the length, in number of octets, of the message to be transmitted.
5	TDLBC5	R/W	0	
4	TDLBC4	R/W	0	
3	TDLBC3	R/W	0	
2	TDLBC2	R/W	0	
1	TDLBC1	R/W	0	
0	TDLBC0	R/W	0	

TABLE 36: RECEIVE DATA LINK BYTE COUNT REGISTER

REGISTER 21	RECEIVE DATA LINK BYTE COUNT REGISTER	HEX ADDRESS: X00,15

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	RBUFPTR	RO	0	**Receive HDLC Buffer-Pointer** Identifies which RxHDLC buffer contains the newly received HDLC message. 0: HDLC message is stored in Rx HDLC Buffer 0, indirect address 85h. 1: HDLC message is stored in Rx HDLC Buffer 1, indirect address 87h.
6	RDLBC6	RO	0	**Receive HDLC Message - byte count** These seven Read Only bit-fields contain the size in bytes of the HDLC message that has been extracted and written into the Rx HDLC buffer. ***NOTE:*** *These bit-fields are only valid if the RxHDLC Controller is operating in the MOS mode.*
5	RDLBC5	RO	0	
4	RDLBC4	RO	0	
3	RDLBC3	RO	0	
2	RDLBC2	RO	0	
1	RDLBC1	RO	0	
0	RDLBC0	RO	0	

TABLE 37: SLIP BUFFER CONTROL REGISTER

REGISTER 22 | **SLIP BUFFER CONTROL REGISTER** | **HEX ADDRESS: X00,16**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-5	Unused	RO	0	
4	SB_FORCESF	R/W	0	Force Signaling Freeze
3	SB_SFENB	R/W	o	Signal Freeze Enable
2	SB_SDIR	R/W	1	**Slip Buffer (RxSync) Direction Select** Allows RxSync output pin to be an input or an output. 0: RxSync is an output pin 1: RxSync is an input pin
1	SB_ENB(1)	R/w	0	**Slip Buffer Mode Select** Selects mode of operation of slip buffer. 00: Buffer bypassed 01: Elastic store enabled 10: Buffer acts as FIFO Data latency dictated by the setting within the FIFO Latency Register 11: Buffer is bypassed
0	SB_ENB(0)	R/W	0	

TABLE 38: FIFO LATENCY REGISTER

REGISTER 23 | **FIFO LATENCY REGISTER** | **HEX ADDRESS: X00,17**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-5	Unused	RO	0	
4-0	Latency	R/W	0	Sets the distance between slip buffer read and slip buffer write.

TABLE 39: DMA 0 (WRITE) CONFIGURATION REGISTER

REGISTER 24 | DMA 0 WRITE CONFIGURATION REGISTER | HEX ADDRESS: X00,18

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-6	Unused	RO	0	
5	DMA0 RST	R/W	0	**DMA_0 Reset** Resets the transmit or write DMA channel.
4	DMA0 ENB	R/W	0	**DMA_0 Enable** Enables DMA_0 interface. 0: Disables DMA_0 interface 1: Enables DMA_0 interface
3	WR TYPE	R/W	0	**Write Type Select** Selects function of $\overline{WR}$ signal. 0: $\overline{WR}$ functions as direction signal (indicates whether the current bus cycle is a read or write operation) and $\overline{RD}$ functions as a data strobe signal. 1: $\overline{WR}$ functions as a write strobe signal and $\overline{RD}$ functions as configured in the DMA 1 configuration register.
2	unused	RO	0	
1	DMA0_CHAN(1)	R/W	0	Selects which framer, within the chip, is to use the DMA_0 (Write) interface. 00: Framer 0 01: Framer 1 01: Framer 2 11: Framer 3
0	DMA0_CHAN(0)	R/W	0	

TABLE 40: DMA 1 (READ) CONFIGURATION REGISTER

REGISTER 25 | DMA 1 (READ) CONFIGURATION REGISTER | HEX ADDRESS: X00,19

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-6	Unused	RO	0	
5	DMA1 RST	R/W	0	**DMA_1 Reset** Resets the receive (or Read) DMA Channel
4	DMA1 ENB	R/W	0	DMA1_ENB Enables DMA_1 interface 0: Disables DMA_1 interface 1: Enables DMA_1 interface
3	RD TYPE	R/W	0	Selects the function of pRD_L signal. 0: $\overline{RD}$ functions as Read Strobe signal 1:
2	unused	RO	0	
1	DMA1_CHAN(1)	R/W	0	**Framer Select** Selects which framer, within the chip, is to use the DMA_1 interface. 00: Framer 0 01: Framer 1 10: Framer 2 11: Framer 3
0	DMA1_CHAN(0)	R/W	0	

TABLE 41: INTERRUPT CONTROL REGISTER

REGISTER 26 | **INTERRUPT CONTROL REGISTER** | **HEX ADDRESS: X00,1A**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-3	Unused	RO	0	
2	INT_WC_RUR	R/W	0	**Interrupt Write-to-Clear or Reset-upon-Read Select** Configures Interrupt Status bits to either RUR or Write-to-Clear 0: Interrupt Status bit RUR 1: Interrupt Status bit Write-to-Clear
1	ENBCLR	R/W	0	**Interrupt Enable Auto Clear** 0: Interrupt Enable bits are not cleared after status reading 1: Interrupt Enable bits are cleared after status reading
0	INTRUP_ENB	R/W	0	**Interrupt Enable for Framer_n** Enables Framer n within the XRT84V24 for Interrupt Generation. 0: Disables corresponding E1 framer block for Interrupt Generation 1: Enables corresponding E1 framer block for Interrupt Generation

TABLE 42: TRANSMIT CHANNEL CONTROL REGISTER 0

REGISTER 27 | **TRANSMIT CHANNEL CONTROL REGISTER 0** | **HEX ADDRESS:X2,00**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R	0	
3-0	TxCond(3:0)	R/W	0	

TABLE 43: TRANSMIT CHANNEL CONTROL REGISTER 1

REGISTER 28 — **TRANSMIT CHANNEL CONTROL REGISTER-1** — **HEX ADDRESS: x2,01**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	RO	0	
3-0	TxCond(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 1** Replaces the contents of timeslot 1 octet (PCM data within the next outbound E1 frame) with signaling codes as follows. **0x00:** Contents of timeslot octet unchanged prior to transmission to Remote Terminal Equipment. Contents are transmitted without modification as received via the TxSer_n input pin. **0x01:** All 8 bits of the timeslot octet are inverted (1's complement) prior to transmission to the Remote Terminal Equipment. This selection is equivalent to executing the following logic operation with each timeslot 1 octet: TX_TIME_SLOT_OCTET=(TE_TIME_SLOT_OCTET) XOR 0xFF **0x02:** The even bits of the timeslot octet are inverted prior to trans mission to the Remote Terminal Equipment. This selection is equivalent to executing the following logic operation: TX_TIME_SLOT_OCTET=(TE_TIME_SLOT_OCTET) XOR 0xAA **03h:** The odd bits of the time slot octet are inverted prior to trans mission to the Remote Terminal Equipment. This selection is equivalent to executing the following logic operation: TX_TIME_SLOT_OCTET=(TE_TIME_SLOT_OCTET) XOR 0x55 **04h**: The contents of the timeslot octet will be substituted with the 8 -bit value in User Code Register 1, address x2,21, prior to transmission to the Remote Terminal Equipment. **05h:** The contents of the timeslot octet will be substituted with the value 0xFF (BUSY) prior to transmission to the Remote Termi nal Equipment. **06h:** The contents of the timeslot octet will be substituted with the value 0xD5 (VACANT 0V) prior to transmission to the Remote Terminal Equipment. **07h:** The BUSY TS(111#_####) code replaces the input data for transmission. **08h:** The BUSY 00 code replaces the input data for transmission **09h:** The A-Law Digital Milliwatt pattern replaces the input data for transmission. **0Ah:** The u-Law Digital Milliwatt pattern replaces the input data for transmission. **0Bh:** The contents of the timeslot octet will be transmitted without **0Fh:** modification to the Remote Terminal Equipment.

TABLE 44: TRANSMIT CHANNEL CONTROL REGISTER -2

REGISTER 29 — **TRANSMIT CHANNEL CONTROL REGISTER -2** — **HEX ADDRESS x2,02**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 2** Please refer to the Transmit Channel Control Register 1 (x2,01) for descrip-tion.

TABLE 45: TRANSMIT CONTROL REGISTER - 3

REGISTER 30 | **TRANSMIT CONTROL REGISTER -3** | **HEX ADDRESS: X2,03**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/w	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 3** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 46: TRANSMIT CONTROL REGISTER -4

REGISTER 31 | **TRANSMIT CONTROL REGISTER -4** | **HEX ADDRESS: X2,04**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 4** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 47: TRANSMIT CONTROL REGISTER - 5

REGISTER 32 | **TRANSMIT CONTROL REGISTER - 5** | **HEX ADDRESS: X2,05**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 5** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 48: TRANSMIT CONTROL REGISTER - 6

REGISTER 33 | **TRANSMIT CONTROL REGISTER - 6** | **HEX ADDRESS: X2,06**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 6** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

Table 49: Transmit Control Register - 7

Register 34 | **Transmit Control Register - 7** | **Hex Address: x2,07**

Bit	Function	Type	Default	Description-Operation
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 7** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

Table 50: Transmit Channel Control Register - 8

Register 35 | **Transmit Channel Control Register - 8** | **Hex Address: x2,08**

Bit	Function	Type	Default	Description-Operation
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 8** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

Table 51: Transmit Channel Control Register - 9

Register 36 | **Transmit Channel Control Register - 9** | **Hex Address: x0,09**

Bit	Function	Type	Default	Description-Operation
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 9** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

Table 52: Transmit Channel Control Register - 10

Register 37 | **Transmit Channel Control Register - 10** | **Hex Address: x0,0A**

Bit	Function	Type	Default	Description-Operation
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 10** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 53: TRANSMIT CHANNEL CONTROL REGISTER - 11

REGISTER 38 — TRANSMIT CHANNEL CONTROL REGISTER - 11 — HEX ADDRESS: X2,0B

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 11** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 54: TRANSMIT CHANNEL CONTROL REGISTER - 12

REGISTER 39 — TRANSMIT CHANNEL CONTROL REGISTER - 12 — HEX ADDRESS: X2,0C

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 12** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 55: TRANSMIT CHANNEL CONTROL REGISTER - 13

REGISTER 40 — TRANSMIT CHANNEL CONTROL REGISTER - 13 — HEX ADDRESS: X2,0D

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/w	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 13** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 56: TRANSMIT CHANNEL CONTROL REGISTER - 14

REGISTER 41 — TRANSMIT CHANNEL CONTROL REGISTER - 14 — HEX ADDRESS: X2,0E

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/w	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 14** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 57: TRANSMIT CHANNEL CONTROL REGISTER - 15

REGISTER 42 | **TRANSMIT CHANNEL CONTROL REGISTER - 15** | **HEX ADDRESS: X2,0F**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/w	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 15** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 58: TRANSMIT CHANNEL CONTROL REGISTER - 16

REGISTER 43 | **TRANSMIT CHANNEL CONTROL REGISTER - 16** | **HEX ADDRESS: 0X2,10**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/w	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 16** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 59: TRANSMIT CONTROL CHANNEL REGISTER - 17

REGISTER 44 | **TRANSMIT CONTROL CHANNEL REGISTER - 17** | **HEX ADDRESS: X2,11**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 17** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 60: TRANSMIT CHANNEL CONTROL REGISTER - 18

REGISTER 45 | **TRANSMIT CHANNEL CONTROL REGISTER - 18** | **HEX ADDRESS: X2,12**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 18** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 61: TRANSMIT CHANNEL CONTROL REGISTER - 19

REGISTER 46 | TRANSMIT CHANNEL CONTROL REGISTER - 19 | HEX ADDRESS:X2,13

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 19** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 62: TRANSMIT CHANNEL CONTROL REGISTER - 20

REGISTER 47 | TRANSMIT CHANNEL CONTROL REGISTER - 20 | HEX ADDRESS: X2,14

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 20** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 63: TRANSMIT CHANNEL CONTROL REGISTER - 21

REGISTER 48 | TRANSMIT CHANNEL CONTROL REGISTER - 21 | HEX ADDRESS: X2,15

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 21** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 64: TRANSMIT CHANNEL CONTROL REGISTER - 22

REGISTER 49 | TRANSMIT CHANNEL CONTROL REGISTER - 22 | HEX ADDRESS: X2,16

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 22** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 65: TRANSMIT CHANNEL CONTROL REGISTER - 23

REGISTER 50 **TRANSMIT CHANNEL CONTROL REGISTER - 23** **HEX ADDRESS: X2,17**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 23** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 66: TRANSMIT CHANNEL CONTROL REGISTER - 24

REGISTER 51 **TRANSMIT CHANNEL CONTROL REGISTER - 24** **HEX ADDRESS: X2,18**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 24** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 67: TRANSMIT CHANNEL CONTROL REGISTER - 25

REGISTER 52 **TRANSMIT CHANNEL CONTROL REGISTER** **HEX ADDRESS: X2,19**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 25** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 68: TRANSMIT CHANNEL CONTROL REGISTER - 26

REGISTER 53 **TRANSMIT CHANNEL CONTROL REGISTER - 26** **HEX ADDRESS: X2,1A**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/w	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 26** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 69: TRANSMIT CHANNEL CONTROL REGISTER - 27

REGISTER 54 TRANSMIT CHANNEL CONTROL REGISTER - 27 HEX ADDRESS: X2,1B

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 27** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 70: TRANSMIT CHANNEL CONTROL REGISTER - 28

REGISTER 55 TRANSMIT CHANNEL CONTROL REGISTER - 28 HEX ADDRESS: X2,1C

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 28** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 71: TRANSMIT CHANNEL CONTROL REGISTER - 29

REGISTER 56 TRANSMIT CHANNEL CONTROL REGISTER - 29 HEX ADDRESS: X2,1D

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 29** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 72: TRANSMIT CHANNEL CONTROL REGISTER - 30

REGISTER 57 TRANSMIT CHANNEL CONTROL REGISTER - 30 HEX ADDRESS: X2,1E

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/W	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 30** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 73: TRANSMIT CHANNEL CONTROL REGISTER - 31

REGISTER 58	TRANSMIT CHANNEL CONTROL REGISTER - 31	HEX ADDRESS:X2,1F

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/w	0	
3-0	TxCOND(3:0)	R/W	0	**Transmit Channel Conditioning for Timeslot 31** Please refer to the Transmit Channel Control Register 1 (x2,01) for description.

TABLE 74: USER CODE REGISTER 0

REGISTER 59	USER CODE REGISTER 0	HEX ADDRESS: X2,20

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 75: USER CODE REGISTER 1

REGISTER 60	USER CODE REGISTER 1	HEX ADDRESS:X2,21

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 76: USER CODE REGISTER 2

REGISTER 61	USER CODE REGISTER 2	HEX ADDRESS:X2,22

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 77: USER CODE REGISTER 3

REGISTER 62	USER CODE REGISTER 3	HEX ADDRESS: X2,23

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/w	0	

TABLE 78: USER CODE REGISTER 4

REGISTER 63	USER CODE REGISTER 4	HEX ADDRESS: X2,24

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 79: USER CODE REGISTER 5

REGISTER 64 | **USER CODE REGISTER 5** | **HEX ADDRESS: X2,25**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 80: USER CODE REGISTER 6

REGISTER 65 | **USER CODE REGISTER 6** | **HEX ADDRESS: X2,26**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 81: USER CODE REGISTER 7

REGISTER 66 | **USER CODE REGISTER 7** | **HEX ADDRESS: X2,27**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 82: USER CODE REGISTER 8

REGISTER 67 | **USER CODE REGISTER 8** | **HEX ADDRESS: X2,28**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 83: USER CODE REGISTER 9

REGISTER 68 | **USER CODE REGISTER 9** | **HEX ADDRESS: X2, 29**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 84: USER CODE REGISTER 10

REGISTER 69 | **USER CODE REGISTER 10** | **HEX ADDRESS: X2,2A**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 85: USER CODE REGISTER 11

REGISTER 70 | **USER CODE REGISTER 11** | **HEX ADDRESS: X2,2B**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/w	0	

TABLE 86: USER CODE REGISTER 12

REGISTER 71 | **USER CODE REGISTER 12** | **HEX ADDRESS: X2,2C**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 87: USER CODE REGISTER 13

REGISTER 72 | **USER CODE REGISTER 13** | **HEX ADDRESS: X2,2D**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 88: USER CODE REGISTER 14

REGISTER 73 | **USER CODE REGISTER 14** | **HEX ADDRESS: X2,2E**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 89: USER CODE REGISTER 15

REGISTER 74 | **USER CODE REGISTER 15** | **HEX ADDRESS: X2,2F**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 90: USER CODE REGISTER 16

REGISTER 75 | **USER CODE REGISTER 16** | **HEX ADDRESS: X2,30**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 91: USER CODE REGISTER 17

REGISTER 76 — USER CODE REGISTER 17 — HEX ADDRESS: X2,31

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 92: USER CODE REGISTER 18

REGISTER 77 — USER CODE REGISTER 18 — HEX ADDRESS: X2,32

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 93: USER REGISTER 19

REGISTER 78 — USER CODE REGISTER 19 — HEX ADDRESS: X2,33

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 94: USER CODE REGISTER 20

REGISTER 79 — USER CODE REGISTER 20 — HEX ADDRESS: X2,34

BIT	FUNCTION	Type	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 95: USER CODE REGISTER 21

REGISTER 80 — USER CODE REGISTER 21 — HEX ADDRESS: X2,35

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 96: USER CODE REGISTER 22

REGISTER 81 — USER CODE REGISTER 22 — HEX ADDRESS: X2,36

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 97: USER CODE REGISTER 23

REGISTER 82 **USER CODE REGISTER 23** **HEX ADDRESS: X2,37**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 98: USER CODE REGISTER 24

REGISTER 83 **USER CODE REGISTER 24** **HEX ADDRESS: X2,38**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 99: USER CODE REGISTER 25

REGISTER 84 **USER CODE REGISTER 25** **HEX ADDRESS: X2,39**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W		

TABLE 100: USER CODE REGISTER 26

REGISTER 85 **USER CODE REGISTER 26** **HEX ADDRESS: X2,3A**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 101: USER CODE REGISTER 27

REGISTER 86 **USER CODE REGISTER 27** **HEX ADDRESS: X2,3B**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 102: USER CODE REGISTER 28

REGISTER 87 **USER CODE REGISTER 28** **HEX ADDRESS:X2,3C**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 103: USER CODE REGISTER 29

REGISTER 88 | USER CODE REGISTER 30 | HEX ADDRESS: X2,3D

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 104: USER CODE REGISTER 30

REGISTER 89 | USER CODE REGISTER 30 | HEX ADDRESS: X2,3E

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 105: USER CODE REGISTER 31

REGISTER 90 | USER CODE REGISTER 31 | HEX ADDRESS: X2,3F

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 106: RECEIVE SIGNALING ARRAY REGISTER 0

REGISTER 91 | RECEIVE SIGNALING ARRAY REGISTER 0 | HEX ADDRESS: X4,00

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0		R/W	0	

TABLE 107: RECEIVE SIGNALING ARRAY REGISTER 1

REGISTER 92 | RECEIVE SIGNALING ARRAY REGISTER 1 | HEX ADDRESS: X4,01

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 1. **NOTE:** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 108: RECEIVE SIGNALING ARRAY REGISTER 2

REGISTER 93 — **RECEIVE SIGNALING ARRAY REGISTER 2** — **HEX ADDRESS: x4,02**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 2. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 109: RECEIVE SIGNALING ARRAY REGISTER 3

