Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!


## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

## Z80230/Z85230/L

# Enhanced Serial
## Communications Controller

## Product Specification

PS005308-0609

Copyright ©2009 by Zilog®, Inc. All rights reserved.
www.zilog.com

**Warning:** DO NOT USE IN LIFE SUPPORT

## LIFE SUPPORT POLICY

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

### As used herein

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

### Document Disclaimer

©2009 by Zilog, Inc. All rights reserved. Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.

Z8 is a registered trademark of Zilog, Inc. All other product or service names are the property of their respective owners.

# Revision History

Each instance in Revision History reflects a change to this document from its previous revision. For more details, refer to the corresponding pages and appropriate links in the table below.

| Date | Revision Level | Description | Page No |
|---|---|---|---|
| June 2009 | 08 | Removed Security Watermark from pages | all |
| May 2009 | 07 | Minor update to page 107 | 107 |
| May 2009 | 06 | system update change only - no technical content revised | n/a |
| Mar 2009 | 05 | Updated document to add 3V product information<br>Removed ISO/BSI certification information<br>Figure 1, 7 and 23 changed 5V to Vcc<br>Added Z8523L DC Characteristics<br>Updated Read and Write AC Characteristics<br>Updated System Timing Characteristics<br>Updated General Timing Diagram<br>Ordering Information updated<br>Updated Standard Test Conditions<br>Updatred Table 43<br>Updated Table 49 - min value | Misc<br>ii<br>2, 13, 76<br>78<br>90<br>98<br>94<br>107<br>75<br>78<br>98 |
| June 2008 | 04 | Updated as per new template and Style Guide.<br>Updated Figure 4. | All<br>3 |
| September 2007 | 03 | Updated Figure 38 and Implemented Style Guide | All |
| November 2002 | 02 | Editorial Updates | All |
| August 2001 | 01 | Original Issue | All |

# Table of Contents

# Pin Descriptions

The Enhanced Serial Communication Controller (ESCC) pins are divided into seven functional groups:

1. Address/Data
2. Bus Timing and Reset
3. Device Control
4. Interrupt
5. Serial Data (both channels)
6. Peripheral Control (both channels)
7. Clocks (both channels)

Figure 1 on page 2 and Figure 2 on page 2 display the pins in each functional group for both the Z80230 and Z85230/L. The pin functions are unique to each bus interface version in the Address/Data group, Bus Timing and Reset group, and Device Control group.

The Address/Data group consists of the bidirectional lines used to transfer data between the CPU and the ESCC (addresses in the Z80230 are latched by $\overline{AS}$). The direction of these lines depends on whether the operation is a Read or a Write operation.

The Timing and Control groups designate the type of transaction to occur and the timing of the occurrence. The interrupt group provides inputs and outputs for handling and prioritizing interrupts. The remaining groups are divided into Channel A and Channel B groups for:

- Serial Data (Transmit or Receive)
- Peripheral Control (such as DMA or modem)
- Input and Output Line for the Receive and Transmit Clocks

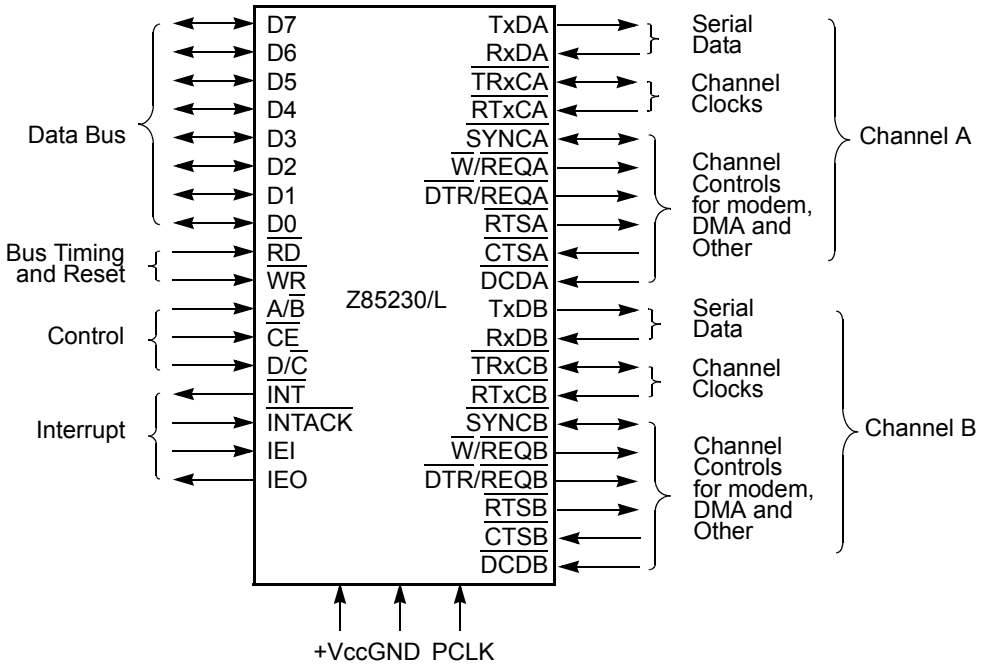


**Figure 1. Z85230/L Pin Functions**

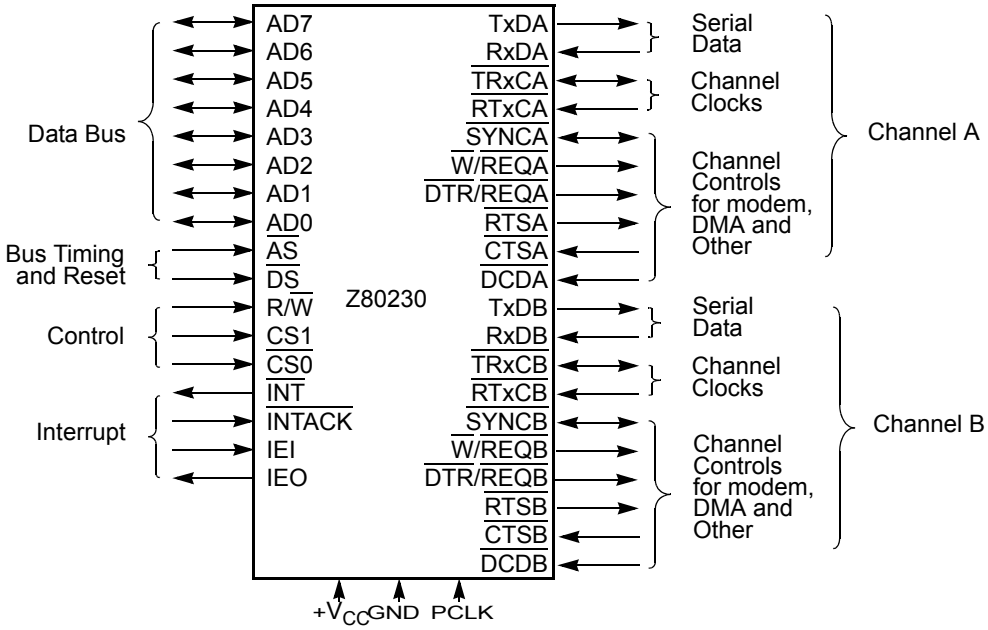


**Figure 2. Z80230 Pin Functions**

Figure 3 displays the Z85230/L DIP and PLCC pin assignments, respectively. Figure 4 displays the Z80230 DIP and PLCC pin assignments.

