

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Ribbon / Performance Technologies / Ziatech ZT-88CT41* at our website: *Click HERE*

ZT 8841 and ZT 88CT41
Quad Serial Interface

HARDWARE MANUAL

For Revision A.2

Reorder Part Number ZT M8841
May 20, 1994

ZIATECH CORPORATION
1050 Southwood Drive
San Luis Obispo, CA 93401 USA
FAX (805) 541-5088
Telephone (805) 541-0488

ZIATECH WARRANTY

Warranty information for Ziatech products is available at Ziatech's website, located at http://www.ziatech.com.

CUSTOMER SUPPORT

If you have a technical question, please call Ziatech's Customer Support Service at one of the following numbers:

Corporate Headquarters: (805) 541-0488
(805) 541-5088 (FAX)

If you have a sales question, please contact your local Ziatech Sales Representative.

Copyright 1993 Ziatech Corporation

Portions copyright Intel Corporation. Used by permission.

IBM PC/XT/AT and MS-DOS are registered trademarks of International Business Machines, Inc.

MCS-51 is a registered trademark of Intel Corporation.

PREFACE

This manual describes the operation and uses of the ZT 8841 Quad Serial Interface. This manual has been organized to allow you to begin reading at many different points. The focus of each chapter is outlined below to help you find the best place to start. Section headings enclosed in boxes indicate the location of a labeled tab for quick access to the appropriate information.

I. INTRODUCTION

Chapter 1, "Introduction," offers an overview of the ZT 8841. It includes a product definition, a list of product features, a functional block diagram, and a brief description of each block. If you are evaluating the ZT 8841 to determine if it fits your needs, this chapter will be especially useful to you.

II. GETTING STARTED

Chapter 2, "Getting Started," summarizes the information essential to getting your ZT 8841 up and running. You can refer to the chapters in the USER'S REFERENCE section of this manual for further explanation of the material covered here.

III. USER'S REFERENCE

Chapter 3, "Theory Of Operation," presents a detailed description of ZT 8841 system level operation. Topics discussed include commonly asked questions, asynchronous communications, interrupt operation, the STD interface, and the serial interface.

Chapter 4, "Application Example," gives a specific example of the ZT 8841 in operation, including assembly language code to initialize, send, and receive characters using the ZT 8841.

Chapter 5, "Interrupt Controller (8259A)," explains the operation and programming of the interrupt controller and discusses 8259A I/O port addresses and interrupts.

Chapter 6, "Serial Communications (82050)," describes the function, configuration, and operation of the Intel 82050 Asynchronous Communication Controller.

IV. APPENDICES

Appendix A, "Jumper Configurations," summarizes the jumper-selectable options that tailor ZT 8841 operation.

Appendix B, "Specifications," contains the electrical, mechanical, and environmental specifications for the ZT 8841.

Appendix C, "Quick Reference," supplies a centralized location for frequently referenced tables and illustrations.

Appendix D, "Customer Support," offers a product revision history, technical assistance, and the necessary information should you need to return your ZT 8841 for repair.

Appendix E, "Glossary," lists definitions for important terms and acronyms used throughout the manual.

CONTENTS

I. INTRODUCTION

Chapter 1. INTRODUCTION 1-1

OVERVIEW 1-1
Product Definition 1-2
Features Of The ZT 8841 1-3
FUNCTIONAL BLOCKS 1-4
Serial Communications 1-4
Enhanced Serial Communications 1-8
Interrupts 1-9

II. GETTING STARTED

Chapter 2. GETTING STARTED 2-1

OVERVIEW 2-2
UNPACKING 2-2
WHAT'S IN THE BOX 2-3
SYSTEM REQUIREMENTS 2-4
PROCESSOR COMPATIBILITY 2-5
I/O ADDRESSING 2-5
I/O Port Selection 2-8
ZT 8801 COM I/O Addresses 2-11
ZT 8806/8807 COM I/O Addresses 2-12
ZT 8808/8809 Rev B or Later COM I/O Addresses 2-13
ZT 8816/8817 Rev B or Later COM I/O Addresses 2-14
ZT 8901 COM I/O Addresses 2-15
ZT 8902 COM I/O Addresses 2-16
ZT 8910 COM I/O Addresses 2-17

ZT 8911 COM I/O Addresses 2-18
DEFAULT CONFIGURATION 2-19
Factory Defaults and Initial Conditions 2-19
JUMPER DESCRIPTIONS 2-21
PIN ASSIGNMENTS 2-22
FURTHER INFORMATION 2-26

III. USER'S REFERENCE

Chapter 3. THEORY OF OPERATION 3-1
OVERVIEW 3-2
COMMONLY ASKED QUESTIONS 3-2
ASYNCHRONOUS COMMUNICATIONS 3-5
Standards 3-5
Configurations 3-6
INTERRUPT OPERATION 3-8
Fundamentals 3-8
Configurations 3-11
FUNCTIONAL OPERATION 3-15
STD Interface 3-16
Serial Interface 3-17

Chapter 4. APPLICATION EXAMPLE 4-1
ZT 8841 SOFTWARE DRIVER 4-1
Objectives 4-1
Program Code 4-2