REGISTER 94 — **RECEIVE SIGNALING ARRAY REGISTER 3** — **HEX ADDRESS:x4,03**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 3. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 110: RECEIVE SIGNALING ARRAY REGISTER 4

REGISTER 95 — **RECEIVE SIGNALING ARRAY REGISTER 4** — **HEX ADDRESS: x4,04**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 4. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 111: RECEIVE SIGNALING ARRAY REGISTER 5

REGISTER 96 | **RECEIVE SIGNALING ARRAY REGISTER 5** | **HEX ADDRESS: x4,05**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 5. **NOTE:** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 112: RECEIVE SIGNALING ARRAY REGISTER 6

REGISTER 97 | **RECEIVE SIGNALING ARRAY REGISTER 6** | **HEX ADDRESS: x4,06**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 6. **NOTE:** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 113: RECEIVE SIGNALING ARRAY REGISTER 7

REGISTER 98 | **RECEIVE SIGNALING ARRAY REGISTER 7** | **HEX ADDRESS: x4,07**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 7. **NOTE:** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 114: RECEIVE SIGNALING ARRAY REGISTER 8

REGISTER 99 | **RECEIVE SIGNALING ARRAY REGISTER 8** | **HEX ADDRESS: X4,08**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 8. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 115: RECEIVE SIGNALING ARRAY REGISTER 9

REGISTER 100 | **RECEIVE SIGNALING ARRAY REGISTER 9** | **HEX ADDRESS: X4,09**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 9. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 116: RECEIVE SIGNALING ARRAY REGISTER 10

REGISTER 101 | **RECEIVE SIGNALING ARRAY REGISTER 10** | **HEX ADDRESS: X4,0A**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 10. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 117: RECEIVE SIGNALING ARRAY REGISTER 11

REGISTER 102 **RECEIVE SIGNALING ARRAY REGISTER 11** **HEX ADDRESS: X4,0B**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 11 ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 118: RECEIVE SIGNALING ARRAY REGISTER 12

REGISTER 103 **RECEIVE SIGNALING ARRAY REGISTER 12** **HEX ADDRESS: X4,0C**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 12. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 119: RECEIVE SIGNALING ARRAY REGISTER 13

REGISTER 104 **RECEIVE SIGNALING ARRAY REGISTER 13** **HEX ADDRESS: X4,0D**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 13. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

Table 120: Receive Signaling Array Register 14

Register 105 | **Receive Signaling Array Register 14** | **Hex Address: x4,0E**

Bit	Function	Type	Default	Description-Operation
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 14. ***Note:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

Table 121: Receive Signaling Array Register 15

Register 106 | **Receive Signaling Array Register 15** | **Hex Address:x4,0F**

Bit	Function	Type	Default	Description-Operation
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 15. ***Note:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

Table 122: Receive Signaling Array Register 16

Register 107 | **Receive Signaling Array Register 16** | **Hex Address: x4,10**

Bit	Function	Type	Default	Description-Operation
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 16. ***Note:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

Table 123: Receive Signaling Array Register 17

Register 108 | **Receive Signaling Array Register 17** | **Hex Address: x4,11**

Bit	Function	Type	Default	Description-Operation
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 17. **Note:** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

Table 124: Receive Signaling Array Register 18

Register 109 | **Receive Signaling Array Register 18** | **Hex Address: x4,12**

Bit	Function	Type	Default	Description-Operation
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 18. **Note:** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

Table 125: Receive Signaling Array Register 19

Register 110 | **Receive Signaling Array Register 19** | **Hex Address: x4,13**

Bit	Function	Type	Default	Description-Operation
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 19. **Note:** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 126: RECEIVE SIGNALING ARRAY REGISTER 20

REGISTER 111 **RECEIVE SIGNALING ARRAY REGISTER 20** **HEX ADDRESS: X4,14**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 20. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 127: RECEIVE SIGNALING ARRAY REGISTER 21

REGISTER 112 **RECEIVE SIGNALING ARRAY REGISTER 21** **HEX ADDRESS: X4,15**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 21. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 128: RECEIVE SIGNALING ARRAY REGISTER 22

REGISTER 113 **RECEIVE SIGNALING ARRAY REGISTER 22** **HEX ADDRESS: X4,16**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 22. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 129: RECEIVE SIGNALING ARRAY REGISTER 23

REGISTER 114 **RECEIVE SIGNALING ARRAY REGISTER 23** **HEX ADDRESS: X4,17**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 23. ***NOTE:*** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 130: RECEIVE SIGNALING ARRAY REGISTER 24

REGISTER 115	RECEIVE SIGNALING ARRAY REGISTER 24	HEX ADDRESS: X4,18

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 24. *NOTE: The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

.

TABLE 131: RECEIVE SIGNALING ARRAY REGISTER 25

REGISTER 116	RECEIVE SIGNALING ARRAY REGISTER 25	HEX ADDRESS: X4,19

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 25. *NOTE: The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 132: RECEIVE SIGNALING ARRAY REGISTER 26

REGISTER 117	RECEIVE SIGNALING ARRAY REGISTER 26	HEX ADDRESS: X4,1A

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 26. *NOTE: The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 133: RECEIVE SIGNALING ARRAY REGISTER 27

REGISTER 118 | **RECEIVE SIGNALING ARRAY REGISTER 27** | **HEX ADDRESS: X4,1B**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 27. ***NOTE:** The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 134: RECEIVE SIGNALING ARRAY REGISTER 28

REGISTER 119 | **RECEIVE SIGNALING ARRAY REGISTER 28** | **HEX ADDRESS: X4,1C**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 28. ***NOTE:** The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 135: RECEIVE SIGNALING ARRAY REGISTER 29

REGISTER 120 | **RECEIVE SIGNALING ARRAY REGISTER 29** | **HEX ADDRESS: X4,1D**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 29. ***NOTE:** The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 136: RECEIVE SIGNALING ARRAY REGISTER 30

REGISTER 121 — **RECEIVE SIGNALING ARRAY REGISTER 30** — **HEX ADDRESS: X4,1E**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 30. **NOTE:** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 137: RECEIVE SIGNALING ARRAY REGISTER 31

REGISTER 122 — **RECEIVE SIGNALING ARRAY REGISTER 31** — **HEX ADDRESS: X4,1F**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4		RO	0	
3	A	R/W	0	Reflects the most recently received signaling value (A,B,C,D) associated with timeslot 31. **NOTE:** *The content of this register only has meaning when the framer is using Channel Associated Signaling.*
2	B	R/W	0	
1	C	R/W	0	
0	D	R/W	0	

TABLE 138: PMON E1 RECEIVE LINE CODE (BIPOLAR) VIOLATION COUNTER

REGISTER 123 — **PMON E1 RECEIVE LINE CODE (BIPOLAR) VIOLATION COUNTER -LSB** — **HEX ADDRESS X8,00**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	RO	0	
3-0	RxLCV Count - Low Byte	RUR	0	These four Reset Upon Read bits along with PMON E1 Receive Line Code Violation Counter - MSB, provides a 12-bit representation of the number of LIne Code violations that have been detected by the Receive E1 Framer block since the last read of these registers. Lower 8 bits. This register contains the lowest four bits within this 12 bit expression

NOTE:

TABLE 139: PMON E1 RECEIVE LINE CODE (BIPOLAR) VIOLATION COUNTER

REGISTER 124 — **PMON E1 RECEIVE LINE CODE (BIPOLAR) VIOLATION COUNTER - MSB** — **HEX ADDRESS: X8,01**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	RxLCV Count - High Byte	RUR	0	These eight Reset Upon Read bits along with PMON E1 Receive Line Code Violation Counter - LSB, provides 12-bit representation of the number of LIne Code violations that have been detected by Receive E1 Framer Block since the last read of these registers. Upper 4 bits. This register contains the upper 8 bits within this 12 bit expression.

NOTE: .

TABLE 140: PMON E1 RECEIVE FRAMING ALIGNMENT BIT ERROR COUNTER

REGISTER 125 **PMON E1 RECEIVE FRAMING ALIGNMENT ERROR COUNTER - LSB** **HEX ADDRESS: x8,02**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-4	Unused	R/O	0	
3-0	Framing Alignment Error Count - Low Byte	RUR	0	These four Reset Upon Read bits along with PMON E1 Receive Framing Alignment Bit Error Counter- MSB, provides 12-bit representation of the number of Framing Alignment errors that have been detected by Receive E1 Framer Block since the last read of these register. This register contains the lowest four bits within this 12-bit expression

TABLE 141: PMON E1 RECEIVE FRAMING ALIGNMENT BIT ERROR COUNTER

REGISTER 126 **PMON E1 RECEIVE FRAMING ALIGNMENT BIT ERROR COUNTER - MSB** **HEX ADDRESS: x8,03**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	Framing Alignment Error Count - High Byte	RUR	0	These eight Reset Upon Read bits along with PMON E1 Receive Framing Alignment Bit Error Counter- LSB, provides a 12-bit representation of the number of Framing Alignment errors that have been detected by Receive E1 Framer number n since the last read of these registers. This register contains the upper 8bits within this 12-bit expression.

TABLE 142: PMON E1 RECEIVE SEVERELY ERRORED FRAME COUNTER

REGISTER 127 **PMON E1 RECEIVE SEVERELY ERRORED FRAME COUNTER** **HEX ADDRESS: x8,04**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	Severely Errored Frame Count	RUR	0	Severely Errored 8-bit frame accumulation Counter ***Note: A severely errored frame event is defined as the occurrence of two consecutive errored frame alignment signals that are not responsible for loss of frame alignment.***

TABLE 143: PMON E1 RECEIVE CRC-4 BLOCK ERROR COUNTER - LSB

REGISTER 128 **PMON E1 RECEIVE CRC-4 BLOCK ERROR COUNTER - LSB** **HEX ADDRESS: x8,05**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-2	Unused	RO	0	
1-0	CRC-4 Block Error Count - Low Byte	RUR	0	These two Reset upon Read bits along with PMON E1 Receive CRC-4 Block Error Counter - MSB, provides a 10-bit representation of the number of CRC-4 Block errors that have been detected by a Receive E1 Framer Block since the last read of these registers. This register contains the lower two bits within this 10 bit expression. ***Note: Counter contains the 10-bit synchronization bit error event. A synchronization bit error event is defined as a CRC-4 error received. Counter is disabled during loss of sync at either the Frame/FAS or ESF/CRC4 level, but it will not be disabled if loss of multiframe sync occurs at the CAS level.***

TABLE 144: PMON E1 RECEIVE CRC-4 BLOCK ERROR COUNTER - MSB

REGISTER 129 | **PMON E1 RECEIVE CRC-4 BLOCK ERROR COUNTER - MSB** | **HEX ADDRESS: x8,06**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	CRC-4 Block Error Count - High Byte	RUR	0	These eight Reset Upon Read bits along with PMON E1 Receive CRC-4 Block Error Counter - LSB, provides a 10-bit representation of the number of CRC-4 Block errors detected by Receive E1 Framer Block since the last read of these registers. This register contains the upper eight bits of this 10 bit expression

TABLE 145: PMON E1 RECEIVE FAR END BLOCK ERROR COUNTER

REGISTER 130 | **PMON E1 RECEIVE FAR END BLOCK ERROR COUNTER - LSB** | **HEX ADDRESS: x8,07**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-2	Unused	RO	0	
1-0	Far-End Block Error Count -Low Byte	RUR	0	These two Reset Upon Read bits along with PMON E1 Receive Far-End Block Error Counter - MSB, provides a 10-bit representation of the number of Far End Block Error events that have been detected by the Receive E1 Framer Block since the last read of these registers. This register contains the lower two bits within this 10 bit expression. ***Note: Counter contains the 10-bit far-end block error event. Counter will increment once each time the received E-bit is set to zero. The counter is disabled during loss of sync at either the FAS or CRC-4 level and it will continue to count if loss of multiframe sync occurs at the CAS level.***

TABLE 146: PMON E1 RECEIVE FAR-END BLOCK ERROR COUNTER - MSB

REGISTER 131 | **PMON E1 RECEIVE FAR-END BLOCK ERROR COUNTER - MSB** | **HEX ADDRESS: x8,08**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	Far-End Block Error Count - High Byte	RUR	0	These eight Reset Upon Read bits along with PMON E1 Receive Far-End Block Error Counter - LSB, provides a 10-bit representation of the number of Far End Block Error events that have been detected by the Receive E1 Framer Block since the last read of these registers. This register contains the upper eight bits within this 10 bit expression.

TABLE 147: PMON E1 RECEIVE SLIP COUNTER

REGISTER 132 | **PMON E1 RECEIVE SLIP COUNTER** | **HEX ADDRESS x8,09**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	Slip Count	RUR	0	Note: counter contains the 8-bit receive buffer slip event. A slip event is defined as a replication or deletion of a T1/E1 frame by the receiving slip buffer. ***Note: A 12 bit counter which counts the occurrence of a bipolar violation on the receive data line. This counter is of sufficient length so that the probability of counter saturation over a one second interval at a 10 -3-Bit Error Rate (BER) is less than 0.001%.***

TABLE 148: PMON E1 RECEIVE LOSS OF FRAME COUNTER

REGISTER 133 **PMON E1 RECEIVE LOSS OF FRAME COUNTER** **HEX ADDRESS: X8,0A**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	Loss of Frame Counts	RUR	0	***Note: LOFC is a count of the number of times a "Loss Of FAS Frame" has been declared. This counter provides the capability to measure an accumulation of short failure events.***

TABLE 149: PMON E1 RECEIVE CHANGE OF FRAME ALIGNMENT COUNTER

REGISTER 134 **PMON E1 RECEIVE CHANGE OF FRAME ALIGNMENT COUNTER** **HEX ADDRESS: X8,0B**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	COFA Count	RUR	0	Change of Frame Alignment Accumulation counter. ***Note: COFA is declared when the newly-locked framing is different from the one offered by off-line framer.***

TABLE 150: PMON LAPD E1 FRAME CHECK SEQUENCE ERROR COUNTER

REGISTER 135 **PMON LAPD E1 FRAME CHECK SEQUENCE ERROR COUNTER** **HEX ADDRESS: X8,0C**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-0	FCS Error Count	RUR	0	Frame Check Sequence error Accumulation Counter. ***Note: Counter accumulates the times of occurrence of receive frame check sequence error detected by LAPD controller.***

TABLE 151: BLOCK INTERRUPT STATUS REGISTER

REGISTER 136	BLOCK INTERRUPT STATUS REGISTER	HEX ADDRESS: xA,02

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-6	Unused	RO	0	
5	RxLine Clk Loss	RUR	0	**RxClk Los Interrupt Status** Indicates if Framer n has experienced a Loss of Recovered Clock interrupt since last read of this register. 0: Loss of Recovered Clock interrupt has not occurred since last read of this register 1: Loss of Recovered Clock interrupt has occurred since last read of this register.
4	One Sec Status	RUR	0	**One Second Interrupt Status** Indicates if the XRT84V24 has experienced a One Second interrupt since the last read of this register. 0: No outstanding One Second interrupts awaiting service 1: Outstanding One Second interrupt awaits service
3	HDLC Status	RO	0	**HDLC Block Interrupt Status** Indicates if the HDLC block has an interrupt request awaiting service. 0: No outstanding interrupt requests awaiting service 1: HDLC Block has an interrupt request awaiting service. Interrupt Service routine should branch to and read Data LInk Status Register (address xA,06). ***NOTE:*** *This bit-field will be reset to 0 after the microprocessor has performed a read to the Data Link Status Register.*
2	Slip Status	RO	0	**Slip Buffer Block Interrupt Status** Indicates if the Slip Buffer block has any outstanding interrupt requests awaiting service. 0: No outstanding interrupts awaiting service 1: Slip Buffer block has an interrupt awaiting service. Interrupt Service routine should branch to and read Slip Buffer Interrupt Status register (address 0xXA,0x09. ***NOTE:*** *This bit-field will be reset to 0 after the microprocessor has performed a read of the Slip Buffer Interrupt Status Register.*
1	Alarm Status	RO	0	**Alarm & Error Block Interrupt Status** Indicates if the Alarm & Error Block has any outstanding interrupts that are awaiting service. 0: No outstanding interrupts awaiting service 1: Alarm & Error Block has an interrupt awaiting service. Interrupt SerStatus Register (address xA,02) ***NOTE:*** *This bit-field will be reset to 0 after the microprocessor has performed a read of the Alarm & Error Interrupt Status register.*
0	E1 Frame Status	RO	0	**E1 Frame Block Interrupt Status** Indicates if an E1 Frame Status interrupt request is awaiting service. 0: No E1 Frame Status interrupt is pending 1: E1 Framer Status interrupt is awaiting service.

TABLE 152: BLOCK INTERRUPT ENABLE REGISTER

REGISTER 137 | **BLOCK INTERRUPT ENABLE REGISTER** | **HEX ADDRESS xA,01**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-6	Unused	RO	0	
5	RxLine Clk Loss	R/W	0	**RxLineClk Loss Interrupt Enable** 0: Disables interrupt 1: Enables interrupt
4	One Sec ENB	R/W	0	**One Second Interrupt Enable** 0: Disables interrupt 1: Enables Interrupt
3	HDLC ENB	R/W	0	**HDLC Block Interrupt Enable** 0: Disables all HDLC Block interrupts 1: Enables HDLC Block (for interrupt generation) at the block level
2	Slip ENB	R/W	0	**Slip Buffer Block Interrupt Enable** 0: Disables all Slip Buffer Block Interrupts 1: Enables Slip Buffer Block at the block level
1	Alarm ENB	R/W	0	**Alarm & Error Block Interrupt Enable** 0: Disables all Alarm & Error Block interrupts 1: Enables Alarm & Error block at the block level
0	E1 Frame ENB	R/W	0	**E1 Frame Block Enable** 0: Disables all E1 Frame Block interrupts 1: enables the E1 Frame Block at the block level

TABLE 153: ALARM & ERROR INTERRUPT STATUS REGISTER

REGISTER 138 **ALARM & ERROR INTERRUPT STATUS REGISTER** **HEX ADDRESS: A,02**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	RxLOF State	RO	0	This Read Only bit field, Receive Loss Of Frame State, reflects a current Loss of Framing condition as detected by the Receive E1 Framer. 0:Receive E1 Framer not declaring Loss of Framing condition 1:Receive E1 Framer declaring Loss of Framing condition
6	RxAIS State	RO	0	**Receive Alarm Indication Status State** This Read Only bit field indicates whether or not the receive E1 Frame is currently detecting an AIS pattern in the incoming E1 data stream. 0: Receive E1 Framer not detecting AIS pattern in incoming E1 data stream 1: Receive E1 Framer detecting AIS pattern in incoming E1 data stream
5	RxMYEL Status	RUR	0	**Receipt of CAS Multiframe Yellow Alarm Interrupt Status.** The Receive E1 Framer will set this bit-field to 1 if it detects the CAS Multiframe Yellow Alarm in the incoming E1 data stream. 0: Receipt of CAS Multiframe Yellow Alarm interrupt has not occurred since the last read of this register. 1: Receipt of CAS Multiframe Yellow Alarm interrupt has occurred since the last read of this register.
4	LOS Status	RUR	0	**Loss of Signal Interrupt Status.** The Receive E1 Framer will set this bit-field to 1 if it detects a consecutive string of 0's at the RxPOX_x and RxNEG_x input pins for 32 bit period. 0: LOS Interrupt has not occurred since the last read of this register 1: LOS Interrupt has occurred since the last read of this register
3	LCV Int Status	RUR	0	**Line Code Violation Interrupt Status**. The Receive LIU Interrupt Block will set this bit-field to 1 if it detects a Line Code Violation in the incoming E1 data stream. 0: Line Code Violation interrupt has not occurred since the last read of this register. 1: Line Code Violation interrupt has occurred since the last read of this register.
2	RxLOF Status	RUR	0	**Change in Receive Loss of Frame Condition Interrupt Status**. The receive E1 Framer block will set this bit-field to 1 if the Receive E1 framer has transition into the In-Frame condition or Loss of Frame condition. 0: Change in RxLOF Interrupt has not occurred since the last read of this register 1: Change in RxLOF Interrupt has occurred since the last read of this register
1	RxAIS Status	RUR	0	**Change in Receive AIS Condition Interrupt Status**. The Receive E1 Framer will generate the Change in AIS Condition interrupt if it starts to detect the AIS pattern in the incoming data stream or if it no longer detects the AIS pattern in the incoming data stream. 0: Change in AIS Condition Interrupt has not occurred since the last read of this register 1: Change in AIS Condition Interrupt has occurred since the last read of this register
0	RxYEL Status	RUR	0	Receipt of FAS Frame Yellow Alarm Interrupt Status. The Receive E1 Framer will generate the FAS Frame Yellow Alarm interrupt if it detects the FAS Frame Yellow Alarm in the incoming E1 data stream. 0: FAS Frame Yellow Alarm interrupt has not occurred 1: FAS Frame Yellow Alarm interrupt has occurred since the last read of this register.