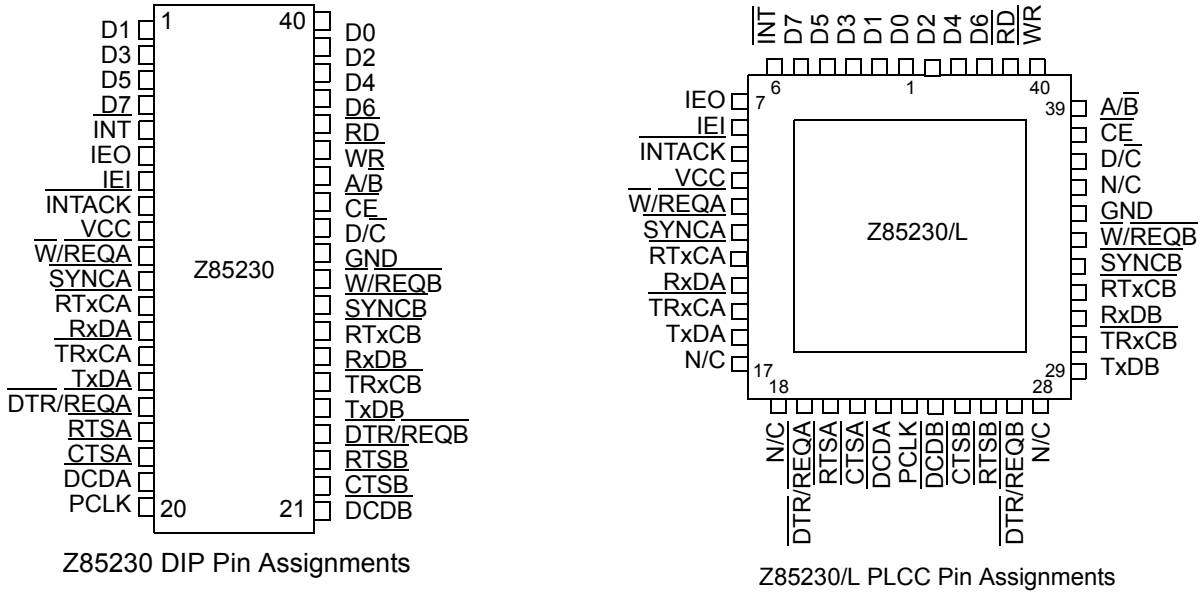


**Figure 3. Z85230/L Pin Assignments**

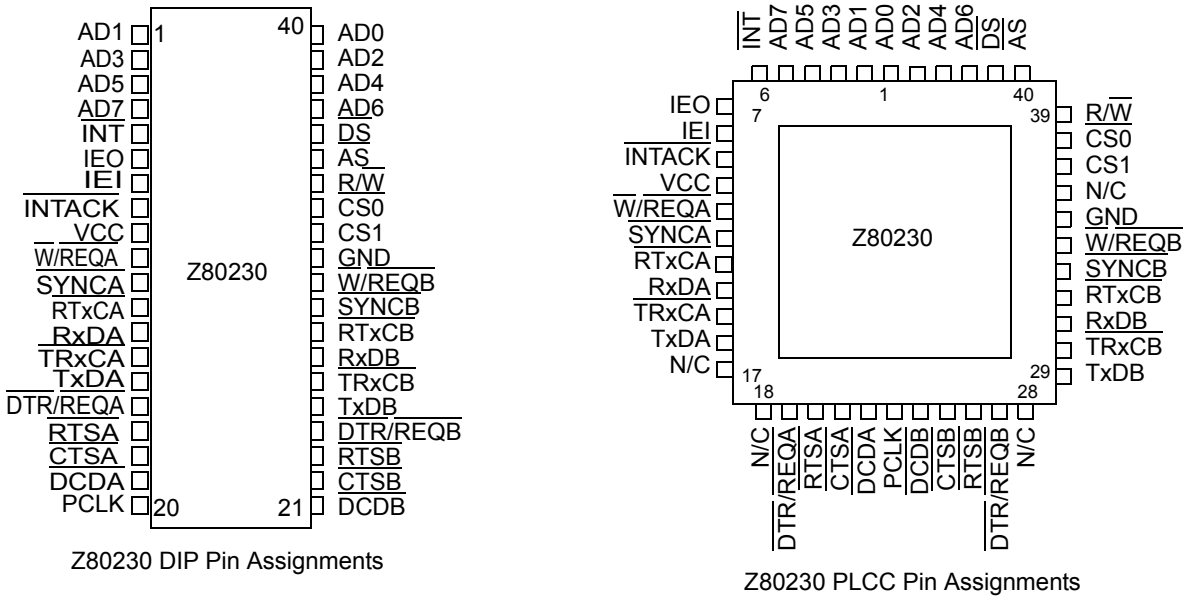


**Figure 4. Z80230 Pin Assignments**

## Pins Common to Both Z85230/L and Z80230

The pin descriptions for pins common to both Z85230/L and Z80230 are provided below:

**$\overline{\text{CTSA}}$, $\overline{\text{CTSB}}$ (Clear To Send (Inputs, Active Low))—**These pins function as transmitter enables if they are programmed for AUTO ENABLE (WR3 bit 5 is 1), in which case a Low on each input enables the respective transmitter. If not programmed as AUTO ENABLE, the pins may be used as general-purpose inputs. These pins are Schmitt-trigger buffered to accommodate slow rise-time inputs. The ESCC detects pulses on these pins and may interrupt the CPU on both logic level transitions.

**$\overline{\text{DCDA}}$, $\overline{\text{DCDB}}$ (Data Carrier Detect (Inputs, Active Low))—**These pins function as receiver enables if they are programmed for AUTO ENABLE (WR3 bit 5 is 1); otherwise, they are used as general-purpose input pins. The pins are Schmitt-trigger buffered to accommodate slow rise-time signals. The ESCC detects pulses on these pins and may interrupt the CPU on both logic level transitions.

**$\overline{\text{RTSA}}$, $\overline{\text{RTSB}}$ (Request To Send (Outputs, Active Low))—**The $\overline{\text{RTS}}$ pins can be used as general-purpose outputs or with the AUTO ENABLE feature. When AUTO-ENABLE is off, these pins follow the inverse state of WR5 bit 1. When used with the AUTO-ENABLE feature in ASYNCHRONOUS mode, this pin immediately goes Low when WR5 bit 1 is 1. When WR5 bit 0 is 0, this pin remains Low until the transmitter is empty.