Chapter 5. INTERRUPT CONTROLLER (8259A) 5-1
INTRODUCTION TO THE INTERRUPT CONTROLLER 5-2
Interrupt Controller Operation 5-2
Interrupt Registers and Control Logic 5-6
OPERATION OF THE INTERRUPT CONTROLLER 5-8
Priorities 5-8
Interrupt Triggering 5-11
Interrupt Status 5-14
PROGRAMMING THE INTERRUPT CONTROLLER 5-16
End-Of-Interrupt (EOI) Formats 5-16
Programming With ICWs and OCWs 5-18
Initialization Command Words (ICWs) 5-19

Operation Control Words (OCWs) 5-24

Chapter 6. SERIAL COMMUNICATIONS (82050) 6-1

OVERVIEW 6-2
ZT 8841 SPECIFICS 6-3
FUNCTIONAL DESCRIPTION 6-5
Read/Write Control 6-5
Receiver 6-5
Transmitter 6-5
Modem Control 6-6
Interrupt Control 6-6
PROGRAMMABLE REGISTERS 6-7
Transmit and Receive Buffer 6-8
Line Control 6-8
Line Status 6-11
Modem Control 6-13
Modem Status 6-16
Divisor Latch 6-18
Interrupt Identify 6-19
Interrupt Enable 6-20
OPERATION 6-21
Reset 6-21
Serial Data Format 6-22
Baud Rate 6-23
Interrupt and Polled Communication 6-25
RS-422/485 Serial Communications 6-25
Programming 6-26

IV. APPENDICES

Appendix A. JUMPER CONFIGURATIONS A-1

OVERVIEW A-1
JUMPER DESCRIPTIONS A-5

Appendix B. SPECIFICATIONS B-1

INTRODUCTION B-1
ELECTRICAL SPECIFICATIONS B-2
Absolute Maximum Ratings B-2
DC Operating Characteristics B-2

Contents

STD Load Characteristics B-4
ENVIRONMENTAL SPECIFICATIONS B-7
Operating Temperature B-7
Storage Temperature B-7
MECHANICAL SPECIFICATIONS B-8
Connectors B-8
Cable Specifications B-12
Card Dimensions & Weight B-15

Appendix C. QUICK REFERENCE C-1

OVERVIEW C-1
I/O MAP C-1
J1-J4 SERIAL PIN ASSIGNMENTS C-5
J5 INTERRUPT PIN ASSIGNMENTS C-7

Appendix D. CUSTOMER SUPPORT D-1

OVERVIEW D-1
ZT 8841 REVISIONS D-2
Revision History D-2
ZT 8841 Revision 0.1 . . . . . D-2
ZT 88CT41 Revision A — 4/23/91 . . . . . D-2
ZT 88CT41 Revision A.1 — 8/7/91 . . . . . D-3
ZT 88CT41 Revision A.2 — 12/11/91 . . . . . D-3
RELIABILITY D-4
TECHNICAL ASSISTANCE D-6
RETURNING THE ZT 8841 FOR SERVICE D-7

Appendix E. GLOSSARY E-1

Index i

TABLES

Table 1–1	ZT 8841 RS-232-C Signals Implemented.	1-6
Table 1–2	ZT 8841 RS-422/485 Signals Implemented.	1-7
Table 2–1	I/O Address Jumper Selection.	2-6
Table 2–2	Decimal/Hexadecimal/Binary Conversion.	2-7
Table 2–3	COM Address Configurations.	2-8
Table 2–4	I/O Registers.	2-9
Table 2–5	ZT 8801 COM Configuration.	2-11
Table 2–6	ZT 8806/8807 COM Configuration.	2-12
Table 2–7	ZT 8808/8809 COM Configuration.	2-13
Table 2–8	ZT 8816/8817 COM Configuration.	2-14
Table 2–9	ZT 8901 COM Configuration.	2-15
Table 2–10	ZT 8902 COM Configuration.	2-16
Table 2–11	ZT 8910 COM Configuration.	2-17
Table 2–12	ZT 8911 COM Configuration.	2-18
Table 2–13	Default COM Address Configuration.	2-21
Table 5–1	8088 or 80X86 Interrupt Vector.	5-3
Table 6–1	ACC Register Addressing.	6-7
Table 6–2	Serial Character Length.	6-8
Table 6–3	ACC Register Defaults.	6-21
Table 6–4	ACC Baud Rate Divisors.	6-24
Table 6–5	ACC Register Summary.	6-27
Table A–1	Jumper Cross Reference.	A-2
Table A–2	ZT 8841 Jumper Descriptions.	A-5
Table B–1	STD Bus Loading, P Connector.	B-5
Table B–2	STD Bus Loading, E Connector.	B-6
Table B–3	J1-J4 RS-232-C Pin Assignments.	B-10
Table B–4	J1-J4 RS-422/485 Pin Assignments.	B-11
Table B–5	J5 Pin Assignments.	B-12
Table C–1	Base Address Jumpers.	C-2
Table C–2	I/O Registers.	C-4
Table C–3	J1-J4 RS-232-C Pin Assignments.	C-5
Table C–4	J2-J3 RS-422/485 Pin Assignments.	C-6
Table C–5	J5 Interrupt Pin Assignments.	C-7