Table 154: Alarm & Error Interrupt Enable Register

Register 139 | **Alarm & Error Interrupt Enable Register** | **Hex Address xA,03**

Bit	Function	Type	Default	Description-Operation
7-6	Unused	RO	0	
5	RxMYEL ENB	R/W	0	0: Disables the Receipt of CAS Multiframe Alarm Interrupt 1: Enables the Receipt of CAS Multiframe Alarm Interrupt
4	LOS ENB	R/W	0	0: Disables Loss of Signal Interrupt 1: Enables Loss of Signal Interrupt
3	LCV ENB	R/W	0	0: Disables Line Code Violation Interrupt 1: Enables Line Code Violation Interrupt
2	RxLOF ENB	R/W	0	0: Disables Change in Receive LOF Condition Interrupt 1: Enables Change in Receive LOF Condition Interrupt
1	RxAIS ENB	R/W	0	0: Disables Change in Receive AIS Condition Interrupt 1: Enables Change in Receive AIS Condition Interrupt
0	RxYEL ENB	R/W	0	0: Disables Receipt of FAS Frame Yellow Alarm Interrupt 1: Enables Receipt of FAS Frame Yellow Alarm Interrupt

TABLE 155: FRAMER INTERRUPT STATUS REGISTER

REGISTER 140	FRAMER INTERRUPT STATUS REGISTER	HEX ADDRESS: XA,04

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	COMFA Status	RUR	0	**Change in CAS Multiframe Alignment Interrupt Status** 0: Change in CAS Multiframe Alignment Interrupt has not occurred since the last read of this register 1: Change in CAS Multiframe Alignment Interrupt has occurred since the last read of this register
6	NBIT Status	RUR	0	**Change in National Bits Interrupt Status** The Receive E1 Framer will generate this interrupt if it has detected a change in the National Bits in the incoming non-FAS E1 Frames. 0: Change in National Bits Interrupt has not occurred since the last read of this register 1: Change in National Bits Interrupt has occurred since the last read of this register.
5	SIG Status	RUR	0	Change in CAS Signaling Interrupt Status The Receive E1 Framer will generate this interrupt if it detects a change in the four-bit signaling values for any one of the 30 voice channels. 0: Change in CAS Signaling Interrupt has not occurred since the last read of this register 1: Change in CAS Signaling Interrupt has occurred since the last read of this register.
4	COFA Status	RUR	0	Change of FAS Frame Alignment Interrupt Status 0: Change in FAS Frame Alignment interrupt has not occurred since the last read of this register 1: Change in FAS Frame Alignment interrupt has occurred since the last read of this register
3	IF Status	RUR	0	Change of In Frame Condition Interrupt Status
2	FMD Status	RUR	0	
1	Sync Error Status	RUR	0	CRC-4 Error Interrupt Status. The Receive E1Framer will declare this interrupt if it detects an error in the CRC-4 bits within a given sub-multiframe. 0: Sync Error has not occurred since the last read of this register 1: Sync Error has occurred since the last read of this register
0	Framing Error Status	RUR	0	0: Framing Bit Error interrupt has not occurred since the last read of this register 1: Framing Bit Error interrupt has occurred since the last read of this register

TABLE 156: FRAMER INTERRUPT ENABLE REGISTER

REGISTER 141 | **FRAMER INTERRUPT ENABLE REGISTER** | **HEX ADDRESS: XA,05**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	COMFA ENB	R/W	0	**Change in CAS Multiframe Alignment Interrupt Enable** 0: Disables the Change in CAS Multiframe Alignment Interrupt 1: Enables the Change in CAS Multiframe Alignment Interrupt
6	NBIT ENB	R/W	0	**Change in National Bits Interrupt Enable** 0: Disables the Change in National Bits Interrupt 1: Enables the Change in National Bits Interrupt
5	SIG ENB	R/W	0	**Change in CAS Signaling Bits Interrupt Enable** 0: Disables the Change in CAS Signaling Bits Interrupt Enable 1: Enables the Change in CAS Signaling Bits Interrupt Enable
4	COFA ENB	R/W	0	**Change in FAS Framing Alignment Interrupt Enable** 0: Disables the Change in FAS Framing Alignment Interrupt Enable 1: Enables the Change in FAS Framing Alignment Interrupt Enable
3	IF ENB	R/w	0	**IF Enable**
2	FMD ENB	R/W	0	**FMD Enable**
1	SE_ENB	R/W	0	**Sync (CRC-4) Error Interrupt Enable** 0: Sync Error Interrupt Disabled 1: Sync Error Interrupt Enabled
0	FE_ENB	R/W0	0	**Framing Bit Error Interrupt Enable** 0: Disables the Framing Bit Error Interrupt 1: Enables the Framing Bit Error Interrupt

TABLE 157: DATA LINK STATUS REGISTER

REGISTER 142	DATA LINK STATUS REGISTER	HEX ADDRESS: XA,06

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	MSG TYPE	RUR	0	**HDLC Message Type Identifier** Indicates type of data link message received by Rx HDLC Controller 0: Bit Oriented Signaling type data link message received 1: Message Oriented Signaling type data link message received
6	TxSOT	RUR	0	**Transmit HDLC Start of Transmission Interrupt Status** Indicates if the Transmit HDLC Start of Transmission Interrupt has occurred since the last read of this register. Transmit HDLC Controller will declare this interrupt when it has started to transmit a data link message. 0: Transmit HDLC Start of Transmission interrupt has not occurred since the last read of this register 1: Transmit HDLC Start of Transmission interrupt has occurred since the last read of this register.
5	RxSOT	RUR	0	**Receive HDLC Start of Reception Interrupt Status** Indicates if the Receive HDLC Start of Reception interrupt has occurred since the last read of this register. Receive HDLC Controller will declare this interrupt when it has started to receive a data link message. 0: Receive HDLC Start of Reception interrupt has not occurred since the last read of this register 1: Receive HDLC Start of Reception interrupt has occurred since the last read of this register
4	TxEOT	RUR	0	**Transmit HDLC End of Transmission Interrupt Status** Indicates if the Transmit HDLC End of Transmission Interrupt has occurred since the last read of this register. Transmit HDLC Controller will declare this interrupt when it has completed its transmission of a data link message. 0: Transmit HDLC End of Transmission interrupt has not occurred since the last read of this register 1: Transmit HDLC End of Transmission interrupt has occurred since the last read of this register
3	RxEOT	RUR	0	**Receive HDLC Controller End of Reception Interrupt Status** Indicates if Receive HDLC End of Reception Interrupt has occurred since the last read of this register. Receive HDLC Controller will declare this interrupt once it has completely received a full data link message. 0: Receive HDLC End of Reception interrupt has not occurred since the last read of this register 1: Receive HDLC End of Reception Interrupt has occurred since the last read of this register
2	FCS Error	RUR	0	**FCS Error Interrupt Status** Indicates if the FCS Error Interrupt has occurred since the last read of this register. Receive HDLC Controller will declare this interrupt if it detects an error in the most recently received data message. 0: FCS Error interrupt has not occurred since last read of this register 1: FCS Error interrupt has occurred since last read of this register
1	Rx ABORT	RUR	0	**Receipt of Abort Sequence Interrupt Status** Indicates if the Receipt of Abort interrupt has occurred since last read of this register. Receive HDLC Controller will declare this interrupt if it detects a string of seven (7) consecutive 1's in the incoming data link channel. 0: Receipt of Abort Sequence interrupt has not occurred since last read of this register 1: Receipt of Abort Sequence interrupt has occurred since last read of this register

TABLE 157: DATA LINK STATUS REGISTER

REGISTER 142 — **DATA LINK STATUS REGISTER** — **HEX ADDRESS: XA,06**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
0	RxIDLE	RUR	0	**Receipt of Idle Sequence Interrupt Status** Indicates if the Receipt of Idle Sequence interrupt has occurred since the last read of this register. The Receive HDLC Controller will declare this interrupt if it detects the flag sequence octet (0x7E) in the incoming data link channel. 0: Receipt of Idle Sequence interrupt has not occurred since last read of this register 1: Receipt of Idle Sequence interrupt has occurred since last read of this register.

TABLE 158: DATA LINK INTERRUPT ENABLE REGISTER

REGISTER 143 — **DATA LINK INTERRUPT ENABLE REGISTER** — **HEX ADDRESS: XA,07**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7	Unused	RO	0	
6	TxSOT ENB	R/W	0	Transmit HDLC Start of Transmission Interrupt Enable 0: Disables the Transmit HDLC Start of Transmission interrupt 1: Enables the Transmit HDLC Start of Transmission interrupt
5	RxSOT ENB	R/W	0	Receive HDLC Start of Reception Interrupt Enable 0: Disables the Receive HDLC Start of Reception interrupt 1: Enables the Receive HDLC Start of Reception interrupt
4	TxEOT ENB	R/W	0	Transmit HDLC End of Transmission Interrupt Enable 0: Disables the Transmit HDLC End of Transmission interrupt 1: Enables the Transmit HDLC End of Transmission interrupt
3	RxEOT ENB	R/W	0	Receive HDLC End of Reception Interrupt Enable 0: Disables the Receive HDLC End of Reception interrupt 1: Enables the Receive HDLC End of Reception interrupt
2	FCS ERR ENB	R/W	0	FCS Error Interrupt Enable 0: Disables FCS Error interrupt 1: Enables FCS Error interrupt
1	RxABORT ENB	R/W	0	Receipt of Abort Sequence Interrupt Enable 0: Disables Receipt of Abort Sequence interrupt 1: Enables Receive of Abort Sequence interrupt
0	RxIDLE ENB	R/W	0	Receipt of Idle Sequence Interrupt Enable 0: Disables Receipt of Idle Sequence interrupt 1: Enables Receipt of Idle Sequence interrupt

TABLE 159: SLIP BUFFER INTERRUPT ENABLE REGISTER

REGISTER 144 — **SLIP BUFFER INTERRUPT ENABLE REGISTER** — **HEX ADDRESS: XA,08**

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-3	Unused	RO	0	
2	FULL ENB	R/W	0	Full Enable
1	EMPTY ENB	R/W	0	Empty Enable
0	SLIP ENB	R/W	0	SLIP Enable

TABLE 160: SLIP BUFFER INTERRUPT STATUS REGISTER

REGISTER 145	SLIP BUFFER INTERRUPT STATUS REGISTER	HEX ADDRESS: XA,09

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-5	Unused	RO	0	
4	CASLOCK	RO	0	
3	CRCLOCK	RO	0	
2	SB_FULL	RUR	0	
1	SB_EMPTY	RUR	0	
0	SB_SLIP	RUR	0	

3.7 THE INTERRUPT STRUCTURE WITHIN THE XRT84V24 QUAD E1 FRAMER IC

The XRT84V24 Quad E1 Framer IC is equipped with a sophisticated Interrupt Servicing Structure. This Interrupt Structure includes an Interrupt Request output pin $\overline{INT}$, numerous Interrupt Enable Registers and numerous Interrupt Status Registers.

The Interrupt Servicing Structure, within the XRT84V24 Quad E1 Framer IC contains three (3) levels of hierarchy:

- The Framer Level
- The Block Level
- The Source Level.

The Framer Interrupt Structure has been carefully designed to allow the user to quickly determine the exact source of this interrupt (with minimal latency) which will aid the µC/µP in determining the which interrupt service routine to call up in order to eliminate or properly respond to the condition(s) causing the interrupt.

The XRT84V24 Quad E1 Framer comes equipped with registers to support the servicing of this wide array of potential "interrupt request" sources. Table 161 lists the possible conditions that can generate interrupts.

TABLE 161: LIST OF THE POSSIBLE CONDITIONS THAT CAN GENERATE INTERRUPTS, IN EACH OF THE FOUR FRAMER

INTERRUPT BLOCK	INTERRUPTING CONDITION
Framer Level	Loss of RxLineClk Signal· One Second Interrupt
HDLC Controller Block	Transmit HDLC - Start of Transmission Receive HDLC - Start of Reception Transmit HDLC - End of Transmission Receive HDLC - End of Reception FCS Error Receipt of Abort Sequence Receipt of Idle Sequence
Slip Buffer Block	Slip Buffer Full Slip Buffer Empty Slip Buffer - Slip
Alarm & Error Block	Receipt of CAS Multi-frame Yellow Alarm Detection of Loss of Signal Condition Detection of Line Code Violation Change in Receive Loss of Framer Condition Change in Receive AIS Condition Receipt of FAS Frame Yellow Alarm
E1 Frame Block	Change in CAS Multi-Frame Alignment Change in National Bits· Change in CAS Signaling Bits Change in FAS Frame Alignment· Change in the "In Frame" Condition Detection of "Frame Mimicking Data" Detection of Sync (CRC-4) Errors Detection of Framing Bit Errors

General Flow of Interrupt Servicing, within the XRT84V24 Quad E1 Framer

When any of the conditions presented in Table 161 occur, (if their Interrupt is enabled), then the Quad E1 Framer IC will generate an interrupt request to the µP/µC by asserting the active-low interrupt request output pin, $\overline{INT}$. Shortly after the local µC/µP has detected the activated $\overline{INT}$ signal, it will enter into the appropriate "user-supplied" interrupt service routine. The first task for the µP/µC, while running this interrupt service routine, may be to isolate the source of the interrupt request down to the device level (e.g, the XRT84V24 Quad E1 Framer), if multiple peripheral devices exist in the user's system. However, once the "interrupting peripheral" device has been identified, the next task for the µP/µC is to determine exactly what feature of functional section within the device requested the interrupt.

Determine the Framer(s) Requesting the Interrupt

If the interrupting device turns out to be the XRT84V24 Quad E1 Framer IC, then the µP/µC must determine which of the four framer channels requested the interrupt. Hence, upon reaching this state, one of the very first things that the µP/µC must do within the user "Quad E1 Framer" interrupt service routine, is to perform a read of each of the "Block Interrupt Status Registers" within all of the Framer channels that have been enabled (for Interrupt Generation) via their respective "Interrupt Control Registers".

Table 162 lists the Indirect Address for the "Block Interrupt Status Registers" associated with each of the Framer channels, within the XRT84V24 Quad E1 Framer.

TABLE 162: ADDRESS OF THE XRT84V24 BLOCK INTERRUPT STATUS REGISTERS

FRAMER NUMBER	ADDRESS OF BLOCK INTERRUPT STATUS REGISTER
0	0x0A, 0x02
1	0x1A, 0x02
2	0x2A, 0x02
3	0x3A, 0x02

The bit-format of each of these "Block Interrupt Status Registers" is listed below.

TABLE 163: BLOCK INTERRUPT STATUS REGISTER

REGISTER 136	BLOCK INTERRUPT STATUS REGISTER	HEX ADDRESS: xA,02

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-6	Unused	RO	0	
5	RxLine Clk Loss	RUR	0	**RxClk Los Interrupt Status** This Reset Upon Read bit field indicates if Framer n has experienced a Loss of Recovered Clock interrupt since last read of this register. 0: Loss of Recovered Clock interrupt has not occurred since last read of this register 1: Loss of Recovered Clock interrupt has occurred since last read of this register.
4	One Sec Status	RUR	0	**One Second Interrupt Status** This Reset Upon Read bit field indicates if the XRT84V24 has experienced a One Second interrupt since the last read of this register. 0: No outstanding One Second interrupts awaiting service 1: Outstanding One Second interrupt awaits service
3	HDLC Status	RO	0	**HDLC Block Interrupt Status** This Read Only bit field indicates if the HDLC block has an interrupt request awaiting service. 0: No outstanding interrupt requests awaiting service 1: HDLC Block has an interrupt request awaiting service. In this case the interrupt Service routine should branch to and read the Data LInk Status Register (address xA,06). *Note: This bit-field will be reset to 0 after the microprocessor has performed a read to the Data Link Status Register.*
2	Slip Status	RO	0	**Slip Buffer Block Interrupt Status** This Read Only bit field indicates if the Slip Buffer block has any outstanding interrupt requests awaiting service. 0: No outstanding interrupts awaiting service 1: The Slip Buffer block has an interrupt awaiting service. In this case the interrupt Service routine should branch to and read Slip Buffer Interrupt Status register (address 0xXA,0x09. *Note: This bit-field will be reset to 0 after the microprocessor has performed a read of the Slip Buffer Interrupt Status Register.*
1	Alarm Status	RO	0	**Alarm & Error Block Interrupt Status** This Read Only bit field indicates if the Alarm & Error Block has any outstanding interrupts that are awaiting service. 0: No outstanding interrupts awaiting service 1: The Alarm & Error Block has an interrupt awaiting service. Interrupt Ser-Status Register (address xA,02) Note: This bit-field will be reset to 0 after the microprocessor has performed a read of the Alarm & Error Interrupt Status register.
0	E1 Frame Status	RO	0	**E1 Frame Block Interrupt Status** This Read Only bit field indicates if an E1 Frame Status interrupt request is awaiting service. 0: No E1 Frame Block interrupt is pending 1: The E1 Framer Block has an interrupt awaiting service.

For a given Framer, the Block Interrupt Status Register presents the "Interrupt Request" status of each "Interrupt Block" within the Framer. The purpose of the "Block Interrupt Status Register" is to help the µP/µC identify which "Interrupt Block(s) have requested the interrupt. Whichever bit(s) are asserted, in this register, identifies which block(s) have experienced an "interrupt generating" condition, as presented in Table 161. Once the µP/µC has read this register, it can determine which "branch" within the interrupt service routine that it must follow; in order to properly service this interrupt.

The Quad E1 Framer IC further supports the "Interrupt Block" Hierarchy by providing the "Block Interrupt Enable Register. The bit-format of this register is identical to that for the "Block Interrupt Status Register", and is presented below for the sake of completeness.