In Synchronous Data Link Control (SDLC) mode, the $\overline{\text{RTS}}$ pins can be programmed to be deasserted when the closing flag of the message clears the TxD pin, if WR7' bit 2 is 1, WR10 bit 2 is 0, and WR5 bit 1 is 0.

**$\overline{\text{SYNCA}}$, $\overline{\text{SYNCB}}$ (Synchronization (Inputs Or Outputs, Active Low))—**These pins can act either as inputs, outputs, or as part of the crystal oscillator circuit. In the ASYNCHRONOUS RECEIVE mode (crystal oscillator option not selected), these pins are inputs similar to $\overline{\text{CTS}}$ and $\overline{\text{DCD}}$. In this mode, transition on these lines affect the state of the SYNC/HUNT status bits in Read Register 0 but have no other function.

In EXTERNAL SYNCHRONIZATION mode, with the crystal oscillator not selected, these lines also act as inputs. In this mode, $\overline{\text{SYNC}}$ is driven Low, two Rx clock cycles after the last bit of the $\overline{\text{SYNC}}$ character is received. Character assembly begins on the rising edge of the receive clock immediately preceding the activation of $\overline{\text{SYNC}}$.

In the INTERNAL SYNCHRONIZATION mode (MONOSYNC and BISYNC) with the crystal oscillator not selected, these pins act as outputs. These outputs go Low each time a SYNC pattern is recognized, regardless of character boundaries. In SDLC mode, pins switch from input to output when MONOSYNC, BISYNC, or SDLC is programmed in WR4 and SYNC modes are enabled.

**$\overline{\text{DTR}}$/$\overline{\text{REQA}}$, $\overline{\text{DTR}}$/$\overline{\text{REQB}}$ (Data Terminal Ready/Request (Output, Active Low))—**

These pins can be programmed (WR14 bit 2) to serve either as general-purpose outputs or as DMA Request lines. When programmed for DTR function (WR14 bit 2 is 0), these outputs follow the inverse of the DTR bit of Write Register 5 (WR5 bit 7). When programmed for REQUEST mode these pins serve as DMA Requests for the transmitter.

When used as DMA Request line (WR14 bit 2 is 1), the timing for the deactivation request can be programmed in Write Register 7′ (WR7') bit 4. If this bit is 1, the $\overline{DTR}/\overline{REQ}$ pin is deactivated with the same timing as the $\overline{W}/\overline{REQ}$ pin. If 0, the deactivation timing of $\overline{DTR}/\overline{REQ}$ pin is four clock cycles, the same as in the Z80C30/Z85C30.

**$\overline{W}/\overline{REQA}$, $\overline{W}/\overline{REQB}$ (Wait/request (Output, Open-drain When Programmed For $\overline{WAIT}$ Function, Driven High And Low When Programmed For Request Function))—**These dual-purpose outputs may be programmed as REQUEST lines for a DMA controller or as WAIT lines to synchronize the CPU to the ESCC data rate. The reset state is WAIT.

**RxDA, RxDB (Receive Data (inputs, active High))—**These inputs receive serial data at standard Transistor-Transistor Logic (TTL) levels.

**$\overline{RTxCA}$, $\overline{RTxCB}$ (Receive/Transmit Clocks (Input, Active Low))—**These pins can be programmed to several modes of operation. In each channel, RTxC may supply the following:

- Receive clock and/or the transmit clock
- Clock for the baud rate generator (BRG)
- Clock for the Digital Phase-Locked Loop

These pins can also be programmed for use with the respective $\overline{SYNC}$ pins as a crystal oscillator. The receive clock may be 1, 16, 32, or 64 times the data rate in ASYNCHRONOUS modes.

**TxDA, TxDB (Transmit Data (Output, Active High))—**These output transmit serial data at standard TTL levels.

**$\overline{TRxCA}$, $\overline{TRxCB}$ (Transmit/Receive Clocks (Input or Output, Active Low))—**These pins can be programmed in several different modes. When configured as an input, the TRxC may supply the receive clock and/or the transmit clock. When configured as an output, TRxC can echo the clock output of the Digital Phase-Locked Loop, the crystal oscillator, the BRG or the transmit clock.

**PCLK (Clock (Input))—**This clock is the master ESCC clock used to synchronize internal signals. PCLK is a TTL level signal. PCLK is not required to have any phase relationship with the master system clock.

**IEI (Interrupt Enable In (Input, Active High))—**IEI is used with IEO to form an interrupt daisy chain when there is more than one interrupt-driven device. A High IEI indicates that no higher priority device has an Interrupt Under Service (IUS) or is requesting an interrupt.

**IEO (Interrupt Enable Out (Output, Active High))—**IEO is High only if IEI is High and the CPU is not servicing an ESCC interrupt. During an Interrupt Acknowledge Cycle, IEO is also driven Low if the ESCC is requesting an interrupt. IEO can be connected to the next lower priority device's IEI input, and in this case inhibits interrupts from lower priority devices.

**INT (Interrupt (Output, Open-Drain, Active Low))—**This pin activates when the ESCC requests an interrupt. The $\overline{\text{INT}}$ is an open-drain output.

**$\overline{\text{INTACK}}$ (Interrupt Acknowledge (Input, Active Low))—**This pin is a strobe which indicates that an Interrupt Acknowledge Cycle is in progress. During this cycle, the ESCC interrupt daisy chain is resolved. The device can return an interrupt vector that may be encoded with the type of interrupt pending. During the acknowledge cycle, if IEI is High, the ESCC places the interrupt vector on the data bus when $\overline{\text{RD}}$ goes active for the Z85230/L, or when $\overline{\text{DS}}$ goes active for the Z80230. $\overline{\text{INTACK}}$ is latched by the rising edge of PCLK.

## Pin Descriptions Exclusive to the Z85230/L

The pin description for pins exclusive to Z85230/L is provided below:

**Pins D7–D0 (Data Bus (Bidirectional, tristate))—**These pins carry data and commands to and from the Z85230/L.

**$\overline{\text{CE}}$ (Chip Enable (Input, Active Low))—**This pin selects the Z85230/L for a Read or Write operation.

**$\overline{\text{RD}}$ ((Read (input, Active Low))—**This pin indicates a Read operation and, when the Z85230/L is selected, enables the Z85230/L's bus drivers. During the Interrupt Acknowledge cycle, $\overline{\text{RD}}$ gates the interrupt vector onto the bus if the Z85230/L is the highest priority device requesting an interrupt.

**$\overline{\text{WR}}$ (Write (Input, Active Low))—**When the Z85230/L is selected, this pin denotes a Write operation, which indicates that the CPU writes command bytes or data to the Z85230/L write registers.

**Note:** $\overline{\text{WR}}$ *and* $\overline{\text{RD}}$ *going Low simultaneously is interpreted as a Reset.*

**A/$\overline{\text{B}}$ (Channel A/Channel B (Input))—**This pin selects the channel in which the Read or Write operation occurs. A High selects Channel A and a Low selects Channel B.