Tables

ILLUSTRATIONS

Figure 1–1	ZT 8841 Functional Block Diagram.	1-5
Figure 2–1	ZT 8841 Factory Default Configuration.	2-20
Figure 2–2	RS-232-C DTE and ZT 90027 Diagram.	2-23
Figure 2–3	RS-232-C DCE and ZT 90014 Diagram.	2-24
Figure 2–4	RS-422/485 and Cable Diagram.	2-25
Figure 3–1	Interrupt For Service.	3-8
Figure 3–2	Master Interrupt Sequence.	3-9
Figure 3–3	Master/Slave Interrupt Sequence.	3-10
Figure 3–4	Shared STD Bus Interrupt Structure.	3-12
Figure 3–5	Dedicated STD Bus Interrupt Structure.	3-13
Figure 3–6	Dedicated Frontplane Interrupt Structure.	3-14
Figure 3–7	ZT 8841 Functional Diagram.	3-15
Figure 3–8	COM1, COM4 Interrupt Enable Circuit.	3-19
Figure 3–9	COM2, COM3 Interrupt Enable Circuit.	3-19
Figure 3–10	COM1-4 Interrupt Structure.	3-20
Figure 3–11	COM2, COM3 RS-485 Enable Circuit.	3-24
Figure 5–1	8259A Block Diagram.	5-5
Figure 5–2	8259A Initialization Command Word Formats.	5-20
Figure 5–3	8259A Operation Control Word Formats.	5-25
Figure 6–1	ACC Block Diagram.	6-4
Figure 6–2	Line Control Register.	6-9
Figure 6–3	Line Status Register.	6-11
Figure 6–4	Modem Control Register.	6-14
Figure 6–5	Modem Status Register.	6-16
Figure 6–6	Divisor Latch.	6-18
Figure 6–7	Interrupt Identify Register.	6-19
Figure 6–8	Interrupt Enable Register.	6-20
Figure 6–9	ACC Serial Data Format.	6-22
Figure A–1	ZT 8841 Default Jumper Configuration.	A-3
Figure A–2	ZT 8841 Customer Jumper Configuration.	A-4
Figure A–3	COM1-4 RS-232-C DTE (Default).	A-11
Figure A–4	COM2-3 RS-422/485 Operation.	A-12
Figure A–5	COM1-COM4 RS-232-C DCE Operation.	A-18

Illustrations

Figure A–6 Active High IRQx. A-21
Figure A–7 Active Low IRQx. A-21
Figure B–1 P/E Connector Pinout. B-9
Figure B–2 ZT 90014 Rev. 0 Cable Drawing. B-13
Figure B–3 ZT 90027 Rev. A Cable Drawing. B-14
Figure B–4 ZT 8841 Card Dimensions. B-15

Chapter 1

INTRODUCTION

Contents **Page**

OVERVIEW 1-1
Product Definition 1-2
Serial Communications 1-2
Processor Compatibility 1-2
CMOS Technology 1-3
Software Support 1-3
Features Of The ZT 8841 1-3
FUNCTIONAL BLOCKS 1-4
Serial Communications 1-4
RS-232-C Interface 1-6
RS-422/485 Interface 1-6
RS-422/485 Optical Isolation Interface Option 1-7
Enhanced Serial Communications 1-8
Interrupts 1-9

OVERVIEW

This chapter briefly introduces some key features of the ZT 8841. It includes a product definition, a list of product features, a functional block diagram, and a description of each block. Review this chapter carefully to gain a basic understanding of the capabilities of the ZT 8841. Unpacking information and installation instructions can be found in Chapter 2, "Getting Started."

Unless specifically noted, all references to the ZT 8841 also refer to the CMOS version ZT 88CT41.

Product Definition

The ZT 8841 is a four-channel serial interface (COM1-4) for the STD-80 bus. This interface features four 82050 UARTs (8250 compatible) that operate at speeds of up to 56 Kbaud in DTE or DCE configurations. Each serial communications (COM) channel features directly addressable UART registers, individual baud rate generators, and direct hardware interrupt capability.

Serial Communications

Each of the four serial channels (COM1-4) is configured for RS-232-C operation. Two of these channels (COM2,3) can be reconfigured for RS-422/485 operation. Interrupts generated by the transmitter or receiver buffers are arbitrated by an on-board 8259A Programmable Interrupt Controller (PIC). The 8259A provides unique interrupt vectors for the STD CPU if enabled.

Processor Compatibility

The ZT 8841 is compatible with the following Ziatech STD bus microcomputers:

- ZT 8801
- ZT 8806/8807
- ZT 8808/8809
- ZT 8816/8817
- ZT 8901
- ZT 8910

It provides IBM address and interrupt compatibility without overlapping serial channels on the processor board. Provisions have also been made for moving one or two ZT 8841 serial channels to another I/O address if occupied by a modem or a CPU COM channel.

CMOS Technology

In addition to these features, the ZT 88CT41 incorporates low-power CMOS devices and operation in -40 to +85 degrees Celsius environments. CMOS devices that are TTL compatible ensure system compatibility with other CMOS or TTL STD bus boards.

Software Support

The ZT 8841 is supported by Ziatech's Device Driver Package (DDP). This package allows you to easily integrate the ZT 8841 into your own program written in Assembly language, C, or Pascal. Library routines to initialize, test, reset, write a bit, and read a bit are included.