TABLE 164: BLOCK INTERRUPT ENABLE REGISTER

REGISTER 137	BLOCK INTERRUPT ENABLE REGISTER	HEX ADDRESS XA,01

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-6	Unused	RO	0	
5	RxLine Clk Loss	R/W	0	**RxLineClk Loss Interrupt Enable** 0: Disables interrupt 1: Enables interrupt
4	One Sec ENB	R/W	0	**One Second Interrupt Enable** 0: Disables interrupt 1: Enables Interrupt
3	HDLC ENB	R/W	0	**HDLC Block Interrupt Enable** 0: Disables all HDLC Block interrupts 1: Enables HDLC Block (for interrupt generation) at the block level
2	Slip ENB	R/W	0	**Slip Buffer Block Interrupt Enable** 0: Disables all Slip Buffer Block Interrupts 1: Enables Slip Buffer Block at the block level
1	Alarm ENB	R/W	0	**Alarm & Error Block Interrupt Enable** 0: Disables all Alarm & Error Block interrupts 1: Enables Alarm & Error block at the block level
0	E1 Frame ENB	R/W	0	**E1 Frame Block Enable** 0: Disables all E1 Frame Block interrupts 1: enables the E1 Frame Block at the block level

The Block Interrupt Enable Register permits the user to individually enable or disable the interrupt requesting capability of each of the "interrupt blocks" within the Framer. If a particular bit-field, within this register contains the value "0"; then the corresponding functional block has been disabled from generating any interrupt requests.

The procedures for configuring, enabling and servicing interrupts for each of these hierarchical levels is discussed below.

3.7.1 Configuring the Interrupt System, at the Framer Level

The XRT84V24 Quad E1 Framer IC permits the user to enable or disable each of the four Framers for interrupt generation. Further, the chip permits the user to make the following additional configuration selection.

1. Whether the "source-level" Interrupt Status bits are "Reset-upon-Read" or "Write-to-Clear".
2. Whether or not an "activated interrupt" is automatically cleared.

3.7.1.1 Enabling/Disabling the Framer for Interrupt Generation

Each of the four (4) Framers of the XRT84V24 Quad E1 Framer can be enabled or disabled for interrupt generation. This selection is made by writing the appropriate “0” or “1” to bit 0 (INTRUP_EN) of the "Interrupt Control Register" corresponding to that framer, (see Table 165.)

TABLE 165: INTERRUPT CONTROL REGISTER

REGISTER 26	INTERRUPT CONTROL REGISTER	HEX ADDRESS: x00,1A

BIT	FUNCTION	TYPE	DEFAULT	DESCRIPTION-OPERATION
7-3	Unused	RO	0	
2	INT_WC_RUR	R/W	0	**Interrupt Write-to-Clear or Reset-upon-Read Select** Configures Interrupt Status bits to either RUR or Write-to-Clear 0: Interrupt Status bit RUR 1: Interrupt Status bit Write-to-Clear
1	ENBCLR	R/W	0	**Interrupt Enable Auto Clear** 0: Interrupt Enable bits are not cleared after status reading 1: Interrupt Enable bits are cleared after status reading
0	INTRUP_ENB	R/W	0	**Interrupt Enable for Framer_n** Enables Framer n within the XRT84V24 for Interrupt Generation. 0: Disables corresponding E1 framer block for Interrupt Generation 1: Enables corresponding E1 framer block for Interrupt Generation

Setting this bit-field to "0" disables all interrupts within the Framer. Setting this bit-field to "1" enables the Framer for interrupt generation (at the Framer Level).

NOTE: *It is important to note that setting this bit-field to "1" does not enable all of the interrupts within the Framer. A given interrupt must also be enabled at the block and source-level, before it is enabled for interrupt generation.*

3.7.1.2 Configuring the "Interrupt Status Bits", within a given Framer to be "Reset-upon-Read" or "Write-to-Clear".

The XRT84V24 Source-Level Interrupt Status Register bits can be configured to be either "Reset-upon-Read" or "Write-to-Clear". If the user configures the Interrupt Status Registers to be "Reset-upon-Read", then when the µP/µC is reading the interrupt status register, the following will happen.

1. The contents of the Source-Level Interrupt Status Register will automatically be reset to "0x00", following the read operation.
2. The Interrupt Request Output pin ($\overline{INT}$) will automatically toggle false (or "high") upon reading the Interrupt Status Register containing the last activated interrupt status bit.

If the user configures the Interrupt Status Registers to be "Write-to-Clear", then when the µP/µC is reading the interrupt status register, the following will happen.

1. The contents of the Source-Level Interrupt Status Register will not be cleared to "0x00", following the read operation. The µP/µC will have to write 0x00 to the interrupt status register in order to reset the contents of the register to 0x00.
2. Reading the Interrupt Status Register, which contains the activated bit(s) will not cause the "Interrupt Request Output" pin ($\overline{INT}$) to toggle false. The Interrupt Request Output pin will not toggle false until the µP/µC has written 0x00 into this register. (Hence, the Interrupt Service Routine must include this write operation).

The Interrupt Status Register (associated with a given framer) can be configured to be either "Reset-upon-Read" or "Write-to-Clear" by writing the appropriate value into Bit 2, within the Interrupt Control Register as indicated in Table 165.

Writing a "0" into this bit-field configures the Interrupt Status registers to be "Reset-upon-Read". Conversely, writing a "1" into this bit-field configures the Interrupt Status registers to be "Write-to-Clear".

3.7.1.3 Automatic Reset of Interrupt Enable Bits

Occasionally, the user's system (which includes the Framer device), may experience a fault condition, such that a "Framer Interrupt Condition" will continuously exist. If this particular interrupt has been enabled (within the Framer), then the Framer device will generate an interrupt request to the µP/µC. Afterwards, the µP/µC will attempt to service this interrupt by reading the appropriate Block-level and Source-Level Interrupt Status Register. Additionally, the local µP/µC will attempt to perform some "system-related" tasks in order to try to resolve these conditions causing the interrupt. After the local µC/µP has attempted all of these things, the Framer IC will negate the $\overline{INT}$ output pin. However, because this system fault still remains, the condition causing the Framer to issue this interrupt also exists. Consequently, the Framer device will generate another interrupt request, which forces the µP/µC to "once again" attempt to service this interrupt. This phenomenon quickly results in the local µP/µC being "tied up" in a continuous cycle of executing this one interrupt service routine. Consequently, the µP/µC (along with portions of the overall system) now becomes non-functional.

In order to prevent this phenomenon from ever occurring, the Framer IC can be configured to automatically reset the "interrupt enable" bits, following their activation. This feature can be implemented by writing the appropriate value to bit 1 of the "Interrupt Control Register" as indicated in Table 165.

Writing a "1" to this bit-field configures the Framer IC to reset a given interrupt following activation. Writing a "0" to this bit-field configures the Framer IC to leave the interrupt enabled, following its activation.

4.0 CLOCK DISTRIBUTION SYSTEM

The purpose of the "Clock Distribution" system is to generate and supply the necessary clock signals to circuitry within the Quad E1 Framer, in order to permit the chip to function. In all, the XRT84V24 generates all of its internal timing from the following three input signals.

- RxLineClk_n input (provides timing for framer n)
- TxSerClk_n input (provides timing for framer n)
- OSCClk input (provides timing for the entire chip).

The Transmit and Receive sections, within each framer, have different Clock Distribution options which are discussed below.

Receive Section

The Receive Section (within a given framer) relies upon both the 16MHz clock signal (derived from the OSCClk clock input signal) and its corresponding Recovered Line Clock (RxLineClk_n) as its sources for timing

Transmit Section

In addition to the 16MHz clock signal (derived from the OSCClk input signal), the Transmit Section (within a given framer) can be configured to use any of the following signals for its timing reference.

- RxLineClk_n input
- TxSerClk_n input
- 2.048 MHz clock signal (divided down from the OSCClk input signal).

A programmable clock input OSCClk is provided to allow the generation (division) of a clock rate of 32.768MHz in the system for digital signal operations. A 2.048MHz clock signal is also derived from this clock from transmit framing operations where both the transmit and receive clocks TxSerClk_n, RxClk are not selected by setting the CSS[1:0] bits of the Clock Select Register (CSR) to "2". Setting CSS[1:0] = 1 will make the E1 Transmitter use the TxSerClk as the transmit clock and setting CSS[1:0] = 0 will force the transmitter to use RxClk as the transmit clock.

The OSCClk input can be connected to either a 65.536MHz, 32.768MHz or 16.384MHz crystal oscillator, and the Clock Frequency Select bits, CFS[1:0] of the Clock Select Register are programmed accordingly. This signal drives a divider that produces an internal signal at the rate of 16.384MHz to be used for framing operations. The table below lists the data bits in the Clock Select Register. In the case the received clock is not recoverable from the LIU device, if CSS[1:0], the E1 framer will detect a loss of clock condition and automatically switch to the clock generated by the OSCClk input signal, provided the CLDET (Clock Loss Detection) bit is set to "high". In the case that CSS[1:0] = 1, the transmit clock is always generated from the TxSerClk clock. However, LOS may be detected if no signal transition occurs for 5ms.

Clock Select Register (CSR)—Address = x0h, 00h (Indirect)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Unused	Reserved	8KHz	CLDET	CFS[1:0]		CSS[1:0]	
RO	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

Bit 0, 1— Clock Source Select (Transmitter)

CSS[1:0] = 00—Recovered Receive Channel Clock, RxClk

The Recovered Receive Channel Clock, RxClk (from the LIU), will be used as the Transmit Clock. The TxSerClk output clock signal will be derived from RxClk.

CSS[1:0] = 01—Transmit Serial Input Clock, TxSerClk

The TxSerClk signal will be configured to be an input signal. Additionally, it will function as the source of timing for the Transmitter.

CSS[1:0] = 10—OSCClk Driven Divided Clock

In this mode, the OSCClk Driven Divided clock signal will function as the source of timing for the Transmitter. The TxSerClk output clock signal will be derived from the OSCClk Driven Divided clock signal.

Bit 2, 3—Clock Frequency Select

CFS[1:0]—Clock Frequency Select.

The following table relates the contents of these bit-fields to the frequency of the signal applied to the OSCClk pin.

TABLE 166: BIT FIELD CONTENTS FOR SIGNAL FREQUENCY APPLIED TO OSCCLK

CFS[1:0]		OSCCLK FREQUENCY	INTERNALLY DIVIDED
0	0	16.384MHz	divided by 1
0	1	32.768MHz	divided by 2
1	0	65.536MHz	divided by 4
1	1	Reserved	

Bit 4—CLDET (Clock Loss Detect)

This Read/Write bit allows the user to configure some additional protection for the Framer, whenever the Recovered Received Line Clock (RxClk) is used as the timing source for the transmit portion of the framer. If this protection feature is employed and if the Recovered Received Line Clock is used as the timing source, and if the LIU somehow loses clock recovery, then the Clock Distribution Block will detect this occurrence and automatically begin to use the OSCClk Driven Divided clock as the Transmitter source, until the LIU is able to regain clock recovery.

Writing a "1" to this bit-field enables this protection. Writing a "0" to this bit-field disables this protection.

***NOTE:** This bit-field is ignored if the TxSerClk or the OSCClk Driven Divided clocks are chosen to be the timing source for the Transmit section of the Framer.*

Bit 5—8KHz (Synchronization between OSCClk and 8KHzREF)

This Read/Write bit-field allows the user to configure the Quad E1 Framer IC to force synchronization between the OSCClk input and the 8KHZREF input signal.

***NOTE:** Setting this configuration in any one framer, forces this configuration for the entire chip.*

A "1" forces this synchronization between the OSCClk input and the 8KHZREF input. A "0" does not force this synchronization.

5.0 TRANSMIT TERMINAL SERIAL INPUT INTERFACE

Each of the four framers, within the XRT84V24 device includes a "Transmit Terminal Serial Input Interface" block. The purpose of this block is to provide an interface to the terminal equipment (e.g., Central Office or switching equipment) that has data to send to a "Far End" terminal over an E1 transport medium.

The Transmit Terminal Serial Input Interface block supplies the following signals to the "local" Terminal equipment circuitry.

- TxSer_n
- TxSerClk_n
- TxSync_n
- TxMSync_n
- TxTSClk_n
- TxTSb[4:0]_n

Figure 31 presents a simple illustration of the "Transmit Terminal Serial Input Interface" block.

FIGURE 31. BLOCK DIAGRAM OF TRANSMIT TERMINAL SERIAL INPUT INTERFACE

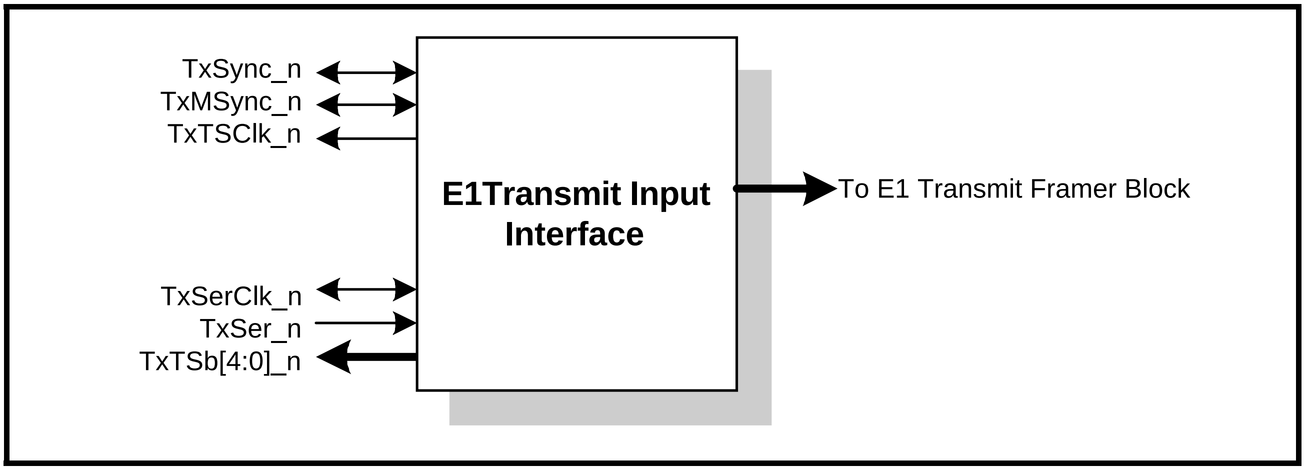


Table 167 presents a detailed description of each of the pins associated with the "Transmit E1 Serial Input Interface".

TABLE 167: PIN DESCRIPTION OF THE "TRANSMIT E1 SERIAL INPUT INTERFACE" BLOCK

SIGNAL NAME	PIN NUMBER	TYPE	DESCRIPTION
TxSer_n	143 127 30 14	I	**Transmit Serial Input—Framer_n:** This input pin, along with TxSerClk_n function as the Transmit Serial input port, for Framer_n. Any data that is applied to this pin, will be inserted into an E1 frame, and ultimately output onto the E1 line, via the TxPOS_n and TxNEG_n output pins. All data that is intended to be transported via Time Slots 1 through 15 and Time slots 17 through 31, within each E1 frame must be applied to this input. If Framer_n is configured accordingly, Time Slots 0 and 16 can also be applied to input pin as well. The signal, applied to this input pin, is latched into the E1 Transmit Payload Data Input Interface on the rising edge of TxSerClk_n.
TxSerClk_n	145 126 31 12	I/O	**Transmit Serial Input—Clock Signal for Framer_n:** This signal is used by the E1 Transmit Input Interface, to latch the contents of the TxSer_n into the Quad E1 Framer IC. Data that is applied at the TxSer_n input is latched into the E1 Transmit Input Interface (for Framer_n) on the rising edge of TxSerClk_n. TxSerClk_n can either be an input or an output. If the Transmit Section of Framer_n has been configured to use the TxSerClk_n signal as the timing source, then this signal will be an **Input**. If the Transmit Section of Framer_n has been configured to use either the RxLineClk signal or the OSCClk signal as the timing source, then TxSerClk_n will be an **Output**.

TABLE 167: PIN DESCRIPTION OF THE "TRANSMIT E1 SERIAL INPUT INTERFACE" BLOCK

SIGNAL NAME	PIN NUMBER	TYPE	DESCRIPTION
TxTSb[4:0]_n	168, 166, 163, 161, 159 106, 109, 111, 114, 116 51, 48, 46, 43, 41 196, 198, 201, 203, 205	O	**Transmit Framer_n-Time Slot Octet Identifier Output-Bit [0:4]:** These output signals (TxTSb4_n through TxTSb0_n) reflects the five-bit binary value of the Time Slot number (in the incoming E1 frame), being accepted and processed by the Transmit Payload Data Input Interface block associated with Framer_n. Terminal Equipment should use the TxChClk_n clock signal to sample the five output pins of each channel in order to identify the time-slot being processed by the Transmit Payload Data Input Interface block of Framer_n.
TxSync_n	148 125 32 9	I or O	**E1 Single Frame Sync Pulse Input/Output—Transmit Framer_0:** This pin is configured to be an **input** if the TxSerClk_0 input pin is configured to be the timing reference for the Transmit Portion of Framer_n. This pin will be configured as an **output** if the RxLineClk input pin or the OSCClk input pins are configured to be the timing reference for the Transmit portion of Framer_n. **Input:** If this pin is configured to be an input, then the user must pulse this pin “High” for one period of TxSerClk_n, when the Transmit Input Interface (of Framer_n) is receiving the International Bit (Si) of an outbound E1 frame. **Output:** If this pin is configured to be an output, then it will pulse “High”, for one period of TxSerClk_n, when the Transmit Input Interface (of Framer_n) is receiving the last bit within a given outbound E1 frame.
TxMSync_n	150 124 33 7	I/O	**E1 Multiframe Sync Pulse Input/Output—Framer_n:** This pin is configured to be an input if the TxSerClk_n input pin is configured to be the timing reference for the Transmit portion of Framer_n. Conversely, this pin will be configured as an output if the RxLineClk input pin or the OSCClk input pins are configured to be the timing reference for the Transmit portion of Framer_n. **Input:** If this pin is configured to be an input, this pin must be pulsed “High” for one period of TxSerClk_n, when the Transmit Input Interface (of Framer_n) is receiving the first International Bit (Si) of an “outbound” CRC Multiframe. ***NOTE:*** *This input pin is ignored if CRC Multiframe Alignment has been disabled.* **Output:** If this pin is configured to be an output, then it will pulse “High”, for one period of TxSerClk_n, when the Transmit Input Interface (of Framer_n) is receiving the last bit, within an “outbound” CRC Multi-frame. ***NOTES:*** *1. This pin is inactive if CRC Multiframe Alignment has been disabled.* *2. The purpose of this output pin is to permit the Terminal Equipment to maintain alignment with the “outbound” CRC-Multi-frame structure.*

TABLE 167: PIN DESCRIPTION OF THE "TRANSMIT E1 SERIAL INPUT INTERFACE" BLOCK

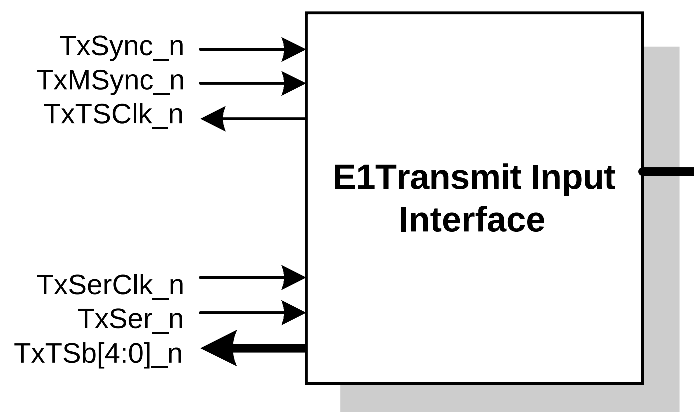
SIGNAL NAME	PIN NUMBER	TYPE	DESCRIPTION
TxTSClk_n	155 120 37 2	O	**Transmit Channel Clock Output Signal—Framer_n:** 256kHz clock output which pulses "high" whenever the Transmit Payload Data Input Interface block accepts the LSB of each of the 32 time slots, within the E1 data stream, being processed via Framer _n. The Terminal Equipment should use this clock signal to sample the TxTSb0_n through TxTSb4_n output signals, and identify the time-slot being processed via the "Transmit Section" of each Framer_n.