**D/$\overline{\text{C}}$ (Data/Control Select (Input))—**This signal defines the type of information transferred to or from the Z85230/L. A High indicates data transfer and a Low indicates a command transfer.

## Pin Descriptions Exclusive to the Z80230

The pin description for pins exclusive to Z80230 is provided below:

**AD7–AD0 (Address/Data Bus (Bidirectional, Active High, tristate))—**These multiplexed lines carry register addresses to the Z80230 as well as data or control information to and from the Z80230.

**R/$\overline{\text{W}}$ (Read/Write (Input, Read Active High))—**This pin specifies if the operation to be performed is a Read or Write operation.

**$\overline{\text{CS0}}$ (Chip Select 0 (Input, Active Low))—**This pin is latched concurrently with the addresses on A7-A0 and must be Low for the intended bus transaction to occur.

**CS1 (Chip Select 1 (Input, Active High))—**This second chip select pin must be High before and during the intended bus transaction.

**DS (Data Strobe (Input, Active Low))—**This pin provides timing for the transfer of data into and out of the Z80230. If $\overline{\text{AS}}$ and $\overline{\text{DS}}$ are both Low, this condition is interpreted as a RESET.

**$\overline{\text{AS}}$ (Address Strobe (Input, Active Low))—**Addresses on A7-A0 are latched by the rising edge of this signal.

# Functional Description

The architecture of the ESCC is described based on its functionality as a:

- Data communications device, which transmits and receives data in a wide variety of protocols
- Microprocessor peripheral, in which the ESCC offers valuable features such as vectored interrupts and DMA support

The details of the communication between the receive and transmit logic of the system bus are displayed in Figure 5 and Figure 6 on page 9. The features and data path for each of the ESCC A and B channels are identical. For more information on SCC/ESCC and ISCC Family of Products, refer to the respective User Manuals available for download from www.zilog.com.