Features Of The ZT 8841

- Compatible with 5- and 8-MHz STD 32 and STD-80 Series systems
- Compatible with 8- or 16-bit I/O addressing
- Four RS-232-C channels (COM1-4) with two channels configurable for RS-422/485 (COM2,3)
- On-board interrupt controller
- Loopback diagnostic capability
- Ziatech Device Driver Package (DDP) support
- Optional optical isolation on RS-422/485 channels
- Optional four-byte transmit and receive FIFO for each channel
- Five-year warranty

FUNCTIONAL BLOCKS

Figure 1-1 on page 1-5 illustrates the functional blocks of the ZT 8841. A description of each block follows.

Serial Communications

The ZT 8841 includes four asynchronous serial communication (COM) channels, each with fully programmable interface characteristics. Programmable functions include:

- 5-, 6-, 7-, or 8-bit characters
- even or odd, or no parity bit generation/detection
- 1, 1½, or 2 stop-bit generation

Programmable diagnostics for each UART allow tests such as loopback, send, and receive verification.

Each serial channel (COM1-4) has its own unique 14-pin latching male connector (J1-J4). This arrangement allows easy connection to an optional 25-pin D-type connector on a flat ribbon cable without conductor scrambling or soldering. Two cables are available from Ziatech:

1. ZT 90014 cable for DCE operation - one meter long, terminated with a female D-type connector.
2. ZT 90027 cable for DTE operation (DOS compatible) - one meter long, terminated with a male D-type connector.

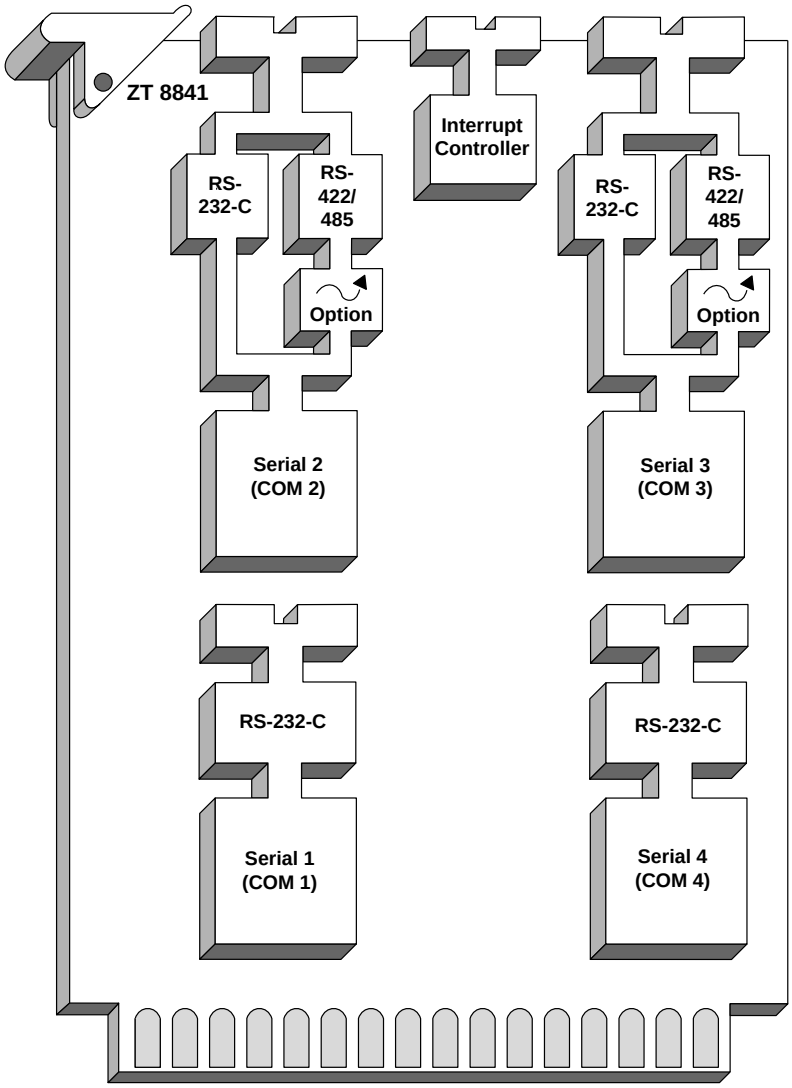


Figure 1–1. ZT 8841 Functional Block Diagram.

RS-232-C Interface

The RS-232-C interface uses drivers and receivers that feature slew rate control to minimize crosstalk on adjacent signals. Hysteresis on inputs helps reduce the effects of noise. These drivers and receivers meet the EIA Standard RS-232-C and CCITT V.28 specifications.

Each channel is shipped in DTE configuration and can be jumpered for DCE operation by rotating the jumpers 90 degrees.

Table 1-1 indicates the RS-232-C signals available for each of the four serial channels on the ZT 8841.

Table 1-1
ZT 8841 RS-232-C Signals Implemented.