The function/role of these output pins depends on whether or not the "Transmit" Section of the framer is configured to use the TxSerClk_n signal as the timing reference. The operation and requirements of the Transmit Terminal Serial Input Interface is discussed below; for each of these two modes.

5.1 TRANSMIT TIMING REFERENCE = TXSERCLK_N

When the TxSerClk_n input signal is configured to be the timing source for the Transmit Section of the framer, the clock signal pin (TxSerClk_n) is configured to be an **input** (2.048Mbps free-running clock). The framing reference signals (TxSync_n and TxMSync_n) are also automatically configured to be **input** signals and they should be pulsed high at the beginning of each frame/multiframe. It is the responsibility of the "local" Terminal Equipment to provide the serial input data (TxSer_n) aligned with the transmit framing signals (TxSync_n and TxMSync_n). See Figure 32.

FIGURE 32. BLOCK DIAGRAM OF THE TRANSMIT TERMINAL SERIAL INPUT INTERFACE - WHEN THE TXSERCLK SIGNAL IS SELECTED AS THE TIMING REFERENCE



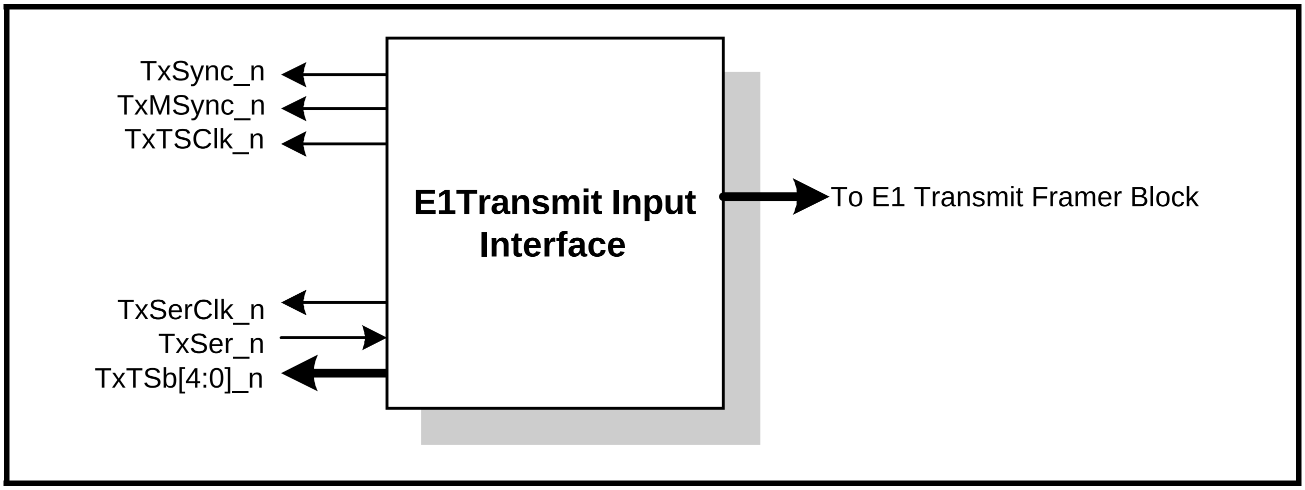
5.2 TRANSMIT TIMING REFERENCE = RXLINECLK OR OSCCLK

OPERATING THE E1 TRANSMIT INPUT INTERFACE - USING RECOVERED LINE CLOCK OR THE 2.048 MHZ SIGNAL

When either the Recovered Line Clock or the OSCClk inputs are configured to be the timing source for the Transmit Section of a given framer (selected via the Clock Source Select bits in the Clock Control Register are set to 00 or 10), the clock signal (TxSerClk_n) is an **output** clock signal and the framing reference signals serve as synchronization outputs which they pulse high at the end of each frame/multiframe.

Figure 33 presents a simple illustration of the Transmit E1 Input Interface block, when the Transmit Section of the framer has been configured to use either the Recovered Line Clock or the 2.048 MHz (derived from the OSCClk input signal) as the Timing Source.

FIGURE 33. BLOCK DIAGRAM OF THE E1 TRANSMIT INPUT INTERFACE - USING EITHER THE RECOVERED LINE CLOCK OR THE 2.048 MHZ OSCCLK INPUT AS THE TIMING SOURCE



5.3 TRANSMIT TERMINAL SERIAL INPUT INTERFACE OPERATION

The purpose of the Transmit Terminal Serial Input Interface is to accept data from the local Terminal Equipment and provide this data to the Transmit E1 Framer block. At a minimum, the Transmit Terminal Serial Input Interface will be configured to accept "user data" from the Terminal Equipment. In addition to accepting the "user data", the Transmit Terminal Serial Input Interface can be configured as follows:

- To accept data link information from the Terminal Equipment.
- To accept the CRC-4 bits (for a given Sub Multiframe) from the Terminal Equipment.
- To accept the FAS (Framing Alignment Signaling) bits, for each FAS frame, from the Terminal Equipment.

The procedure to configure the Transmit Terminal Serial Input Interface into each of these modes, as well as how it operates is discussed below.

5.3.1 Transmit Terminal Serial Input Interface Operation when, it has been configured to accept data intended for Timeslots 1 through 15 and 17 through 31.

The Transmit Terminal Serial Input Interface will be configured to only accept data for Timeslots 1 through 15 and 17 through 31, if all of the following conditions are true.

1. The Data Link Source Select bits (DLSRC[1:0]) within the Synchronization MUX Register (Address = x00h, 09h) are set to "01" or "10".

NOTES:

1. *Setting DLSRC[1:0] = "01" configures the Transmit Section of the Framer to user the Transmit HDLC Controller as the source of the Data Link bits.*
2. *Setting DLSRC[1:0] = "10" configures the Transmit Section of the Framer to use the "Transmit Overhead Input Port" as the source of the Data Link bits.*

2. The CRCSRC bit, within the Synchronization MUX Register (Address = x00h, 09h) is set to "0". This will configure the "Transmit Section" of the Framer to internally compute and insert the CRC-4 bits into the "outbound" CRC Multiframes.

3. The FSRC bit, within the Synchronization MUX Register, is set to "0". This will configure the Transmit Section of the Framer to internally generate and insert the FAS pattern into the "outbound" FAS frames.

4. The SIGDL[2:0] bits, within the "Transmit Signaling and Data Link Select" register (Address = x00h, 0Ah) are set to "1xx". This setting configures the Transmit Section of the Framer to use Common Channel Signaling. As a result, the Transmit HDLC Controller will be the source of the "Timeslot 16" data.

If the Transmit Terminal Serial Input Interface is configured to operate in this mode, then data (which is intended to be transported via Time Slots 1 through 15 or Time Slots 17 through 31) will be latched into the "Transmit Terminal Serial Input Interface" on the rising edge of the corresponding TxSerClk_n signal. For a given E1 frame, the Transmit Terminal Serial Input Interface will be responsive to 240 out of 256 rising edges of TxSerClk. However, for the remaining 16 rising edges of TxSerClk_n, the Transmit Serial Input Interface will simply ignore the data on the "TxSer" line.

5.3.2 Operation of the Transmit Terminal Serial Input Interface when it has been configured to be the source of Data Link Information.

As mentioned earlier, the Transmit Terminal Serial Input Interface will always accept data (from the Terminal Equipment) that is intended to be transported via the "User" Timeslots (e.g., Timeslots 1 through 15 and Timeslots 17 through 16). However, the Transmit Terminal Serial Input Interface can also be configured to be the source of data link information (to the Transmit E1 Framer block). More specifically, the Transmit Terminal Serial Input Interface block can be configured to accept data (from the Terminal Equipment) that is intended to be transported via the National Bits (e.g., Sa4 through Sa8).

The Transmit Terminal Serial Input Interface can be configured to accept Data Link information (via the TxSer_n input pin) by executing the following steps.

Step 1: The Data Link Source (DLSRC[1:0]) bits, within the Synchronization MUX Register, must be set to "00" or "11" as illustrated below.

Synchronization MUX Register (Address = x00h, 09h)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Not Used			SyncInv	DLSRC[1:0]	CRCSRC	FSRC	Not Used
RO	RO	RO	R/W	R/W	R/W	R/W	R/W
			X	01	01	X	X

Step 2: Specify which National Bits will be transporting Data Link Information

The XRT84V24 IC permits the user to designate any combination of the National Bits to be used to carry data link information. The user can specify which National Bits are to be used by writing the appropriate data into the Transmit Signaling and Data Link Select Register; as illustrated below

5.3.3 Operation of the Transmit Terminal Serial Input Interface when it has been configured to be the source of the CRC-4 bits.

5.3.4 Operation of the Transmit Terminal Serial Input Interface when it has been configured to be the source of the FAS (Framing Alignment Signaling) bits.

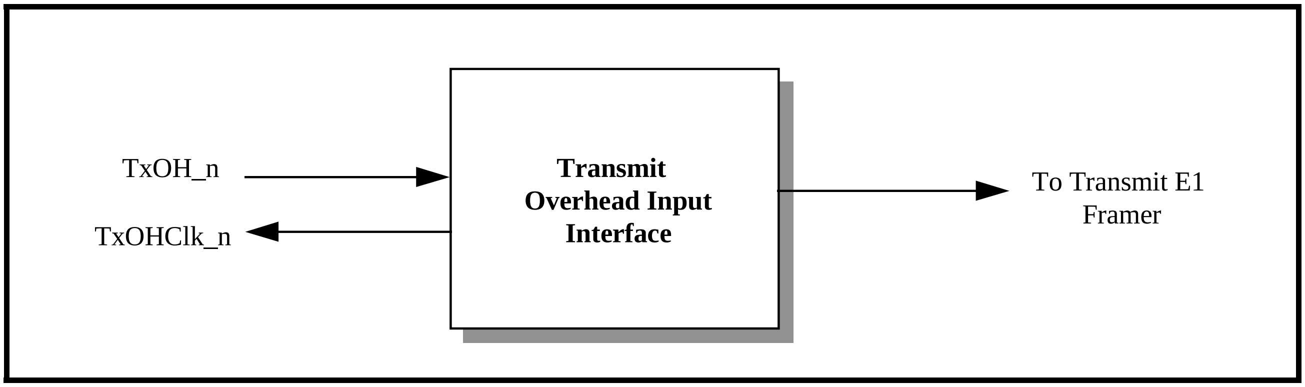
6.0 TRANSMIT OVERHEAD INPUT INTERFACE

The Transmit Overhead Input Interface, for a given E1 Framer, consists of two signals.

- TxOH_n: The Transmit Overhead Input Interface Input pin
- TxOHClk_n: The Transmit Overhead Input Interface - Clock Output signal.

Figure 34 presents a simple illustration of the "Transmit Overhead Input Interface" block

FIGURE 34. BLOCK DIAGRAM OF THE E1 TRANSMIT OVERHEAD INTERFACE BLOCK



The purpose of the Transmit E1 Overhead Input Interface is to permit "Data Link" equipment direct access to the "Sa4" through "Sa8" bits that are to be transported via the "outbound" E1 frames. The Transmit Overhead Input Interface will be active only if the DLSRC[1:0] bits, within the Synchronization MUX Register (Address = x00h, 09h) has been set to "10", as illustrated below.

Synchronization MUX Register (Address = x00h, 09h)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Not Used			Sync Inv	DLSRC[1:0]	CRCSource	FAS Source	Not Used
RO	RO	RO	R/W	R/W	R/W	R/W	R/W
X	X	X	X	1	0	X	X

When this setting is made, the Transmit E1 Framer block will be using the Transmit Overhead Input Interface as the source for the Data Link bits. If the DLSRC[1:0] bits are set to any value other than "10", then the Transmit Overhead Input port will be disabled, and the TxOHClk_n output clock signal will become inactive.

How the Transmit Overhead Input Interface Works

If the "Data Link Source" (e.g., DLSRC[1:0]) bits within the Synchronization MUX Register are set to "10", then the Transmit Overhead Input Interface will become active. From this point on, the exact behavior of the Transmit Overhead Input Interface depends upon the following.

1. How many of the National Bits will be used to carry the Data Link bits, and
2. Which of these National Bits will be used to carry the Data Link bits.

Depending upon the user's selection, via the "Transmit Signaling and Data Link Select" Register (Address = x00h, 0Ah) either of the following cases may exists:

1. None of the "National Bits" are used to transport the Data Link bits (e.g., TxOH is inactive).
2. Any combination of between 1 and all 5 of the National bits can be selected to transport Data Link bits.

For every "Sa" bit that is selected to carry Data Link information, the Transmit Overhead Input Interface will do the following.

The Transmit Overhead Input Interface block will supply a clock pulse, via the TxOHClk_n output pin, such that:

1. The Data Link equipment, interfaced to the Transmit Overhead Input Interface should update the data on the TxOH line, upon detection of the rising edge of TxOHClk_n.

2. The "Transmit Overhead Input Interface" will sample and latch the data, on the TxOH line, on the falling edge of TxOHClk_n.

7.0 THE TRANSMIT E1 FRAMER BLOCK

The purpose of the Transmit E1 Framer block is to embed and encode user timeslot data into E1 frames; and to route this E1 frame data to the Transmit E1 LIU Interface block. This section presents a detailed functional block diagram of the Transmit E1 Framer block. Please note that the Quad E1 Framer actually consists of four (4) of these Transmit E1 Framer blocks. Hence, the following description applies to all four of these individual Transmit E1 Framer blocks.

The purpose of the Transmit E1 Framer block is:

- To encode user data, from the "Terminal Equipment Side" into a standard E1 framing format.
- To support the transmission of HDLC messages, from the Local Transmitting terminal, to the Remote Receiving terminal.
- To transmit indications that the Local Receive E1 Framer has received errored E1 frames from the Remote terminal.
- To transmit "alarm condition" indicators to the Remote Terminal.

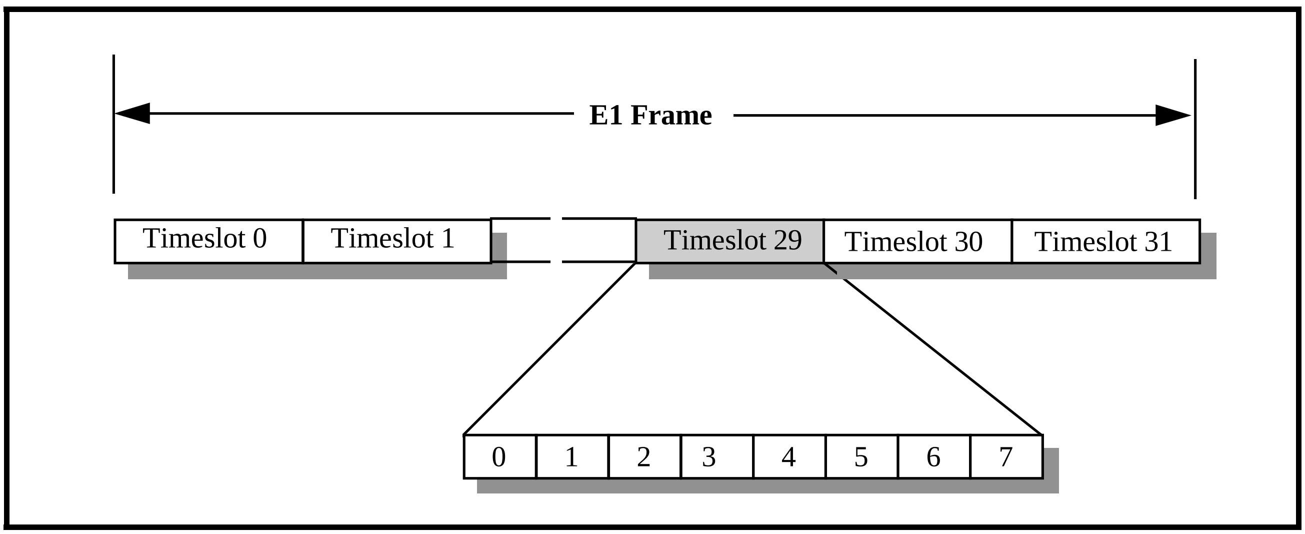
7.1 THE E1 FRAME STRUCTURE

In order to adequately discuss the functionality of the Transmit E1 Framer, one should begin by describing the E1 framing structure.

The Single E1 Frame

A single E1 frame consists of 256 bits, which is created 8000 times a second; thereby yielding a bit-rate of 2.048Mbps. The 256 bits, within each E1 frame, is grouped into 32 octets (or timeslots). These timeslots are numbered from 0 to 31. Figure 35 presents a simple diagram of a single E1 frame.

FIGURE 35. E1 FRAME DIAGRAM



A single E1 frame consists of 32 timeslots. However, not all of these timeslots are available to transmit voice or user data. For instance, timeslot 0 is always reserved for system use; and timeslot 16 is sometimes used (or reserved) by the system. Hence, within each E1 frame, typically either 30 or 31 of the 32 timeslot are available for transporting user (or voice) data. The exact role of timeslot 0 and timeslot 16 will be described in greater detail in the next few sections.

Timeslot 0

In general, there are two types of E1 frames.

- FAS (Frame Alignment Signaling) frames; and
- Non-FAS frames.

In any E1 data stream, the E1 frame type will alternate between the "FAS" and "non-FAS" frames.

The exact role of the "timeslot 0" octet depends upon which type of E1 frame it is residing in. In general, the "timeslot 0" octet within the "FAS" E1 frame contain a framing alignment pattern and therefore supports framing. The "timeslot 0" octet within the "non-FAS" E1 frame contains bits that support signaling or data link message transmission.

Timeslot 0 octets within FAS frames

The bit-format of a "timeslot 0" octet, within a FAS frame, is presented in Figure 36.

FIGURE 36. BIT FORMAT OF TIMESLOT 0 OCTET WITHIN A "FAS" E1 FRAME

BIT 0	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7
Si	0	0	1	1	0	1	1
International Bit	FAS Pattern						

Figure 36 indicates that the "FAS" frame "timeslot 0" octet consists of a single "International Bit", (within bit-field 0) Si, followed by a fixed 7-bit pattern (within bit-fields 1 through 7)

Bit 0 - Si (International Bit)

In practice, the Si bit, within the "FAS" E1 Frame carries the results of a CRC-4 calculation, which is discussed in greater detail within the section discussing the CRC Multiframe Structure. The "fixed" framing pattern (e.g., 0, 0, 1, 1, 0, 1, 1) will be used by the Receive E1 Framer, at the Remote terminal for frame synchronization/alignment purposes. Section _ discusses how the Receive E1 Framer uses these bits.

Timeslot 0 octets within "non-FAS" frames

The bit-format of a "timeslot 0" octet within a "non-FAS" frame is presented in Figure 37.

FIGURE 37. BIT FORMAT OF THE "TIMESLOT 0" OCTET WITHIN A "NON-FAS" E1 FRAME

BIT 0	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7
Si	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
International Bit	Fixed Value	Yellow Alarm	National Bits				

Figure 37 indicates the "non-FAS" frame "timeslot 0" octet consists of a single international bit, Si, within bit-field 0.

Bit 0 - Si (International Bit)

In practice, the Si bit, within the "non-FAS" E1 Frame carries a specific value that will be used by the Receive E1 Framer, for CRC Multi-frame alignment purposes. Section _ discusses the exact role of the Si bit-field within the "non-FAS" frames.

Bit 1 - Fixed at "1"

Bit-field "1" contains a fixed value "1". This bit-field will be used for "FAS framing synchronization/alignment purposes by the Remote Receive E1 Framer. Section _ discusses how the Receive E1 Framer uses this bit-field.