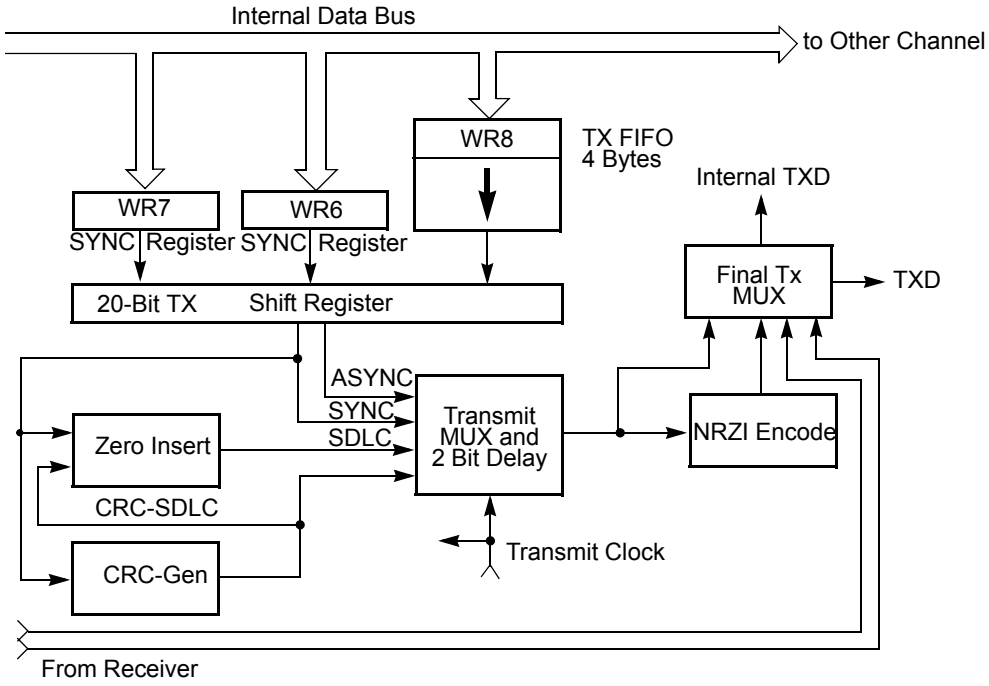


**Figure 5. ESCC Transmit Data Path**

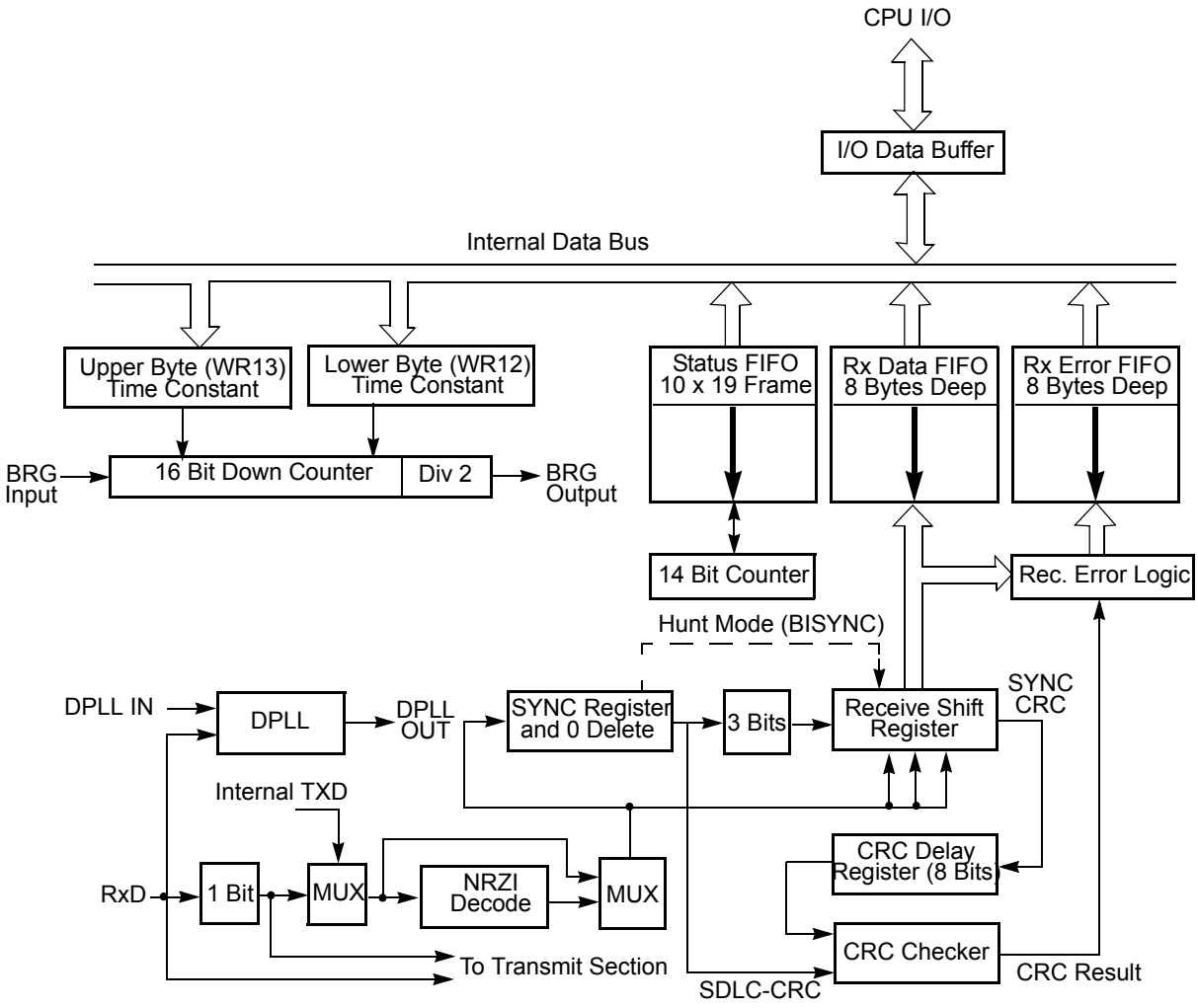


**Figure 6. ESCC Receive Data Path**

## Input/Output Capabilities

System communication to and from the ESCC is accomplished using the ESCC register set. There are 17 Write registers and 16 Read registers. Many of the features on the ESCC are enabled through a new register in the ESCC: Write Register 7 Prime (WR7'). This new register can be accessed if bit 0 or WR15 is set to 1. Table 1 on page 10 lists the Write registers and a brief description of their functions. Table 2 on page 11 lists the Read Registers.

**Note:** *Throughout this document the Write and Read registers are referenced with the notations WR for Write Register and RR for Read Register. For example:*

*WR4A – Write Register 4 for Channel A*

*RR3 – Read Register 3 for either or both channels*

**Table 1. ESCC Write Registers**

| Write Register | Functions |
|---|---|
| WR0 | Command Register; Select Shift Left/Right Mode, Cyclic Redundancy Check (CRC) Initialization, and Resets for Various Modes |
| WR1 | Interrupt Conditions, Wait/DMA Request Control |
| WR2 | Interrupt Vector, Accessed Through Either Channel |
| WR3 | Receive and Miscellaneous Control Parameters |
| WR4 | Transmit and Receive Parameters and Modes |
| WR5 | Transmit Parameters and Controls |
| WR6 | SYNC Character or SDLC Address Field |
| WR7 | SYNC Character or SDLC Flag |
| WR7’ | SDLC Enhancements Enable, Accessible if WR15 bit D0 is 1 |
| WR8 | Transmit FIFO, 4-Bytes Deep |
| WR9 | Reset Commands and Master INT Enable, Accessible Through Either Channel |
| WR10 | Miscellaneous Transmit and Receive Controls |
| WR11 | Clock Mode Control |
| WR12 | Lower Byte of BRG Time Constant |
| WR13 | Upper Byte of BRG Time Constant |
| WR14 | Miscellaneous Controls and Digital Phase-Locked Loop (DPLL) Commands |
| WR15 | External Interrupt Control |

**Table 2. ESCC Read Registers**

| Register Name | Functions |
|---|---|
| RR0 | Transmit, Receive, and External Status |
| RR1 | Special Receive Condition Status Bits |
| RR2A | Unmodified Interrupt Vector |
| RR2B | Modified Interrupt Vector |
| RR3A | Interrupt Pending Bits |
| RR4 | WR4 Mirror, if WR7' bit D6 equals 1 |
| RR5 | WR5 Mirror, if WR7' bit D6 equals 1 |
| RR6 | SDLC Frame LSB Byte Count, if WR15 bit D2 equals 1 |
| RR7 | SDLC Frame 10 X 19 FIFO Status and MSB Byte Count, if WR15 bit DS equals 1 |
| RR8 | Receive Data FIFO, 8 Bits Deep |
| RR9 | WR9 Mirror, if WR7' bit D6 Equals 1 |
| RR10 | Miscellaneous Status Bits |
| RR11 | WR11 Mirror, if WR7' bit D6 Equals 1 |
| RR12 | Lower Byte of BRG Time Constant |
| RR13 | Upper Byte of BRG Time Constant |
| RR14 | WR14 Mirror, if WR7' bit D6 Equals 1 |
| RR15 | WR 15 Mirror, if WR7' bit D6 Equals 1 |

There are three modes used to move data into and out of the ESCC:

1. POLLING
2. INTERRUPT (vectored and non-vectored)
3. BLOCK TRANSFER

The BLOCK TRANSFER mode can be implemented under CPU or DMA control.

### POLLING

When POLLING, data interrupts are disabled, three registers in the ESCC are automatically updated whenever any function is performed. For example, end-of-frame (EOF) in SDLC mode sets a bit in one of these status registers. The purpose of POLLING is for the CPU to periodically read a status register until the register contents indicate the need that data requires transfer. RR0 is the only register that must be read to determine if data needs to be transferred. An alternative to polling RR0 for each channel is to poll the Interrupt

Pending register. Status information for both channels resides in one register. Only one register may be read. Depending on its contents, the CPU performs one of the three operations listed below:

1. Write data
2. Read data
3. Continues processing

Two bits in the register indicate the requirement for data transfer.

## INTERRUPT

The ESCC INTERRUPT mode supports vectored and nested interrupts. The fill levels at which the transmit and receive FIFOs interrupt the CPU are programmable, allowing the ESCC requests for data transfer to be tuned to the system interrupt response time.

Nested interrupts are supported with the interrupt acknowledge ($\overline{\text{INTACK}}$) feature of the ESCC. It allows the CPU to acknowledge the occurrence of an interrupt, and re-enable higher priority interrupts. Since an $\overline{\text{INTACK}}$ cycle releases the $\overline{\text{INT}}$ pin from the active state, a higher priority ESCC interrupt or another higher priority device can interrupt the CPU. When an ESCC responds to $\overline{\text{INTACK}}$ signal from the CPU, it can place an interrupt vector on the data bus. This vector is written in WR2 and may be read in RR2. To increase the interrupt response time, the ESCC can modify 3 bits in this vector to indicate status. If the vector is read in Channel A, status is not included. If it is read in Channel B, status is included.

Each of the six sources of interrupts in the ESCC (Transmit, Receive, and External/Status interrupts in both channels) has 3 bits associated with the interrupt source as listed below:

1. Interrupt Pending (IP)
2. Interrupt Under Service (IUS)
3. Interrupt Enable (IE)

If the IE bit is set for a given interrupt source, then that source can request interrupts. However, when the Master Interrupt Enable (MIE) bit in WR9 is reset, no interrupts can be requested. The IE bits are write-only. The other two bits are related to the interrupt priority chain (see Figure 7 on page 13). The ESCC can request an interrupt only when no higher priority device is requesting an interrupt (that is, when IEI is High). If the device in question requests an interrupt, it pulls down $\overline{\text{INT}}$. The CPU then responds with $\overline{\text{INTACK}}$, and the interrupting device places a vector on the data bus.