Name	Description
TXD, RXD	Transmit and receive data
RTS, CTS	Request and clear to send
DTR, DSR	Data terminal and set ready
RI	Ring indicator
DCD	Data carrier detect

RS-422/485 Interface

The RS-422/485 interface uses differential drivers for greater noise immunity, higher speed, and longer distance operation. Multidrop operation is also possible because the transmitter outputs can be enabled and disabled with software. Software control is useful if peer-to-peer or master/slave(s) communication is desired. COM2 and COM3 are the only serial channels that can be jumpered for RS-232-C or RS-422/485 operation.

Table 1-2 shows the RS-422/485 signals available on ZT 8841 serial channels COM2 and COM3.

Table 1-2
ZT 8841 RS-422/485 Signals Implemented.

Name	Description
TXD, RXD	Transmit and receive data
RTS, CTS	Request and clear to send

RS-422/485 Optical Isolation Interface Option

You can choose to optically isolate the ZT 8841's UARTs from noise and voltage spikes up to 600 V on both RS-422/485 channels. Environments that present this type of excessive noise could cause damage to the UARTs or the computer without the optical isolation option.

Note that ZT 8841 boards with the optical isolation option must utilize both COM2 and COM3 for RS-422/485 in order to retain their optical isolation. Using COM2 or COM3 for RS-232-C with optical isolation on the remaining RS-422/485 channel defeats all the benefits of optical isolation.

Enhanced Serial Communications

Optionally, the ZT 8841 can be purchased with high performance 82510 UARTs that are 8250 compatible. These UARTs increase system efficiency by providing the following features:

- DC to 288 Kbaud
- complete error detection
- separate four-byte receive and transmit FIFOs
- control character recognition
- two independent 16-bit baud rate generators/timers

Enhanced features are configured by 35 control registers while maintaining emulation of the 8250 UARTs used in the IBM PC.

The enhanced serial communications option also includes MCS-51 support. The MCS-51 protocol developed by Intel can be used for multiprocessor communications in a master/slave(s) configuration. See page 2-26 and Intel's MCS-51 User's Manual for more information.

When the master processor wants to transmit a block of data to one of several slaves, it first sends out the address byte of the target slave. An address byte differs from a data byte in that the ninth bit is **1 in an address byte** and **0 in a data byte**. Each UART is initialized not to interrupt on a data byte. An address byte, however, interrupts all slaves so that each slave can examine the receive byte to see if it is being addressed. The addressed slave clears its interrupt and prepares to receive subsequent data bytes. Slaves not being addressed ignore all data sent to the addressed slave.

Interrupts

The ZT 8841 includes one 8259A Programmable Interrupt Controller (PIC). The 8259A can generate a vectored interrupt to the CPU when a character is transmitted or received or when an error occurs. When operating as a slave interrupt controller to a master on the CPU board, the 8259A cascade address is supplied via STD bus address lines A8-A10 as defined in the *STD-80 Series Bus Specification.* This feature will save the CPU from polling the UART every time a character is sent or received or an error occurs.

Interrupts from the ZT 8841 PIC or from the individual UARTs (bypassing the PIC) can be routed over the frontplane or backplane. Frontplane interrupts are jumper selectable on the 10-pin header consistently used on Ziatech CPU and I/O boards. Backplane interrupts are jumper selectable on STD signals INTRQ*, INTRQ1*, or CNTL (INTRQ2*).

Chapter 2

GETTING STARTED

Contents **Page**

OVERVIEW 2-2
UNPACKING 2-2
WHAT'S IN THE BOX 2-3
SYSTEM REQUIREMENTS 2-4
PROCESSOR COMPATIBILITY 2-5
I/O ADDRESSING 2-5
I/O Port Selection 2-8
ZT 8801 COM I/O Addresses 2-11
ZT 8806/8807 COM I/O Addresses 2-12
ZT 8808/8809 Rev B or Later COM I/O Addresses 2-13
ZT 8816/8817 Rev B or Later COM I/O Addresses 2-14
ZT 8901 COM I/O Addresses 2-15
ZT 8902 COM I/O Addresses 2-16
ZT 8910 COM I/O Addresses 2-17
ZT 8911 COM I/O Addresses 2-18
DEFAULT CONFIGURATION 2-19
Factory Defaults and Initial Conditions 2-19
JUMPER DESCRIPTIONS 2-21
PIN ASSIGNMENTS 2-22
FURTHER INFORMATION 2-26

OVERVIEW

This chapter summarizes the information essential to getting your ZT 8841 up and running. You should read this chapter before attempting to use the board.

Many topics in "Getting Started" are discussed in greater detail in subsequent chapters. You will find a number of references for pages containing further information.

UNPACKING

Please check the shipping carton for damage. If the shipping carton and contents are damaged, notify the carrier and Ziatech for an insurance settlement. Retain the shipping carton and packaging material for inspection by the carrier. Do not return any product to Ziatech without a Return Material Authorization (RMA) number. Appendix D explains the procedure for obtaining an RMA number from Ziatech (see page D-7).

WHAT'S IN THE BOX

When the package is opened, you should find:

1. Optional ZT 8841 Hardware Manual, in binder
2. One ZT 8841 Quad UART board
3. Anti-static packing material
4. Optional ZT 90014 or ZT 90027 serial cable(s)

Attach the sticker included with this manual onto the spine of the binder for easy identification. Insert the tabbed pages in the manual for quick reference. Save the anti-static packing material for storing or returning the ZT 8841.

WARNING!