Bit 2 - A (FAS Frame Yellow Alarm Bit)

This bit-field is used to transmit a "Yellow" alarm to the Remote Terminal. This bit-field is set to "0" during normal conditions, and is set to "1" whenever the Receive E1 Framer detects an LOS (Loss of Signal) or LOF (Loss of Framing) condition in the incoming E1 frame data.

The user can specify the criteria for which a "FAS" Yellow Alarm is generated by writing the appropriate value to the "YEL[1:0]" bit-fields within the Alarm Generation Register, as depicted below.

Alarm Generation Register (Address = x0h, 08h)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Unused	LOF	YEL[1:0]		AISG[1:0]		AISD[1:0]	
RO	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

The relationship between the contents with YEL[1:0] and the "Yellow Alarm" generation criteria is tabulated below.

YEL[1:0]	HANDLING OF YELLOW ALARMS AND MULTIFRAME YELLOW ALARMS
00	Disables the Transmission of Yellow Alarm
01	**Enables the Automatic Transmission of Yellow Alarms** 1. The Yellow Alarm bits (Bit 2 of the non-FAS frames in TS0) is transmitted by echoing the receive FAS alignment status. A logic "1" is transmitted if a loss of FAS alignment occurs. 2. The Multiframe yellow alarm bits (bit 6 of frame 0 in TS16) is transmitted by echoing the receive CAS multiframe alignment status. Logic "1" is transmitted is a loss of CAS multi-frame occurs.
10	Yellow and Multiframe Yellow Alarm bit-fields are transmitted as "0".
11	Yellow and Multiframe Yellow Alarm bit-fields are transmitted as "1".

Bits 3 through 7 - Sa4 - Sa8 (National Bits)

These bit-fields can be used to carry data link information from the Local transmitting terminal to the Remote receiving terminal. Since the National bits only exist in the "non-FAS" frames, they offer a maximum signaling data link bandwidth of 20kbps.

7.1.1 The E1 Multi-frame Structures

The XRT84V24 Quad E1 Framer supports two kinds of E1 Multi-frames:

- CRC Multi-frame
- CAS Multi-frame

THE CRC MULTI-FRAME STRUCTURE

A CRC Multi-frame consists of 16 consecutive E1 frames, with the first of these frames being a "FAS" frame. From a "frame alignment" point of view, the "timeslot 0" octets of each of these E1 frames (within the Multi-frame) are the most important 16 octets within the Multi-frame. Figure 38 presents the bit-format for all "timeslot 0" octets within a 16 frame CRC Multi-frame.

FIGURE 38. BIT FORMAT OF ALL "TIMESLOT 0" OCTETS WITHIN A CRC MULTI-FRAME

SMF	FRAME NUMBER	BIT 0	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7
1	0	C1	0	0	1	1	0	1	1
	1	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	2	C2	0	0	1	1	0	1	1
	3	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	4	C3	0	0	1	1	0	1	1
	5	1	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	6	C4	0	0	1	1	0	1	1
	7	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
2	8	C1	0	0	1	1	0	1	1
	9	1	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	10	C2	0	0	1	1	0	1	1
	11	1	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	12	C3	0	0	1	1	0	1	1
	13	E	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	14	C4	0	0	1	1	0	1	1
	15	E	1	A	Sa4	Sa5	Sa6	Sa7	Sa8

Figure 38 indicates that the CRC Multi-frame can be divided into 2 sub multiframes. Sub-multi-frame 1 is designated as "SMF1" and Sub-multi-frame 2 is designated as "SMF2".

SMF1 consists of E1 frames 0 through 7. Hence, SMF1 consists of 4 "FAS" frames and 4 "non-FAS" frames.

There are two points to note, in Figure 38. First, all of the bit-field 0 positions, within each of the "FAS" frames are designated as "C1", "C2", "C3" and "C4". These four bit-fields contain the CRC-4 values which has been computed over the previous SMF. Hence, while the Transmit E1 Framer is assembling a given SMF, it will compute the CRC-4 value for that SMF and will insert these results into the C1 through C4 bit-fields within the very next SMF. These CRC-4 values will ultimately be used by the Remote Receive E3 Framer, for error-detection purposes.

***NOTE:** This framing structure is referred to as a "CRC Multi-frame" because it permits the remote receiving terminal to locate (and in turn, verify) the CRC-4 bit-fields.*

The second point to note regarding Figure 38 is that the bit-field 0 positions, within each of the "non-FAS" frames are of a fixed six (6) bit pattern: 0, 0, 1, 0, 1, 1; along with two bits, each designated at "E". This six bit pattern is referred to as the "CRC Multi-Frame" alignment pattern. This six-bit pattern will ultimately be used by the Remote Receive E1 Framer for "CRC Multi-Frame" synchronization/alignment. Section _ presents a detailed discussion on how the Receive E1 Framer uses this 6-bit CRC Multi-frame alignment pattern for frame synchronization/alignment. The "E" bits are used to indicate that the Local Receive E1 has detected errored sub-multi-frames.

CAS Multi-frames

CAS Multiframes are only relevant if the user is using CAS or "Channel Associated Signaling". If the user is implementing Common Channel Signaling then the CAS Multiframe is not available. The exact role of CAS Multiframes is discussed in some detail in Section 6.1.2.1 where Channel Associated Signaling is discussed.

7.1.1.1 Channel Associated Signaling

If the user operates an E1 channel in Channel Associated Signaling (CAS) mode, then the "timeslot 16" octets within each E1 frame will be reserved for signaling. Such signaling would convey information such as "On-Hook", "Off-Hook" conditions, call set-up, control, etc. In CAS, this type of signaling data that is associated with a particular voice channel, will be carried within timeslot 16 of a particular E1 frame; within a CAS Multi-frame.

The CAS is carried in a multiframe structure which consists of 16 consecutive E1 frames. The framing/byte format of a CAS Multiframe is presented in Figure 39.

FIGURE 39. FRAME/BYTE FORMAT OF THE CAS MULTI-FRAME STRUCTURE

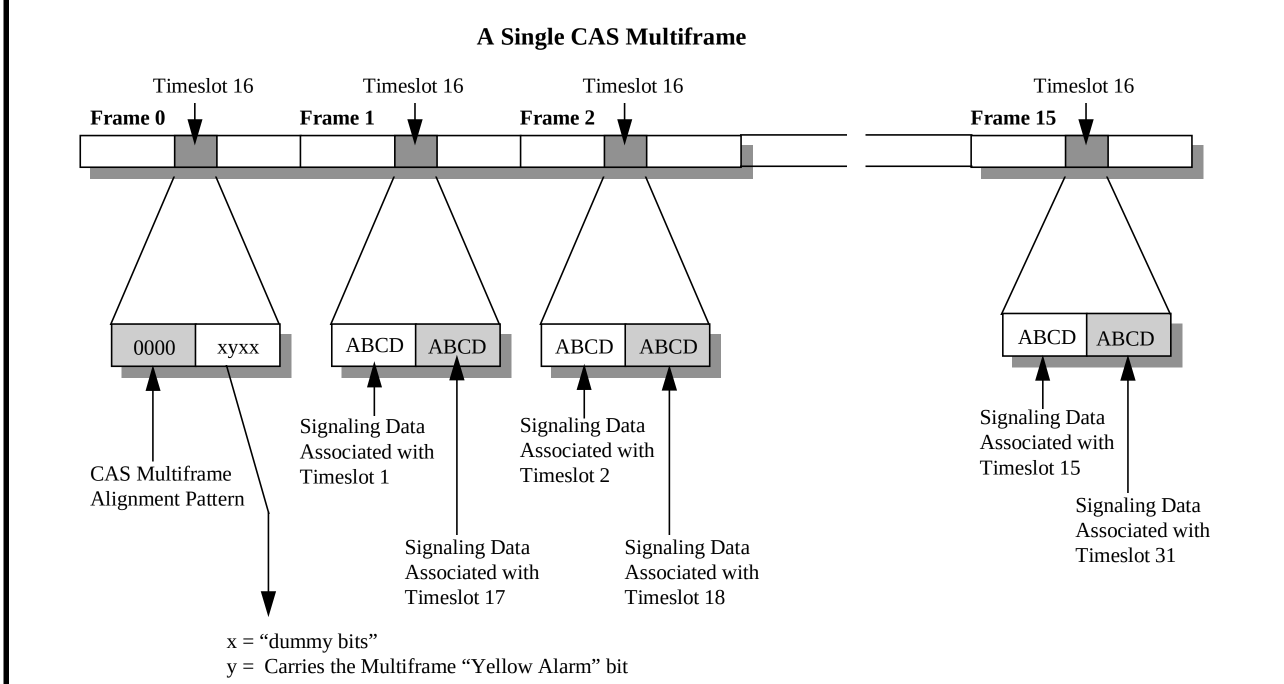


Figure 39 indicates that timeslot 16, within frame 1 of the CAS Multiframe, contains 4 bits of signaling data for voice channel 1 and 4 bits of signaling data for voice channel 17. Likewise, timeslot 16 within frame 2 contains 4 bits of signaling data for voice channels and 4 bits of signaling data for voice channel 18; and so on. Timeslot 16, within frame 0 is a special octet that is used for two purposes.

1. To convey CAS Multiframe alignment information, and
2. To convey "Multi-Frame" alarm information to the Remote Terminal.

The bit-format of timeslot 16, within frame 0 of a CAS Multiframe is presented below: **0000 xyxx**

The upper nibble of this octet contains all zeros and is used to identify itself as the CAS Multiframe alignment signal. If CAS is used, then the user is advised to insure that none of the other timeslot 16 octets contain the value "0000". The lower nibble of this octet contains the expression "xyxx". In this case, the "x" bits are the spare bits and should be set to "0" if not used. The "y" bit is used to indicate a "Multi-Frame" alarm condition to the Remote terminal. During normal operation, this bit-field is cleared to "0". However, if the Local Receive E1 Framer detects a problem with the incoming multiframes, then the Transmit E1 Framer will set this bit-field to "1".

7.1.1.2 Common Channel Signaling (CCS)

Common Channel Signaling is an alternative form of signaling, from Channel Associated Signaling. In CCS, whatever signaling data which is transported via the "outbound" E1 data stream, carries information that applies to all of the voice channels as a set (e.g., timeslots 1 through 15 and 17 through 31) in the E1 frame. There are numerous other variations of Common Channel Signaling that are available. Some of these are listed below.

- 31 Voice Channels, with the common channel signaling being transported via the National Bits.
- 30 Voice Channels (with the common channel signaling data being transported via the National Bits) and CAS data being transported via timeslot 16.
- 30 Voice Channels, with the Common Channel Signaling being processed via timeslot 16. (e.g., Primary Rate ISDN Signaling).

7.1.1.2.1 Transport CCS Data via the National Bits

As mentioned earlier, the Timeslot 0 bits, within the "non-FAS" frames contains five (5) bits which are referred to as the "National Bits". Figure 40 presents an illustration of the bit-format for timeslot 0 within the non-FAS frame.

FIGURE 40. BIT FORMAT OF THE "TIMESLOT 0" OCTET WITHIN A "NON-FAS" E1 FRAME

BIT 0	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7
Si	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
International Bit	Fixed Value	Yellow Alarm	National Bits				

The 5 National bits are uniquely identified as "Sa4" through "Sa8". The XRT84V24 Quad E1 Framer supports the transmission of CCS data via any combination of these "Sa4 - Sa8" bit-fields, within the "outbound" E1 frame. The user selects which of these bits will be used to carry CCS (or data link information) by writing the appropriate data to the "Transmit Signaling and Data Link Select" Register (Address = X00h, 0Ah), as illustrated below.

Transmit Signaling and Data Link Select Register (Address = X00h, 0Ah)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TxSa8 Enable	TxSa7 Enable	TxSa6 Enable	TxSa5 Enable	TxSa4 Enable	TxSIGDL[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Setting any of the bit-fields (from Bit 3 to Bit 7) within this register will configure the XRT84V24 to transport Data Link Information to the Remote Terminal, via the corresponding Sax bit.

In the XRT84V24 Quad E1 Framer, the source of the Data Link Information (which is transported via the National Bits) will be either the Transmit Overhead Input Interface or the Transmit Terminal Input Interface. The user can configure the XRT84V24 to select either one of these blocks as the source for the Data Link information, by writing the appropriate data into the DLSRC[1:0] bits within the "Synchronization MUX Register (Address = 00h, 09h).

7.2 FUNCTION OF OVERHEAD BITS

7.2.1 Timeslot 0 Overhead Bits

All bits, within timeslot 0, whether residing within the FAS or non-FAS frames are considered to be "overhead" bits; because none of these bits are specifically used to carry voice data. The main purpose of timeslot 0 is to support FAS and CRC Multiframe alignment. However, certain additional bits (e.g., the National Bits) can be used to carry data link (or signaling) information.

7.2.1.1 International Bits

7.2.1.2 National Bits

7.2.2 Timeslot 16 Overhead Bits

Timeslot 16 can be used for the following three roles:

1. To carry CAS Signaling
2. To carry CCS Signaling
3. To carry an additional PCM channel

Whenever timeslot 16 is used for the first two roles, then it is considered to be an overhead bit.

7.2.3 Transmit HDLC Controller

The Transmit HDLC Controller can be configured to transmit three different kinds of data link messages.

- Bit Oriented Signaling Messages
- Message Oriented Signaling - Periodic Performance Report Messages
- Message Oriented Signaling - Path or Test Signal Identification Message.

These data link message types are described in section 14; Data Link Controller

8.0 TRANSMIT E1 LIU INTERFACE

The purpose of the Transmit E1 LIU Interface is to take the "outbound" E1 frame data, from the Transmit E1 Framer block, and to do the following.

- To encode the "outbound" E1 frame data into any one of the following formats
- Single Rail (e.g., a binary data stream)
- Dual Rail, AMI Line Code
- Dual Rail, HDB3 Line Code
- To output this "encoded" data to an LIU device via the TxPOS, TxNEG and TxLineClk output pins.

9.0 RECEIVE E1 LIU INTERFACE

The purpose of the Receive E1 LIU Interface is to "receive" either single-rail or "dual-rail" data from an LIU IC, and to do the following.

- Decode this "incoming" data from the Single Rail, AMI or HDB3 line code, and convert it into a binary data stream
- Route this binary data stream to the Receive E1 Framer Block
- Detect and Declare the "Loss of Signal" Condition.

10.0 RECEIVE E1 FRAMER

The E1 Receive Framer provides the synchronization, signaling extraction, alarm indication and per-channel data and signaling conditioning functions. The major blocks in this module are:

- E1 Synchronization Framer
- Frame counters and timing generation
- CRC-4 Verification
- Receive Data Link Interface
- Signaling extractor
- Alarm and error indicator
- Channel and Signaling Conditioning

E1 Synchronization Framer

The E1 Framer establishes frame and multiframe boundaries by searching for frame alignment, CRC multiframe alignment, and channel associated signaling multiframe alignment in the incoming PCM data stream, and provides output clock useful for data conditioning and decoding. User access to the E1 Framer is going through the Microprocessor Interface. The framer incorporates a robust framing algorithm which prevents false synchronization on patterns that mimic the framing bits.

The E1 framer monitors the incoming data stream from the LIU for Loss of Frame, Loss of CRC-Multiframe, and Loss of CAS Multiframe alignment based on user-selectable criteria, and searches for new frame alignment patterns when sync loss is detected. Whenever sync loss is detected, the framer begins an offline search for the new alignment and shifts into RESYNC mode. While the Receive E1 Framer is operating in the RESYNC mode, all output timing signals remains at the old alignment during this period. When one and only one candidate is qualified (based upon the algorithms presented below), the output timing will move to the new alignment at the beginning of the next frame (or multiframe). One frame later, the E1 Receive Framer resumes the normal sync MONITOR mode and outputs valid sync signals.FAS Synchronization

Three steps are involved in the synchronization process.

Step 1 - Finding the FAS frame alignment pattern (in alternating Time Slot 0s)

A given E1 frame is classified into one of two categories: "FAS frame" or "non-FAS frame". The FAS Frame will contain the FAS pattern in Time Slot 0 of the E1 frame (e.g., "Si, 0, 0, 1, 1, 0, 1, 1"); whereas the non-FAS frame will not. The following tables presents the "Time Slot 0" bit format for each of these types of frames.

In a given channel, the E1 frame type will alternate between FAS frame and non-FAS frames. This fact is illustrated below in Table 168.

TABLE 168: TIME SLOT 0 FORMAT FOR FAS AND NON-FAS TYPE E1 FRAMES

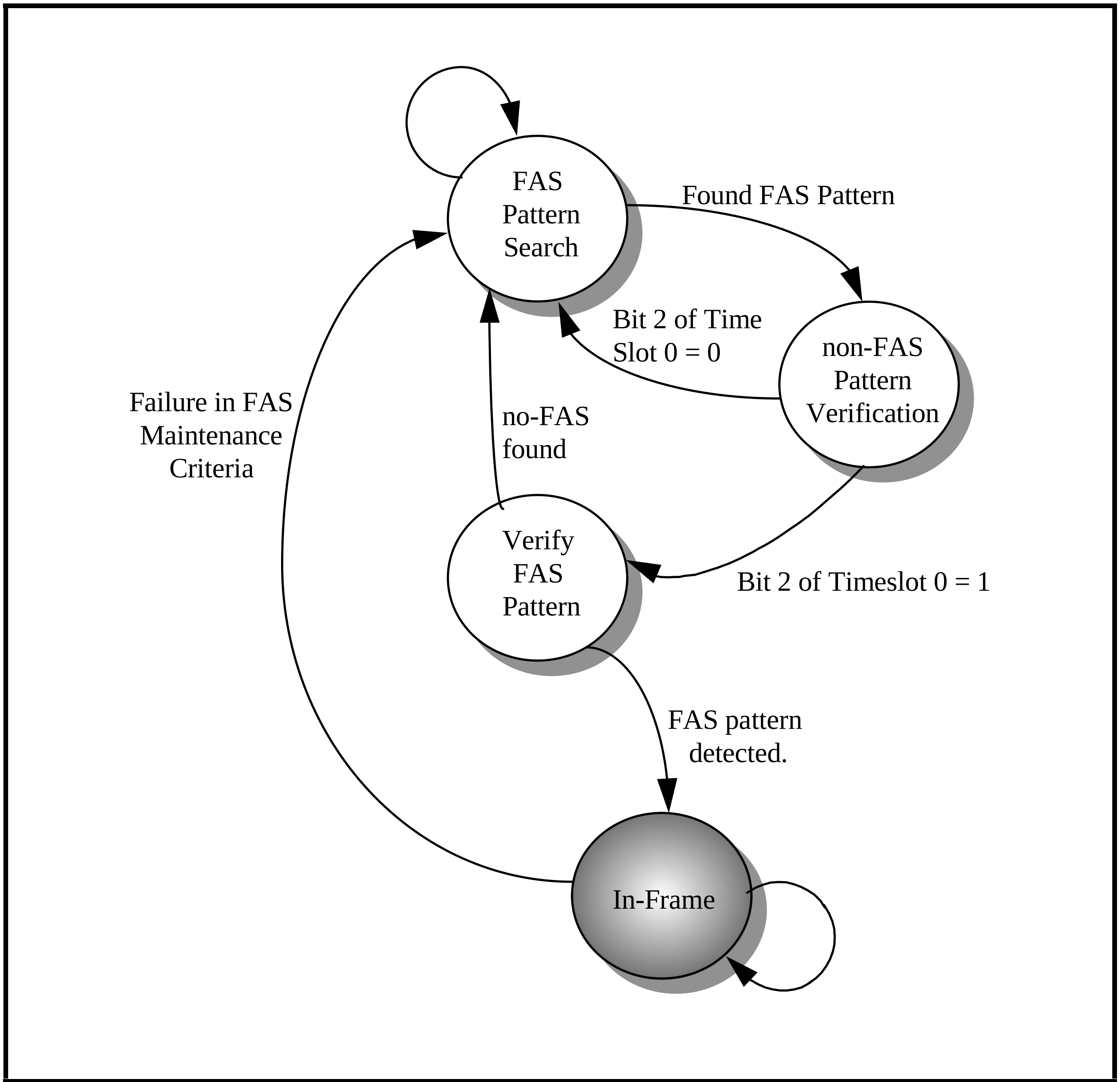
E1 FRAME TYPE	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7	BIT 8
FAS Frame	Si	0	0	1	1	0	1	1
non-FAS Frame	Si	1	A	Sa4	Sa5	Sa6	Sa7	Sa8

The XRT84V24 Quad E1 Framer supports two different FAS Synchronization Algorithms. Each of these algorithms are explained below.