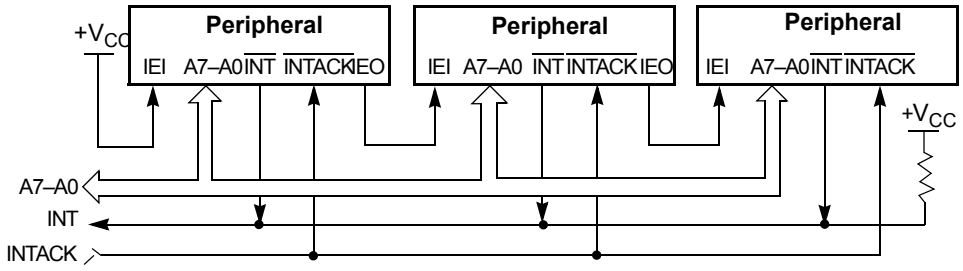


**Figure 7. ESCC Interrupt Priority Schedule**

The ESCC can also execute an Interrupt Acknowledge cycle using software. Sometimes it is difficult to create the $\overline{\text{INTACK}}$ signal with the necessary timing to acknowledge interrupts and allow the nesting of interrupts. In such cases, interrupts can be acknowledged with a software command to the ESCC. For more information, Z80230/Z85230/L Enhancements on page 22

Interrupt Pending (IP) bits signal a need for interrupt servicing. When an IP bit is 1 and the IEI input is High, the $\overline{\text{INT}}$ output is pulled Low, requesting an interrupt. In the ESCC, if an IE bit is not set, then the IP for that source is never set. The IP bits are read in RR3A.

The Interrupt Under Service (IUS) bits signal that an interrupt request is serviced. If IUS is set to 1, all interrupt sources of low priority in the ESCC and external to the ESCC are prevented from requesting interrupts. The internal interrupt sources are inhibited by the state of the internal daisy chain, while lower priority devices are inhibited by setting IEO Low for subsequent peripherals. An IUS bit is set during an Interrupt Acknowledge cycle if there are no higher priority devices requesting interrupt.

There are three type of interrupts as listed below:

1. Transmit
2. Receive
3. External/Status

Each interrupt type is enabled under program control with Channel A having higher priority than Channel B, and with Transmit, Receive, and External/Status interrupts prioritized in that order within each channel. When the Transmit interrupt is enabled (WR1 bit 1 is 1), the occurrence of the interrupt depends on the state of WR7' bit 5. If WR7' bit 5 is 0, the CPU is interrupted when the top byte of the transmit First In First Out (FIFO) becomes empty. If WR7' bit 5 is 1, the CPU is interrupted when the transmit FIFO becomes completely empty. The transmit interrupt occurs when the data in the exit location of the Transmit FIFO loads into the Transmit Shift Register and the Transmit FIFO becomes completely empty. This condition means that there must be at least one character written to the Tx FIFO for it to become empty.

When the receiver is enabled, the CPU is interrupted in one of the following three methods:

1. Interrupt on First Receive Character or Special Receive Condition
2. Interrupt on All Receive Characters or Special Receive Conditions
3. Interrupt on Special Receive Conditions Only

If WR7' bit 3 is 1, and the Special Receive Condition is selected, the Receive character occurs when there are four bytes available in the Receive FIFO. This is most useful in synchronous applications as the data is in consecutive bytes. Interrupt on First Character or Special Condition and Interrupt on Special Condition Only are typically used with the BLOCK TRANSFER mode. A special Receive Condition consists of one of the following:

- Receiver Overrun
- Framing error in ASYNCHRONOUS mode
- EOF in SDLC mode
- Parity error (optional)

The Special Receive Condition interrupt is different from an ordinary receive character available interrupt only by the status placed in the vector during the Interrupt Acknowledge cycle. In Receive Interrupt on First Character or Special Condition mode, an interrupt occurs from Special Receive Conditions  any time after the first receive character interrupt.

The primary function of the External/Status interrupt is to monitor the signal transitions of the $\overline{\text{CTS}}$, $\overline{\text{DCD}}$, and $\overline{\text{SYNC}}$ pins. However, an External/Status interrupt is also caused by any of the following:

- A Transmit Underrun condition
- A zero count in the BRG
- A detection of a Break (ASYNCHRONOUS mode)
- An ABORT (SDLC mode)
- An End Of Poll (EOP) sequence in the data stream (SDLC LOOP mode)

The interrupt caused by the ABORT or EOP sequence has a special feature that allows the ESCC to interrupt when the ABORT or EOP sequence is detected or terminated. This feature facilitates the proper termination of the current message, correct initialization of the next message, and the accurate timing of the ABORT condition by external logic in SDLC mode. SDLC LOOP mode allows secondary stations to recognize the primary station and regain control of the loop during a poll sequence.

### CPU/DMA BLOCK TRANSFER

The ESCC provides a BLOCK TRANSFER mode to accommodate CPU/DMA controller. The BLOCK TRANSFER mode uses the $\overline{\text{WAIT}}/\overline{\text{REQUEST}}$ output in conjunction with the $\overline{\text{WAIT}}/\overline{\text{REQUEST}}$ bits in WR1. The $\overline{\text{WAIT}}/\overline{\text{REQUEST}}$ output can be defined as a $\overline{\text{WAIT}}$ line in the CPU BLOCK TRANSFER mode or as a $\overline{\text{REQUEST}}$ line in the DMA BLOCK TRANSFER mode.

To a DMA controller, the ESCC $\overline{\text{REQUEST}}$ output indicates that the ESCC is ready to transfer data to or from memory.

To the CPU, the $\overline{\text{WAIT}}$ line indicates that the ESCC is not ready to transfer data, thereby requesting the CPU to extend the I/O cycle.

The $\overline{\text{DTR}}/\overline{\text{REQUEST}}$ line allows full-duplex operation under DMA control. The ESCC can be programmed to deassert the $\overline{\text{DTR}}/\overline{\text{REQUEST}}$ pin with the same timing as the $\overline{\text{WAIT}}/\overline{\text{REQUEST}}$ pin if WR7' bit 4 is 1.

## ESCC Data Communications Capabilities

The ESCC provides two independent full-duplex programmable channels for use in any common ASYNCHRONOUS or SYNCHRONOUS data communication protocols (see Figure 8). The channels have identical features and capabilities.