Like all equipment utilizing CMOS devices, the ZT 8841 must be protected from static discharge. Never remove any of the socketed parts except at a static-free work station.

SYSTEM REQUIREMENTS

The ZT 8841 is designed for use in an STD bus application. The board is mechanically and electrically compatible with the STD 32 and STD-80 Series bus specifications.

The board requires +5 VDC ±5% at 200 mA maximum and ±12 VDC ±10% at 60 mA maximum. A typical ZT 8841 requires about 92 mA at +5 V and 30 mA at ±12 V.

If RS-422/485 is used, the board requires 345 mA maximum, 162 mA typical to +5 V; ±12 V are not required.

If RS-422/485 with optical isolation is used, the board requires 406 mA maximum, 242 mA typical to +5 V; ±12 V are not required.

The ambient temperature range must be maintained between 0 and +65˚Celsius for proper operation and to avoid damage. If the extended temperature option (XTP) is ordered (ZT 88CT41), the ambient temperature range must be between -40 and +85˚ Celsius for operation.

When RS-422/485 is used, the ambient temperature must be between 0 and +65˚ Celsius even if the ZT 88CT41 is purchased. Military quality parts can be substituted in the RS-422/485 driver sockets if extended temperature operation is required.

The relative humidity must be <95% at 40˚ C, non-condensing. Refer to Appendix B for additional specifications.

Vertical mounting is recommended in convective cooling systems not equipped with a fan. However, most systems require a fan for reliable operation. Horizontal mounting is not recommended unless forced air cooling of at least 30 cubic ft/min. is provided.

If a standard STD bus card cage is not used, refer to the figure on page B-15 for dimensions. Be sure to allow enough clearance for good convective or forced cooling.

PROCESSOR COMPATIBILITY

The ZT 8841 is designed for STD 32 and STD-80 systems. It can be used with other microprocessors only if they meet the STD-80 cascade interrupt and WAITRQ* timing specifications.

I/O ADDRESSING

Each device communicating across the STD bus is assigned a range of locations between I/O addresses 0h and FFFFh (16 bits). The ZT 8841 I/O address range is comprised of 40 nonconsecutive I/O addresses and a Base I/O Address.

The ZT 8841 uses several jumpers to set the board at a specific Base I/O Address. Table 2-1 on page 2-6 illustrates how installing or removing these jumpers changes the Base I/O Address.

Each jumper is associated with an address bit (A10-A15). The bits are grouped in fours (nibbles). Each nibble forms a hexadecimal digit (see Table 2-2 on page 2-7 for binary to hexadecimal conversion). The four nibbles make up the four hexadecimal digits of the Base I/O Address.

The Base I/O Address determines the upper six address bits (W1-6 correspond to A15-10), which are added to the remaining non-consecutive I/O address bits determined by jumpers W9,10 in Table 2-3 on page 2-8.

Table 2-1
I/O Address Jumper Selection.

Base I/O Address	Jumper Number [1] W1	W2	W3	W4	W5	W6	-	-	-	-	-	-	-	-	-	-	-	-
	Address Bit A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
Default: 0000 [2]	(0) 0	0	0	0	(0) 0	0	X	X	(0) X	X	X	X	(0) X	X	X	X
Example: 1400	(1) 0	0	0	1	(4) 0	1	X	X	(0) X	X	X	X	(0) X	X	X	X
Customer Base Address	(___)				(___)				(0)				(0)			
(_____ - _____)	—	—	—	—	—	—	X	X	X	X	X	X	X	X	X	X

Notes: [1] 0 indicates jumper installed
1 indicates jumper removed
X indicates "don't care," where jumper may be installed or removed

[2] Factory default jumper configuration

Table 2-2
Decimal/Hexadecimal/Binary Conversion.

Decimal	Hexadecimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

I/O Port Selection

Each serial channel on the ZT 8841 is implemented with an 82050 UART integrated circuit, also referred to as a communications (COM) channel. Since the ZT 8841 is a Quad UART board, communication channels are numbered COM1 to COM4. Each COM (UART) channel is programmed through I/O read and write instructions to various I/O registers.

The ZT 8841 uses 40 nonconsecutive I/O registers to program the board. The four UARTs and the PIC are each allocated eight consecutive registers (see Table 2-4). The PIC registers are redundantly mapped.

These registers are addressed according to the Base I/O Address jumper settings as described on page 2-5 and the COM/PIC address jumper settings indicated in Table 2-3 below.

Table 2-3
COM Address Configurations.

W10	W9	COM1	COM2	COM3	COM4	PIC
IN	IN	3F8-3FF	2F8-2FF	3E8-3EF	2E8-2EF	020-027
OUT	IN	3E0-3E8	2F8-2FF	3E8-3EF	2E8-2EF	020-027
IN	OUT	3E0-3E8	2F8-2FF	3E8-3EF	2E8-2EF	0A8-0AF
OUT*	OUT*	3E0-3E8	2E0-2E8	3E8-3EF	2E8-2EF	0A8-0AF

After setting W10 and W9 to the desired COM address configuration, use Table 2-4 for programming the various registers within each COM channel (UART) and the PIC. The actual value for the "UART Base +" entry in Table 2-4is determined by adding the ZT 8841 Base I/O Address to the COM address selection made above. Factory default jumpers set the upper six ZT 8841 Base I/O Address bits to zero.