FAS Synchronization Algorithm 1 (FASSEL = 0)

Figure 41 presents a State Machine Diagram for the FAS Synchronization Algorithm number 1.

FIGURE 41. STATE MACHINE DIAGRAM FOR FAS SYNCHRONIZATION ALGORITHM # 1



In all, FAS Synchronization Algorithm # 1consists of the following four states.

- FAS Pattern Search
- non-FAS Pattern Verification
- Verify FAS Pattern
- In-Frame (FAS)

Each of these states are discussed below.

FAS Pattern Search:

The Receive E1 Framer will, upon power up, be initially operating in this state. While the Receive E1 Framer is operating in this state, it will be searching through the incoming E1 frame data for the FAS pattern (Si, 0, 0, 1, 1, 0, 1, 1). Once the Receive E1 Framer has found this sequence of bit-values in the incoming E1 data stream, it will transition to the "non-FAS Pattern Verification" state.

"Non-FAS Pattern" Verification:

It is entirely possible that the Receive E1 Framer, when detecting the (Si, 0, 0, 1, 1, 0, 1, 1) pattern could have been confused by some other timeslot data mimicking the FAS pattern. Hence, the purpose of the "Non-FAS Pattern" Verification state is to further the evaluation of this pattern, and alignment, and determine if the Receive E1 Framer has truly found the FAS pattern in Timeslot 0.

If the current E1 frame is a "FAS frame" (thereby containing the FAS pattern), then it stands to reason that the very next E1 frame will be a non-FAS frame. Hence, this very next E1 frame should definitely not exhibit the FAS pattern. Bit 2, within a "FAS Frame" will be of the value "0". Likewise, Bit 2 within a "non-FAS Frame" will be of the value "1". Hence, in order to verify that this "very next" E1 frame is not a "FAS frame", the Receive E1 Framer will if the FAS pattern is absent in the following frame by verifying that bit 2 of the assumed time slot 0 byte is a "1".

If the Receive E1 Framer detects a "0" in this bit-field, then it will realize that it has been confused by "mimicking" data in the incoming E1 data stream, and will transition back to the "FAS Pattern Search" state. Conversely, if the Receive E1 Framer detects a "1" in this bit-field, then it will realize that this is certainly not an "FAS frame" that it is currently evaluating. At this point, the Receive E1 Framer will then transition to the "Verify FAS Pattern" state.

"Verify FAS Pattern" State

The purpose of this state is to verify that the Receive E1 Framer has found timeslot 0, of the incoming E1 data stream. Hence, in this particular state, the Receive E1 Framer will attempt to verify that it is currently evaluating timeslot 0 of a FAS frame. The Receive E1 Framer will do this by checking bit-field 2 (within this "suspected" timeslot 0) to see if it contains the value "0". If the Receive E1 Framer detects a "0" in this bit-field, then it will transition into the "In-FAS Frame" state. Conversely, if the Receive E1 Framer detects a "1" in this bit-field, then it will transition back to the "FAS Pattern Search" state.

In FAS Frame

Once the Receive E1 Framer achieves FAS Synchronization, it will monitor the alignment signals for errors.

The user can specify the "Loss of Frame" criteria by writing the appropriate values to the "FASC[2:0]" bits within the Framing Control Register, (Address = x0h, 0Bh), as depicted below.

Framing Control Register (Address = x0h, 0Bh)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RSYNC	CASC[1:0]		CRCC[1:0]		FASC[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	1	0	0	0	0	1	1

The relationship between the FASC[2:0] bit-fields and the "Loss of FAS Frame" criteria is shown in Table 169.

TABLE 169: LOSS OF FAS CRITERIA

FASC[2:0]	LOSS OF FAS FRAME CRITERIA
000	Illegal - Do Not Use
001	One Errored FAS Pattern
010	Two Errored FAS Patterns
011	Three Errored FAS Patterns
100	Four Errored FAS Patterns
101	Five Errored FAS Patterns
110	Six Errored FAS Patterns
111	Seven Errored FAS Patterns

Options Associated with the FAS Synchronization Algorithm 1

If either of the conditions in Steps 2 or 3 are not met, a new search for the FAS pattern is initiated in the bit immediately following the errored timeslot 0 location. If both conditions in Steps 2 and 3 are met and frame check sequence is enabled (e.g., if CKSEQ_ENB = 1, within the Framing Select Register), then an additional check sequence is initiated. The check sequence consists of verifying correct frame alignment for an additional two frames.

- Additional Frame 1: Once the FAS pattern is found, check if the FAS pattern is absent in the following frame by verifying the bit 2 of timeslot 0 being a "1". If this test fails, return to Step 1.
- Additional Frame 2: Verify that the FAS pattern is present. If this test fails, then return to Step 1.

The bit-format of the Framing Select Register (Address = x00h, 07h) is presented below.

Framing Select Register (FSR) - Address = x00h, 07h (Indirect)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Unused	CRCDIAG	CASSEL[1:0]		CRCSEL[1:0]		CKSEQ_ENB	FASSEL
RO	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

CRC Synchronization

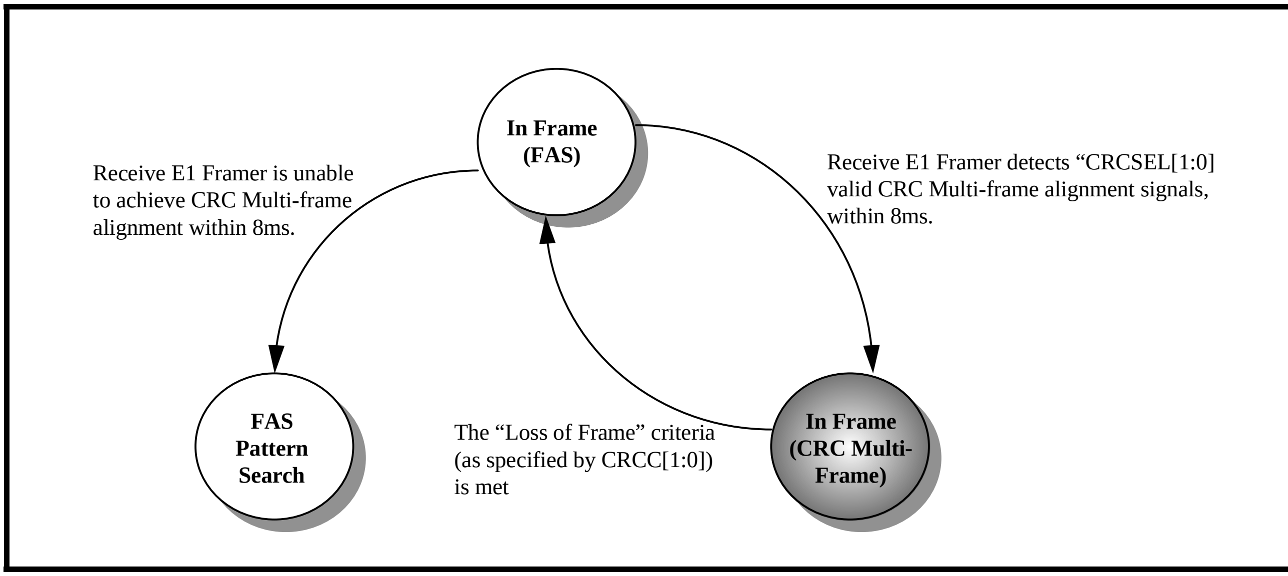
One the Receive E1 Framer has reached the "In-FAS Frame" state, the next step is to find the CRC-4 Multi-frame alignment. The Receive E1 Framer attempts to find the frame boundary of the CRC Multi-frames. Figure 42 presents the "Timeslot 0" bit format of an E1 multi-frame.

FIGURE 42. THE "TIMESLOT 0" BIT-FORMAT OF AN E1 MULTI-FRAME

SMF	FRAME NUMBER	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7	BIT 8
1	0	C1	0	0	1	1	0	1	1
	1	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	2	C2	0	0	1	1	0	1	1
	3	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	4	C3	0	0	1	1	0	1	1
	5	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	6	C4	0	0	1	1	0	1	1
	7	0	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
2	8	C1	0	0	1	1	0	1	1
	9	1	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	10	C2	0	0	1	1	0	1	1
	11	1	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	12	C3	0	0	1	1	0	1	1
	13	E	1	A	Sa4	Sa5	Sa6	Sa7	Sa8
	14	C4	0	0	1	1	0	1	1
	15	E	1	A	Sa4	Sa5	Sa6	Sa7	Sa8

Figure 43 presents a state machine diagram of the "CRC Multi-Frame" framing algorithm.

FIGURE 43. STATE MACHINE DIAGRAM OF THE "CRC MULTI-FRAME" FRAMING ALIGNMENT ALGORITHM



Once the Receive E1 Framer enters the "In Frame (FAS)" state, then it will begin to search for the CRC-4 Multi-frame alignment signals. The CRC-4 Multi-frame Alignment signal consists of the contents of bit-field 1, within timeslot 0 of the non-FAS frame. Hence, the CRC-4 Multiframe alignment pattern consists of a repeating pattern of 0, 0, 1, 0, 1, 1, E, E.

If the Receive E1 Framer detects a "user-selectable" number of valid "CRC Multi-Frame" alignment signals within an 8ms period, then the Receive E1 Framer will transition to the "In-Frame (CRC Multi-Frame)" state. The user can specify the "CRC Multi-Frame" Alignment criteria by writing the appropriate values into the CRCSEL[1:0] bit-fields within the Framing Select Registers (FSR), as depicted below.

Framing Select Register (FSR) - Address = x00h, 07h (Indirect)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
UNUSED	CRCDIAG	CASSEL[1:0]		CRCSEL[1:0]		CKSEQ_ENB	FASSEL
RO	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

The relationship between the values of the bits within CRCSEL[1:0] and the "CRC Multi-Frame" alignment pattern are listed below.

CRCSEL[1:0]	CRC Multi-Frame Alignment Criteria
00	**No CRC Multiframe alignment is enabled.**
01	**CRC Multiframe Alignment is enabled.** Alignment is declared if at least one valid "CRC Multi-Frame" alignment signal is detected within 8ms.
10	**CRC Multiframe Alignment is enabled.** Alignment is declared if at least two (2) valid "CRC Multi-frame" alignment signal is detected within 8ms; with the time separating the two-alignment signals being multiples of 2ms.
11	**CRC Multiframe Alignment is enabled.** Alignment is declared if at least three (3) valid "CRC Multi-frame" alignment signal is detected within 8ms; with the time separating the two-alignment signals being multiples of 2ms.

Once the Receive E1 Framer has found the "user-selected" number of CRC Multiframe alignment signals, within the 8ms period, then the Receive E1 Framer will transition into the "In-Frame (CRC Multi-frame)" state.

If the Receive E1 Framer is unable to acquire "CRC Multiframe" alignment within 8ms after entering the "In-Frame (FAS) state, then the Receive E1 Framer will transition back into the "FAS Pattern Search" state.

In-Frame (CRC Multi-Frame) state

One the Receive E1 Framer has reached the "In Frame (CRC Multi-Frame)" state, then it will begin to monitor the Multi-Frame alignment signals for errors. If the Receive E1 Framer detects a sufficient number of errors, then it will declare a "Loss of CRC Multi-Frame" condition' and will transition back into the "In-Frame (FAS) state. The actually "Loss of CRC Multi-Frame" criteria can be selected by writing the appropriate values into the CRCC[1:0] bits within the Framing Control Register; as depicted below.

Framing Control Register (Address = x0h, 0Bh)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RSYNC	CASC[1:0]		CRCC[1:0]		FASC[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	1	0	0	0	0	1	1

The relationship between the contents of the CRCC[1:0] bit-fields and the "Loss of CRC Multi-Frame" criteria is tabulated below.

CRCC[1:0]	Loss of CRC Multi-frame Criteria
00	Declared if four (4) consecutive CRC Multiframe alignment signals have been received in error.
01	Declared if two (2) consecutive CRC Multiframe alignment signals have been received in error.
10	Declared if eight (8) consecutive CRC Multiframe alignment signals have been received in error.
11	Declared if 915 of more CRC-4 errors have been detected in one second.

CAS Synchronization (applies only if Channel Associated Signaling is used)

After the FAS and CRC Multiframe Alignment has been declared, the third step is to find the CAS Multiframe alignment signal. Two user selectable algorithms are available.

Algorithm 1:

This algorithm monitors the 16th Time slot of each frame and declares CAS Multiframe Alignment when 15 consecutive frames with bits 1 - 4 of time slot 16 not containing the alignment pattern are observed to precede a frame with timeslot 16 containing the correct alignment pattern ("0000"), as depicted below.

	CAS BIT ALLOCATION OF TIME SLOT 16							
Time slot 16 of Frame 0		Time slot 16 of Frame 1		Time slot 16 of Frame 2		....	Time slot 16 of Frame 15	
0000	xyxx	abcd of ch.1	abcd of ch. 17	abcd of ch. 2	abcd of ch. 18	....	abcd of ch. 15	abcd of ch. 31

Algorithm 2:

This algorithm monitors the 16th timeslot of each frame and declares CAS multiframe alignment when non-zero bits 1-4 of timeslot 16 are observed to precede a timeslot 16 containing the correct alignment pattern.

Once the CAS Multiframe Alignment is found, the Out of CAS Multiframe Alignment indication is cleared. CAS Synchronizer monitors the multiframe alignment signal, indicates errors occurring in the 4-bit alignment pattern, and indicates debounced values of the remote signaling multiframe alarm bit (bit 6 of timeslot 16 of frame 0 of the multiframe).

When synchronization is achieved, the framer monitors multiframe alignment signals for errors. The CAS LOF indication turns on if frame alignment is lost. The criteria for Loss of CAS Multiframe is dictated by the CASC bits in the E1 Framing Control Register shown below.

Framing Control Register (FCR) - Address = x00h, 0Bh (Indirect)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RSYNC	CASC[1:0]		CRCC[1:0]		FASC[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	1	0	0	0	0	1	1

Bits 6, 5 - CASC[1:0] - Loss of CAS Multiframe Criteria

These two bit-fields combine to allow the user to specify the "Loss of CAS Multiframe Criteria" for the Receive Framer. The relationship between these bit-fields and the "Loss of CAS Multi-Frame" Criteria follow:

CASC[1:0]	LOSS OF CAS MULTIFRAME CRITERIA
00	Two consecutive Multiframe Alignment Signals (MAS) errors.
01	Three consecutive Multiframe Alignment Signals (MAS) errors
10	Four consecutive Multiframe Alignment Signals (MAS) errors
11	Eight consecutive Multiframe Alignment Signals (MAS) errors

Frame Counters and Timing Generation

Receive Frame and Multiframe counters and timing generators provide timing for frame and multiframe alignment, CRC-4 check, signaling extraction, facility data link extraction, yellow alarm, and all the timing for per-channel parameter fetch. The data extracted from this timing is placed into the appropriate internal storage elements for the microprocessor to access. The information extracted is not valid unless the receive module has achieved valid synchronization.

CRC-4 Verification

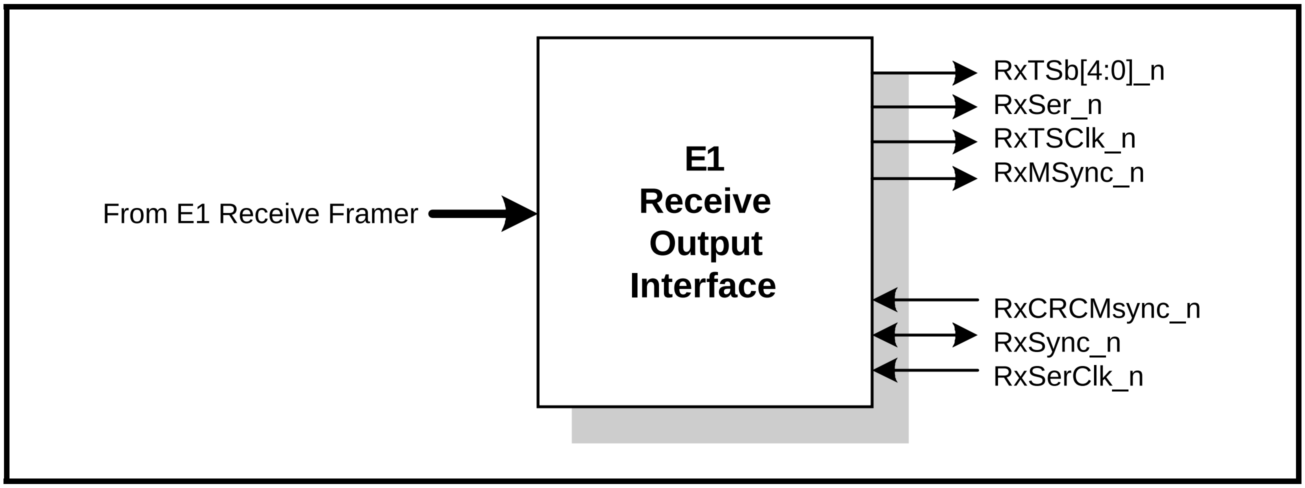
The CRC Verification is performed by calculating the 4-bit CRC checksum for each incoming sub-multiframe and comparing the results to the received CRC remainder bits in the subsequent sub-Multi-Frame. The CRC errors are accumulated over one second intervals. Optionally, a CRC frame resync can be initiated with 915 or more CRC-4 errors occur in one second. The number of CRC errors accumulated during the previous second is available by reading the E1 Receive Synchronization Bit Error Counter.

11.0 RECEIVE E1 OVERHEAD OUTPUT INTERFACE

12.0 RECEIVE E1 OUTPUT INTERFACE

In the Receive section (of each framer) the incoming data stream is either written into a slip buffer or output directly to this receive interface. This interface consists of a data output signal (RxSer), an input/output framing reference signal (RxSync), a CRC Multiframe reference output signal (RxCRCMSync), a CAS multi-frame reference output signal (RxCASMSync), a channel clock output signal (RxChClk) pulsing high at the end of each timeslot, and receive time slot indicator signals (RxChn[4:0]). Figure 44 shows a simple illustration of the Receive E1 Output Interface block.

FIGURE 44. BLOCK DIAGRAM OF E1 RECEIVE OUTPUT INTERFACE



The role of each of the Receive E1 Output Interface block signals is described, in detail below.

RxSer_n - Receive Data Serial Output pin

All "end-user" information which is embedded in the "inbound" E1 frames, is output to the Terminal Equipment via this pin. If "channelized" E1 frames are being processed, then the content of timeslots 1 through 15 and 17 through 31 (within the incoming E1 frames) are output via this pin.