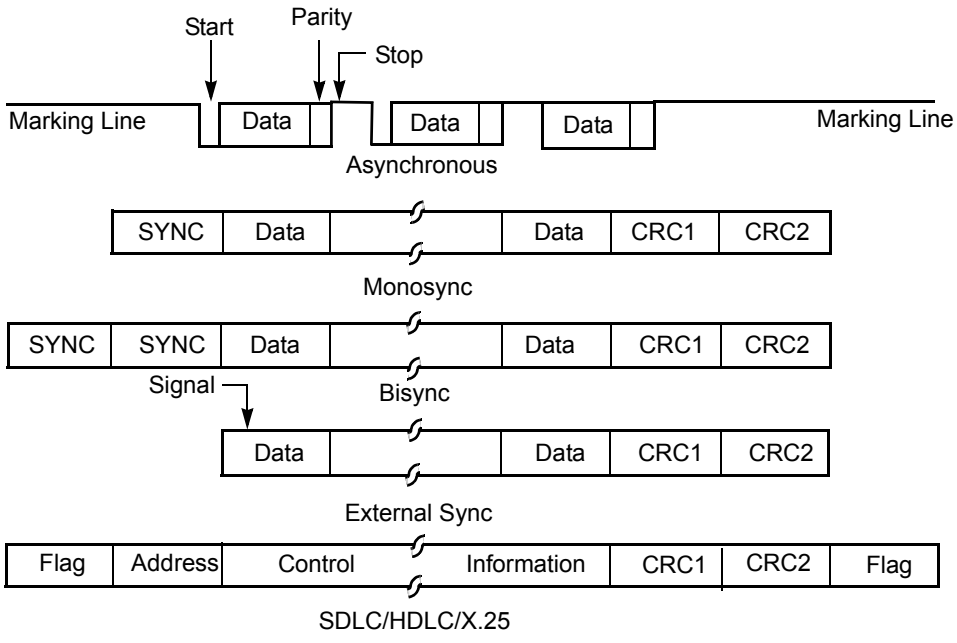


**Figure 8. Various ESCC Protocols**

### ASYNCHRONOUS Mode

The ESCC has significant improvements over the standard Serial Communications Controller (SCC). The addition of the deeper data FIFOs provide greater protection against underruns and overruns as well as more efficient use of bus bandwidth. The deeper data FIFOs are accessible regardless of the protocol used and they need not be enabled. For information on these improvements, see Z80230/Z85230/L Enhancements on page 22

Send and Receive allow 5 to 8 bits per character, plus optional Even or Odd parity. The transmitters can supply 1, 1.5, or 2 stop bits per character and can provide break indication. The receiver break-detection logic interrupts the CPU both at the start and at the end of a received break. Reception is protected from spikes by start-bit validation that delays the signal for a length of time equal to one half the time period required to process 1 bit of data after a Low level is detected on the receive data input (RxDA or RxDB pins). If the Low level does not persist (that is, a transient), the character assembly process does not start.

Framing errors and overrun errors are detected and buffered together with the character at which they occur. Vectored interrupts allow fast servicing of error conditions. Furthermore, a built-in checking process avoids the interpretation of a framing error as a new start bit. A framing error results in the addition of a delay of one half the amount of time required to process 1 bit of data at the point at which the search for the next start bit begins. Transmit and Receive clock can be selected from any of the several sources. In ASYNCHRONOUS mode, the $\overline{\text{SYNC}}$ pin may be programmed as an input with interrupt capability.

### SYNCHRONOUS Mode

The ESCC supports both byte-oriented and bit-oriented SYNCHRONOUS communication. SYNCHRONOUS byte-oriented protocols are handled in several modes. They enable character synchronization with a 6- or 8-bit SYNC character (MONOSYNC) or a 12-bit or 16-bit synchronization pattern (BISYNC), or with an external sync signal. Leading sync characters are removed without interrupting the CPU.

5- or 7-bit sync characters are detected from 8- or 16-bit patterns in the ESCC by overlapping the larger pattern across multiple incoming sync characters as displayed in Figure 9.

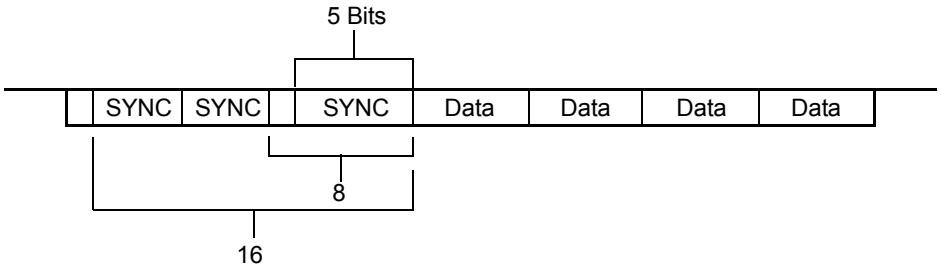


**Figure 9. Detecting 5- or 7-Bit Synchronous Characters**

CRC checking for SYNCHRONOUS BYTE-ORIENTED mode is delayed by one character time so that the CPU may disable CRC checking on specific characters. This action permits the implementation of protocols such as IBM BISYNC.

Both CRC-16 ($X^{16} + X^{15} + X^2 + 1$) and CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) error checking polynomials are supported. Either polynomial may be selected in all synchronous modes. You can preset the CRC generator and checker to all 1s or all 0s. The ESCC also provides a feature that automatically transmits CRC data when no other data is available for transmission. This feature enables high-speed transmissions under DMA control, with no need for CPU intervention at the end of a message. When there is no data or CRC to send in the SYNCHRONOUS mode, the transmitter inserts 6-, 8-, 12-, or 16-bit SYNC characters, regardless of the programmed character length.

### SDLC Mode

The ESCC supports SYNCHRONOUS bit-oriented protocols, such as SDLC and High-Level Data Link Control (HDLC), by performing automatic flag sending, zero insertion, and CRC generation.

A special command is used to abort a frame which is in transmission. At the end of a message, the ESCC automatically transmits the CRC and trailing flag when the transmitter underruns. The transmitter may also be programmed to send an idle line consisting of continuous flag characters or a steady marking condition.

If a transmit underrun occurs in the middle of a message, an External/Status interrupt warns the CPU of this status change so that an `Abort` command can be issued. The ESCC may also be programmed to send an `Abort` command by itself, in the event of an underrun, relieving the CPU of the task. The last character of a frame may consist of 1- to 8-bits, allowing reception of frames of any length.

The receiver automatically synchronizes on the leading flag of a frame in SDLC or HDLC and provides a synchronization signal on the $\overline{\text{SYNC}}$ pin (an interrupt may also be programmed). The receiver may search for frames addressed by 1-byte or 4-bits within a byte of a user-specified address or for a global broadcast address. Frames not matching either the user-selected address or broadcast address are ignored.

The number of address bytes are extended under software control. To receive data, an interrupt can be selected on the first received character, or on every character, or On Special Condition Only (EOF). The receiver automatically deletes all zeros inserted by the transmitter during character assembly. CRC is also calculated and is automatically checked to validate frame transmission. At the end of transmission, the status of a received frame is available in the status registers. In SDLC mode, the ESCC must be programmed to use the CRC-CCITT polynomial, but the generator and checker may be pre-set to all 1s or all 0s. The CRC data is inverted before transmission and the receiver checks against the bit pattern `0001110100001111`.