Table 2-4
I/O Registers.

UART Base +	I/O Read Register	I/O Write Register
0000	Receive Buffer	Transmitter Buffer
0001	Interrupt Enable	Interrupt Enable
0002	Interrupt Ident.	---
0003	Line Control	Line Control
0004	Modem Control	Modem Control
0005	Line Status	Line Status
0006	Modem Status	---
0007	Scratchpad	Scratchpad
PIC Base +	**I/O Read Register**	**I/O Write Register**
0000	IRR, ISR, IL	ICW1, OCW2, OCW3,
0001	IMR	ICW2,3,4, OCW1

For example, to program the Line Control register for COM2 with W10 and W9 removed, and with a Base I/O Address of zero (W1-6 installed, factory default):

1. Add the upper six bits of the Base I/O Address (0h) to the COM address found in Table 2-3 (COM2 with W10 and W9 removed = 2E0h). The result is a UART Base Address of 02E0h.
2. Add the UART Base Address (02E0h) to the "UART Base +" value from Table 2-4 for the Line Control register (0003h). The result is an I/O address of the COM2 Line Control register of 02E3h.

Individual register definitions are found in Chapter 6, "Serial Communications (82050)." Refer to this chapter before programming the UARTs.

Each UART occupies eight consecutive I/O addresses within the 64K I/O address space of the processor. The optimum I/O address map for any given Ziatech CPU board with one or more on-board UARTs is given in the remaining sections of this chapter. Review the I/O map for your processor to determine the recommended W10, W9, and interrupt configurations.

Table 2-3 on page 2-8 provides a summary of the possible I/O addresses for each UART and the PIC. Interrupt configuration jumpers are described on pages A-9, A-19, and in the "Interrupts" section beginning on page 3-18.

The following tables *assume Ziatech default jumper configurations.* They provide an overview of the STD DOS system COM addresses. Refer to Ziatech's STD DOS Manual for DOS-specific information or Ziatech's Device Driver Manual (DDP) for software driver information.

ZT 8801 COM I/O Addresses

The ZT 8801 processor boards do not have an on-board IBM-equivalent UART at an address that will overlap with the ZT 8841.

Table 2-5
ZT 8801 COM Configuration.

COM Port	ZT 8801 w/o Modem †	ZT 8801 w/ Modem ††	I/O Address
COM1	ZT 8841 J1	ZT 8841 J1	03F8h
COM2	ZT 8841 J2	ZT 8843	02F8h
COM3	ZT 8841 J3	ZT 8841 J3	03E8h
COM4	ZT 8841 J4	ZT 8841 J4	02E8h

† ZT 8841 W9 and W10 are installed, W11 is removed.

†† ZT 8841 W9 is installed, W10 and W11 are removed. ZT 8843 modem must be rejumpered for COM1 operation. Remaining ZT 8841 J1 serial channel is at 03E0h.

ZT 8806/8807 COM I/O Addresses

The ZT 8806/8807 processor boards do not have an on-board IBM-equivalent UART at an address that will overlap with the ZT 8841.

Table 2-6
ZT 8806/8807 COM Configuration.

COM Port	ZT 8806/8807 w/o Modem †	ZT 8806/8807 w/ Modem ††	I/O Address
COM1	ZT 8841 J1	ZT 8843	03F8h
COM2	ZT 8841 J2	ZT 8841 J2	02F8h
COM3	ZT 8841 J3	ZT 8841 J3	03E8h
COM4	ZT 8841 J4	ZT 8841 J4	02E8h

† ZT 8841 W9 and W10 are installed, W11 is removed.

†† ZT 8841 W9 is installed, W10 and W11 are removed. ZT 8843 modem must be rejumpered for COM1 operation. Remaining ZT 8841 J1 serial channel is at 03E0h.

ZT 8808/8809 Rev B or Later COM I/O Addresses

The ZT 8808/8809 processor boards have two on-board IBM-equivalent UARTs at addresses that could overlap with the ZT 8841.

Table 2-7
ZT 8808/8809 COM Configuration.

COM Port	ZT 8808/8809 w/o Modem †	ZT 8808/8809 w/ Modem †, ††	I/O Address
COM1	ZT 8808/8809 J1	ZT 8808/8809	03F8h
COM2	ZT 8808/8809 J2	ZT 8843	02F8h
COM3	ZT 8841 J3	ZT 8841 J3	03E8h
COM4	ZT 8841 J4	ZT 8841 J4	02E8h

† ZT 8841 W11 is installed, W9 and W10 are removed; remaining ZT 8841 J1 and J2 serial channels are at 3E0h and 2E0h, respectively.

†† ZT 8808/8809 jumper W66 must be out to disable on-board COM2 for the ZT 8843 modem board.

ZT 8816/8817 Rev B or Later COM I/O Addresses

The ZT 8816/8817 Revision B processor boards have one on-board IBM-equivalent UART at an address that could overlap with the ZT 8841.

Table 2-8
ZT 8816/8817 COM Configuration.