The data from this pin is updated on the rising edge of the RxSerClk pin.

RxSerClk_n - Receive E1 Output Interface Clock Input pin

This 2.048MHz clock signal is used to clock the data out via the RxSer_n pin.

NOTE: *If the slip buffer is enabled, then it will be depleted (and routed to the RxSer_n output pin) upon the rising edge of this clock signal.*

RxCASMSync_n - Receive CAS Multiframe Boundary Indicator output pin

This output pin pulses "high" while the Receive E1 Output Interface outputs the first bit, within a given CAS Multi-frame. This pin has a nominal pulse rates of 500Hz.

RxCRCMSync_n - Receive CRC Multiframe Boundary Indicator output pin

This output pin pulses "high" while the Receive E1 Output Interface outputs the first bit, within a given CRC Multi-frame. This pin has a nominal pulse rate of 500Hz.

RxTSClk_n - Receive Channel Clock Output

This 256kHz clock output pin pulses "high" while the Receive E1 Output Interface outputs the LSB of a given timeslot octet.

RxTSb[4:0]_n - Receive Channel Number

These five (5) output pins reflect the binary values of the timeslot number which is being output via the Receive E1 Output Interface.

RxSync_n - Receive Sync Input/Output pin

This pin can be configured to be either an input or an output. The "direction" of this pin is selected via the "SB_DIR" bit within the "Slip Buffer Control" Register.

Input

The Terminal Equipment should assert this signal while it is reading out the first bit, within a given E1 frame, from the Receive E1 Output Interface.

Output

The Receive E1 Output Interface will "pulse" this signal "high" while it outputs the first bit, within a given E1 frame.

12.1 SLIP BUFFER

The Receive E1 Output Interface, for each of the four framers is equipped with an on-board, two-frame (e.g., 512 bits) elastic store buffer. The slip buffer can be enabled or disabled via programming bits SB_ENB in the Slip Buffer Control Register (SBCR), as depicted below.

Slip Buffer Control Register (SBCR) - Address = x00h, 16h (Indirect)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Unused			SB_FORCESF	SB_SFENB	SB_SDIR	SB_ENB[1:0]	
RO	RO	RO	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	1	0	1

The following table relates the contents of these bit-fields to the resulting operation of the slip buffer.

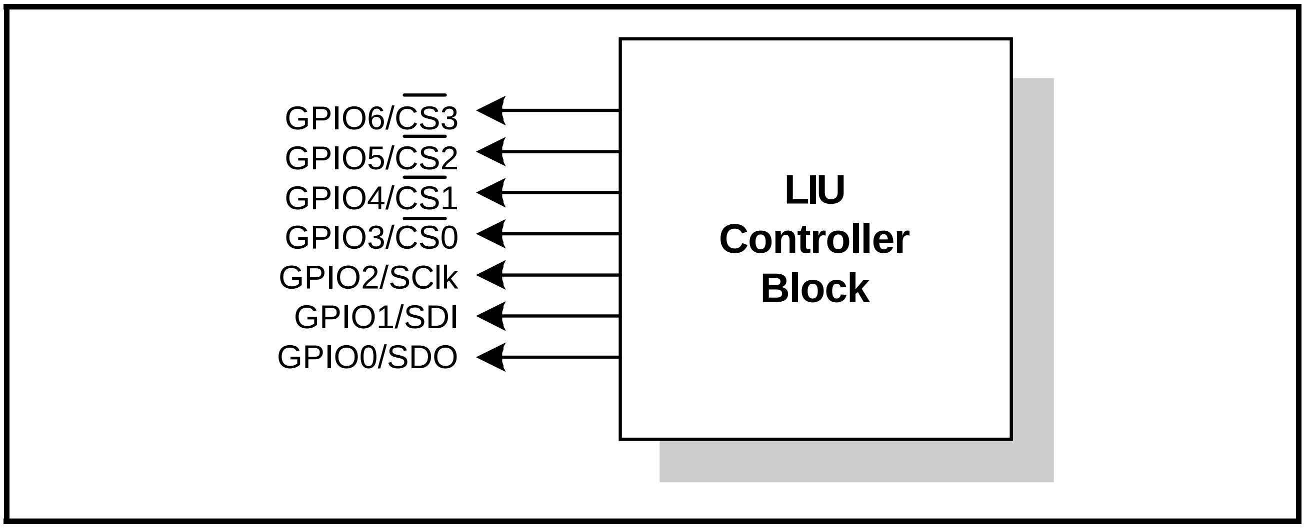
SB_ENB[1:0]	RESULTING MODE OF OPERATION
00	Buffer is by-passed
01	Elastic Store is enabled
10	Buffer acts as a FIFO. The data latency is dictated by the settings within the FIFO Latency Register.
11	Buffer is by-passed

If the elastic store is enabled, then the user has the option of either providing a frame sync or getting a frame pulse on the frame boundary at the RxSync pin. If the elastic buffer either fills or empties, a controlled slip will occur. If the buffer empties and a read occurs, then a full frame of data will be repeated and a status bit will be updated. If the buffer fills and a write comes, then a full frame of data will be deleted and another status bit will be set. If the slip-buffer is bypassed (SB_ENB[1:0]), the received clock, sync and data are output to pins directly following receive timing. If SB_ENB = 2, then the slip buffer is put into a FIFO mode. In the FIFO mode, the slip buffer is acting like a standard first-in-first out storage. The sync signal can be either input or output. A fixed read and write latency is maintained in a programmable fashion controlled by the FIFO Latency Register.

13.0 LIU CONTROLLER BLOCK

The LIU Controller Block provides the user with a significant amount of on-chip Control and Monitoring capability. A simple block diagram of the LIU Controller Block is presented below in Figure 45.

FIGURE 45. A SIMPLE BLOCK DIAGRAM OF THE LIU CONTROLLER BLOCK



The Line Control Register (LCR) shown below, allows the user to control the operations of external line interface devices. The LIU Controller block can be configured to operate in one of two modes.

- Host Mode
- Hardware Mode

The user can configure the LIU to operate in one of these modes by writing the appropriate value into Bit 7 (MODE) within the Line Control Register; as depicted below.

Line Control Register (LCR) - Address = x00h, 02h (Indirect)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MODE	GPO[6]/ CEDGE	GPO[5]	GPO[4]	GPO[3]	GPO[2]	GPO[1]	GPO[0]
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

Writing a "0" to this bit-fields configures the LIU Controller Block to operate in the "Hardware" Mode. Writing a "1" to this bit-field configures the LIU Controller Block to operate in the "Host" Mode. Each of these LIU Controller Block operating modes are discussed below.

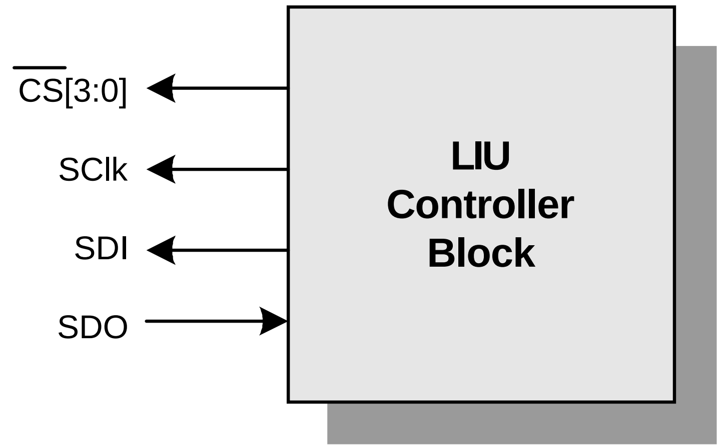
13.1 THE HARDWARE MODE

If the Quad E1 Framer device is Configured to operate in the Hardware Mode, then output pins GP0 through GP6 become "General Purpose" output pins. Additionally, the state of the seven output pins: GP[6] through GP[0] are controlled via the contents of within bits 6 though 0 in the Line Control Register. Writing a "1" into a particular bit-field, within Bit 6 through Bit 0, will result in the corresponding output pin toggling "high". Likewise, writing a "0" into a particular bit-field, within Bits 6 through 0, will result in the corresponding output pin toggling "low". A typical application of the Line Control Register, along with these associated output pins allows the user to toggle and control various discrete input signals of the LIU. Some examples are this usage are Transmit All Ones Select (TAOS), Local Loopback (LLOOP), Remote Loopback (RLOOP), LIU Length Selection (LENS), etc.

13.2 THE HOST MODE

In the "Host" Mode, the GP[6] through GP[0] pins now assume the role of supplying signals that permit data transfers over a Microprocessor Serial interface. Figure 46 presents a simple block diagram of the LIU Controller Block, when it is operating in the "Host" Mode.

FIGURE 46. A SIMPLE BLOCK DIAGRAM OF THE LIU CONTROLLER BLOCK, WHEN IT IS OPERATING IN THE "HOST" MODE



When the LIU Controller block is operating in the "Host" Mode, then it consists of four (4) chip select output pins, a serial clock output pin (SCLK), a "Serial Data Input" (SDI) output pin, and a "Serial Data Out" (SDO) input pin. The "Host" mode is useful for interfacing to and configuring LIU devices that contains a Microprocessor Serial Interface.

NOTE: *In this mode, the input/output pins used to control LIU devices are serving as a serial interface port. A serial port controller writes the newly written data into two LIU Access Registers (LAR1 and LAR2) into the line interface unit device by creating Chip Select (CS_L), Serial Clock (SCLK), To Serial Data Input (To_SDI), and From Serial Data Output (Fr_SDO) signals. This control function is driven off of the transmit clock; therefore RxClk must be present for proper operation. Refer to interface signals section for signal description*

14.0 DATA LINK CONTROLLER

The E1 framer provides a serial data link through either E1 National Bits or Time Slot 16 signaling channel (CCS). The data link controller handles three major functions associated with E1 framing format. They are Transmit/Receive LAPD Controller, and Bit-Oriented Signal Processor. There exists two 96-byte transmit message buffers and two 96 byte receive message buffers in shared memory to accommodate transmitting and receiving information. The Data Link Control Register (Indirect Address = x0h, 13h) is used to configure and conduct the various operations associated with all the data link functions. The description of each follows.

Data Link Control Register, Address = x0h, 13h

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DIAG	MOSA	FCSDIS	Auto Rx	ABORT	IDLE	FCS	LAPD
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

Bit 7 - DIAG

Reserved for diagnostic purposes. This bit should be set to zero in normal operations.

Bit 6 - MOSA

This bit-fields allows the user to insert an MOS abort sequence during the transition from MOS mode to BOS mode.

Writing a "0" inserts a MOS abort sequence during the transition, if a MOS message is interrupted. Writing a "1" configures the Transmit LAPD Controller to not insert an abort sequence.

Bit 5 - FCSDIS (Frame Check Sequence - Disable)

This "Read/Write" bit-field allows the user to enable or disable the Frame Check Sequence Verification process. Writing a "1" disables Frame Check Sequence octet verification.

Bit 4 - AutoRx

Bit 3 - ABORT

Bit 2 - IDLE

This "Read/Write" bit-field allows the user to transmit a continuous stream of Flag Sequence octets (7Eh) on the data link channel.

Bit 1 - FCS

Bit 0 - LAPD

This "Read/Write" bit-field allows the user to specify whether the HDLC Controller will be used to transmit MOS or BOS messages. Writing a "1" to this bit-field configures the HDLC Controller to transmit and receive MOS HDLC messages. Writing a "0" to this bit-field configures the HDLC Controller to transmit and receive BOS HDLC Messages.

A 7-bit Transmit Data Link Byte Count register holds the length of the message to be transferred. In Bit-Oriented Signal Transmission, this count should contain the value of the number of message transmission repetitions before each transmit interrupt (TxEOT). If this value is set to "0", then the message will be transmitted indefinitely, and no interrupt (TxEOT) will be generated. The bit-format of this register is presented below.

Transmit Data Link Byte Count Register, Address = x0h, 14h

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
BUFAVAL/ BUFSEL	TDLBC[6:0]						
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

The Data Link Status Register shown below, contains status information about the data link operations. This particular location needs to be read to interpret data link interrupts. The status indicators are changed only when a data link interrupt is set, so they are valid until read. Reading this register clears the associated interrupt if Reset upon Read is selected in the Interrupt Control Register. Otherwise, a write-to-clear is necessary.

Data Link Status Register, Address = xAh, 06h

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MSG TYPE	TxSOT	RxSOT	TxEOT	RxEOT	FCSERROR	RxABORT	RxIDLE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

BIT-ORIENTED SIGNAL (BOS) PROCESSOR

Transmit

To transmit a bit-oriented signal, a repeating message is sent of the form "**0xxxxxx011111111**", where the "**xxxxxx**" represents a six bit message. The eight bits that are to be transmitted in the form "**0xxxxxx0**" are loaded into the first location of the transmit message buffer. A zero is then written into Bit 0 of the Data Link Control Register, which sets the transmitter to bit-oriented mode. The Transmit Data Link Byte Count register should contain the value of message transmission repetitions before each TxEOT transmit interrupt. If the value is set to "0", then the message will be transmitted for an indefinite number of times and no interrupt will be generated.

If the IDLE bit, within the Data Link Control Register, is set, then repeated flag sequence octets will be transmitted, during the idle periods between the transmission of BOS data.

If the ABORT bit is set in this register, then a BOS abort sequence (e.g., a string of 9 consecutive "1's") is activated followed by an all "1's" transmission. All data link bits will be set to "1" after the transmission of the current message.

Switching the Data Link Mode from LAPD to BOS, while a message (MOS type) will interrupt the MOS Message, after the octet in progress is transmitted. If the MOSA bit is set, then a MOS abort sequence string (e.g., a "**01111111**") will be inserted before switching. Switching the Data Link Mode from BOS to

LAPD will not take place until the current operation completes. However, if the BOS byte count is set to zero initially, then the transition will occur immediately after transmission of the current message octet.

Receive

In Bit-Oriented Signal mode, this processor will generate the RxEOT interrupt each time a message is received. Interrupt is caused when the bit-oriented message of "**111111110xxxxxx0**" is received. The message content, (which is bounded by "zeros") will be stored in the receive data link buffer. If nine "1's" are received, the data link will be set to ABORT (ABORT status); no further interrupts will occur unless a bit-oriented signal message or a flag sequence octet is received.

Transmit LAPD Controller

The transmit LAPD Controller implements the Message-Oriented Protocol based upon ITU-T Q.921 LAPD type of protocol. The functions performed by this controller are zero stuffing, frame check sequence generation, flag octet insertions, abort sequence generation, E1 transmitter interface, and transmit buffer access.

Two 96-byte buffers in shared memory are allocated for the LAPD Transmitter to reduce the frequency of microprocessor interrupts and to alleviate the response time requirement for the microprocessor to handle each interrupt. There are no restrictions on the length of the message. However, the 96 byte buffer is deep enough to hold one entire LAPD Path or Test Signal Identification Message. the block diagram of both transmit and receive LAPD Controller.

15.0 FRAME AND MULTIFRAME COUNTERS AND TIMING GENERATORS

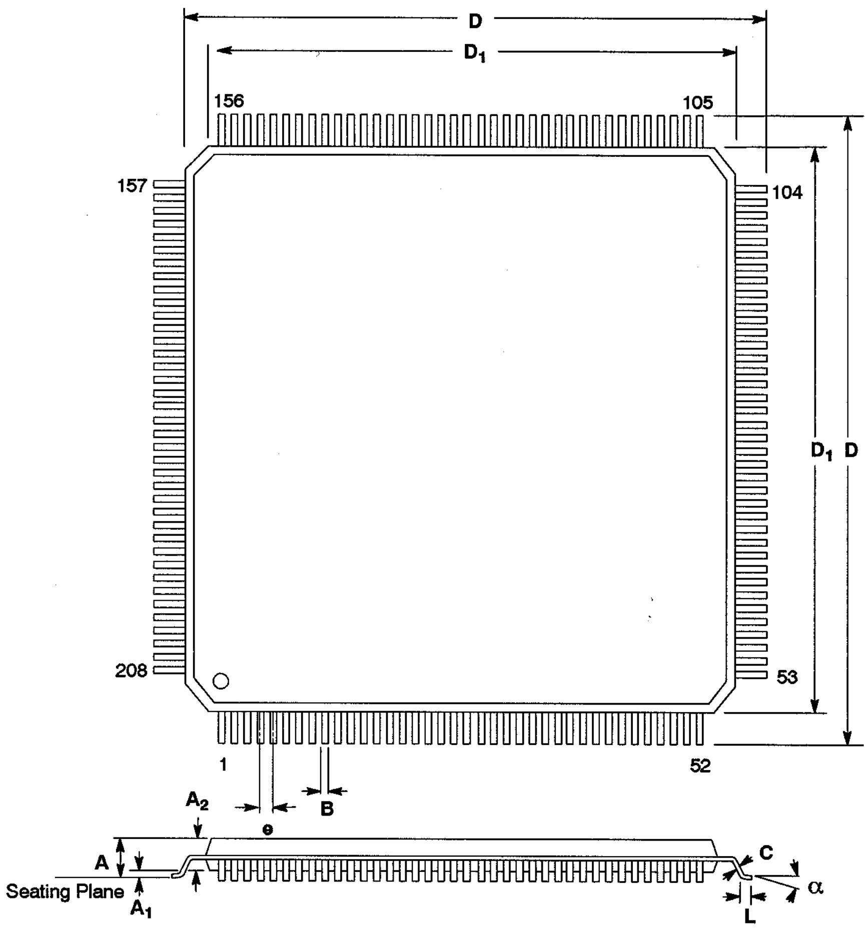
Frame and Multiframe counters and timing generators provides timing for frame, CAS multiframe alignment, CRC-4 multiframe, CRC-4 bits, signaling, data link, yellow alarm, National bits, and all timing for per channel parameter fetch. The data derived from this timing are sent to transmit multiplexers, where they are added to the raw input data stream to format the raw data into the appropriate E1 format, according to the selected mode of operation. The timing generator also produces the Transmit Maximum (TxMX) signal, which pulses high for one TxClk cycle coincident with the sampling of the last bit of the multiframe.

The frame and multiframe counters are initialized to zero after power up; this means that they are pointing to the first bit location of the frame or multiframe. When the Transmit Synchronize pins (TxSync, TxMSync) are configured as inputs signals by setting the SYNCDIR bit (bit 3) of Synchronization Select Register to zero, these counters may be reset to establish the frame and multiframe boundaries when the TxSync and TxMSync inputs are pulsed high. If the TxSync pins are configured as output signals, then they will pulse high at frame and multiframe boundaries. TxMSync provides the synchronization for both CRC and CAS Multiframe alignments and there is only one multiframe counter which produces timing for these multiframe alignments.

PACKAGE DIMENSIONS

208 LEAD PLASTIC QUAD FLAT PACK
(28 mm x 28 mm, QFP)

Rev. 1.00



SYMBOL	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.128	0.161	3.25	4.10
A_1	0.002	0.020	0.05	0.50
A_2	0.126	0.142	3.20	3.60
B	0.007	0.011	0.17	0.27
C	0.004	0.008	0.09	0.20
D	1.197	1.212	30.40	30.80
D_1	1.098	1.106	27.90	28.10
e	0.0197 BSC		0.50 BSC	
L	0.018	0.030	0.45	0.75
α	0°	8°	0°	8°

Note: The control dimension is the millimeter column

REVISION HISTORY

Rev. 1.0.2 added missing sections, renamed some pins, put pin list in by function, modified block diagram.

Rev. 1.0.3 Minor edits and added info in some register tables.

Rev. 1.0.4 More minor edits.

NOTICE

Datasheet June 2001