NRZ, NRZI, or FM coding may be used in any 1X mode. The parity options available in ASYNCHRONOUS mode are also available in SYNCHRONOUS mode. However, parity checking is not normally used for SDLC because CRC checking is more robust.

### SDLC LOOP Mode

The ESCC supports SDLC LOOP mode as well as normal SDLC. In SDLC LOOP mode, a primary controller station manages the message traffic flow on the loop and any number of secondary stations. In SDLC LOOP mode, the ESCC performs the functions of a secondary station. An ESCC operation in regular SDLC mode may act as a controller (see Figure 10). SDLC LOOP mode is selected by setting WR10 bit 1 to 1.

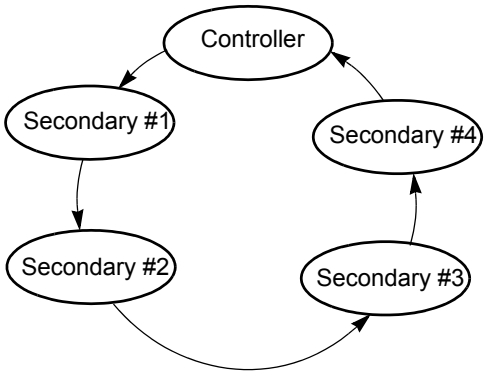


**Figure 10. SDLC LOOP mode**

A secondary station in an SDLC LOOP mode always monitors the messages sent around the loop and passes these messages to the rest of the loop, retransmitting them with a one-bit time delay. The secondary station places its own message in the loop only at specific times. The controller indicates that the secondary stations can transmit messages by sending a special character, called EOP, around the loop. The EOP character has a bit pattern `11111110`, the same pattern as an `Abort` character in normal HDLC. This bit pattern is unique and easily recognized, because of the zero insertion in the message.

When a secondary station has a message to transmit and recognizes an EOP on the line, it changes the last binary 1 of the EOP to a 0 before transmission. This action changes the EOP into a flag sequence. The secondary station now places its message on the loop and terminates the message with an EOP. Any secondary stations further down the loop with messages to transmit appends their messages to the message of the first secondary station using the same process. Secondary stations without any messages to transmit merely echo the incoming message. All secondary stations are prohibited from placing messages on the loop except upon recognizing an EOP. In SDLC LOOP mode, NRZ, NRZI or FM coding can be used.

### SDLC Status FIFO

The ESCC's ability to receive high speed back-to-back SDLC frames is maximized by a 10-bit deep by 19-bit wide status FIFO buffer. When enabled (through WR15 bit 2 is 1), the storage area enables DMA to continue data transfer into the memory, so that the CPU examines the message later. For each SDLC frame, 14 counter bits and 5 Status/Error bits are stored. The byte count and status bits are accessed through Read Registers, RR6, and RR7. RR6 and RR7 are only used when the SDLC FIFO buffer is enabled. The 10 x 19 status FIFO buffer is separate from the 8-byte receive data FIFO buffer.

### Baud Rate Generator

Each channel in the ESCC contains a programmable BRG. Each generator consists of two 8-bit registers that form a 16-bit time constant, a 16-bit down counter, and a flip-flop on the output, producing a square wave. At start-up, the flip-flop at the output is set High, the value in the time constant register is loaded into the counter, and the count down begins. When the BRG reaches zero, the output toggles, the counter is reloaded with the time constant, and the process repeats. The time constant can be changed at any time, but the new value does not take effect until the counter is loaded again.

The output of the BRG may be used as the Transmit clock, the Receive clock, or both. The output can also drive the DPLL. For more information, see Digital Phase-Locked Loop.

If the receive clock or the transmit clock is not programmed to come from the TRxC pin, the output of the BRG may be echoed out by the TRxC pin.

The following formula relates the time constant to the baud rate. PCLK or $\overline{\text{RTxC}}$ is the clock input to the BRG. The clock mode is 1, 16, 32, or 64, as selected in WR 4 bits 6 and 7.

$$\text{Time Constant} = \frac{\text{PCLK or RTxC Frequency}}{2(\text{Baud Rate})(\text{Clock Mode})} - 2$$

### Digital Phase-Locked Loop

The ESCC contains a DPLL to recover clock information from a data stream with NRZI or FM encoding. The DPLL is driven by a clock that is nominally 32 (NRZI) or 16 (FM) times the data rate. The DPLL uses this clock, along with the data stream, to construct a clock for the data. This clock is then used as the ESCC receive clock, the transmit clock, or both. When the DPLL is selected as the transmit clock source, it provides a jitter-free clock output. The clock output is the DPLL input frequency divided by the appropriate divisor for the selected encoding technique.

For NRZI encoding, the DPLL counts the 32x clock to create nominal bit times. As the 32x clock is counted, the DPLL searches the incoming data stream for edges (either 1 to 0 or 0 to 1). When a transition is detected the DPLL makes a count adjustment (during the next counting cycle), producing a terminal count closer to the center of the bit cell.

For FM encoding, the DPLL counts from 0 to 32, but with a cycle corresponding to two bit times. When the DPLL is locked, the clock edges in the data stream occurs between counts 15 and 16 and between counts 31 and 0. The DPLL looks for edges only during a time centered on the 15 to 16 counting transition.

The 32x clock for the DPLL can be programmed to come from either the RTxC input or the output of the BRG. The DPLL output is programmed to be echoed out the ESCC by the TRxC pin (if this pin is not being used as an input).

### Data Encoding

Data encoding allows the transmission of clock and data information over the same medium. This capability saves the need to transmit clock and data over separate medium as is normally required tor synchronous data. The ESCC provides four different data encoding methods, selected by bits 6 and 5 in WR10. Examples of these 4 encoding methods is displayed in Figure 11. Any encoding method is used in any X1 mode in the ESCC, ASYNCHRONOUS or SYNCHRONOUS. The data encoding selected is active even if the transmitter or receiver is idling or disabled.

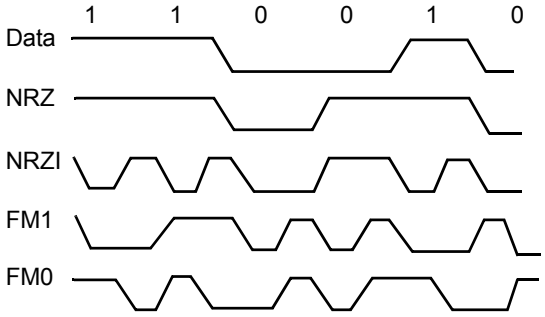


**Figure 11. Data Encoding Methods**

Table 3 lists the four encoding methods, their levels, and values.

**Table 3. Data Encoding Descriptions**

| Code Type | Level | Value |
|---|---|---|
| NRZ | High<br>Low | 1<br>0 |
| NRZI | No Change<br>Change | 1<br>0 |