COM Port	ZT 8816/8817 w/o Modem †, ‡	ZT 8816/8817 w/ Modem ††, ‡	I/O Address
COM1	ZT 8816/8817 J2	ZT 8816/8817 J2	03F8h
COM2	ZT 8841 J2	ZT 8843	02F8h
COM3	ZT 8841 J3	ZT 8841 J3	03E8h
COM4	ZT 8841 J4	ZT 8841 J4	02E8h

† ZT 8841 W10, W11 are installed, W9 removed; remaining ZT 8841 J1 serial channel is at 03E0h. Also install ZT 8841 W19.

†† ZT 8841 W11 is installed, W9 and W10 are removed; remaining ZT 8841 J1 and J2 serial channels are at 03E0h and 02E0h, respectively.

‡ ZT 8816 IR3 must be connected to STD CNTL* (INTRQ2*); install jumper from the left side of W49 to the left side of W50.

ZT 8901 COM I/O Addresses

The ZT 8901 processor boards have two on-board IBM-equivalent UARTs at addresses that could overlap with the ZT 8841.

Table 2-9
ZT 8901 COM Configuration.

COM Port	ZT 8901 w/o Modem †	ZT 8901 w/ Modem †, ††	I/O Address
COM1	ZT 8901 J5	ZT 8901	03F8h
COM2	ZT 8901 J4	ZT 8843	02F8h
COM3	ZT 8841 J3	ZT 8841 J3	03E8h
COM4	ZT 8841 J4	ZT 8841 J4	02E8h

† ZT 8841 W11 is installed, W9 and W10 are removed; remaining ZT 8841 J1 and J2 serial channels are at 3E0h and 2E0h, respectively.

†† ZT 8901 jumper W66 must be removed to disable on-board COM2 for the ZT 8843 modem board.

ZT 8902 COM I/O Addresses

The ZT 8902 processor board has two on-board IBM-equivalent UARTs at addresses that could overlap with the ZT 8841.

Table 2-10
ZT 8902 COM Configuration.

COM Port	ZT 8902 w/o Modem †	ZT 8902 w/ Modem †, ††	I/O Address
COM1	ZT 8902 J8	ZT 8902 J8	03F8h
COM2	ZT 8902 J7	ZT 8843	02F8h
COM3	ZT 8841 J3	ZT 8841 J3	03E8h
COM4	ZT 8841 J4	ZT 8841 J4	02E8h

† ZT 8841 W11 is installed, W9 and W10 are removed; remaining ZT 8841 J1 and J2 serial channels are at 3E0h and 2E0h, respectively.

†† ZT 8902 jumper W17 must be removed to disable on-board COM2 for the ZT 8843 modem board.

ZT 8910 COM I/O Addresses

The ZT 8910 processor board has two on-board IBM-equivalent UARTs at addresses that could overlap with the ZT 8841.

Table 2-11
ZT 8910 COM Configuration.

COM Port	ZT 8910 w/o Modem †	ZT 8910 w/ Modem †, ††	I/O Address ‡
COM1	ZT 8910 J1	ZT 8910 J1	03F8h
COM2	ZT 8910 J2	ZT 8843	02F8h
COM3	ZT 8841 J3	ZT 8841 J3	03E8h
COM4	ZT 8841 J4	ZT 8841 J4	02E8h

† ZT 8841 W11 is installed, W9 and W10 are removed; remaining ZT 8841 J1 and J2 serial channels are at 3E0h and 2E0h, respectively.

†† ZT 8910 jumper W66 must be removed to disable on-board COM2 for the ZT 8843 modem board.

‡ The ZT 8910 reserves I/O port addresses 0 - FFh for on-board use. To access the interrupt controller on the ZT 88CT41, an I/O port address above this range must be used. This will affect the I/O addresses for the COM ports as well. Refer to the description of jumpers W1-6 in Appendix A.

ZT 8911 COM I/O Addresses

The ZT 8911 processor board has two on-board IBM-equivalent UARTs at addresses that could overlap with the ZT 8841.

Table 2-12
ZT 8911 COM Configuration.

COM Port	ZT 8911 w/o Modem †	ZT 8911 w/ Modem †, ††	I/O Address
COM1	ZT 8911 J10	ZT 8911 J10	03F8h
COM2	ZT 8911 J9	ZT 8843	02F8h
COM3	ZT 8841 J3	ZT 8841 J3	03E8h
COM4	ZT 8841 J4	ZT 8841 J4	02E8h

† ZT 8841 W11 is installed, W9 and W10 are removed; remaining ZT 8841 J1 and J2 serial channels are at 3E0h and 2E0h, respectively.

†† Disable COM2 on the ZT 8911 via the software configuration registers.

DEFAULT CONFIGURATION

The ZT 8841 includes several jumpers that increase its flexibility by allowing the selection of various operating conditions. The list below shows the operating conditions set by the factory default jumper configuration. Figure 2-1 illustrates the jumper settings for the factory default.

The ZT 8841 does not require jumper changes to operate. However, if a different CPU or I/O address configuration is needed, the jumpers must be changed. Refer to Appendix A, "Jumper Configurations," for more information.

Factory Defaults and Initial Conditions

- PCI and PCO are connected together
- 16-bit I/O addressing
- IOEXP disabled
- 40 I/O ports required
- Serial channels (COM1-4) jumpered for DTE
- RS-422/485 disabled on COM2 and COM3
- Jumpered for ZT 8808/8809 without a modem

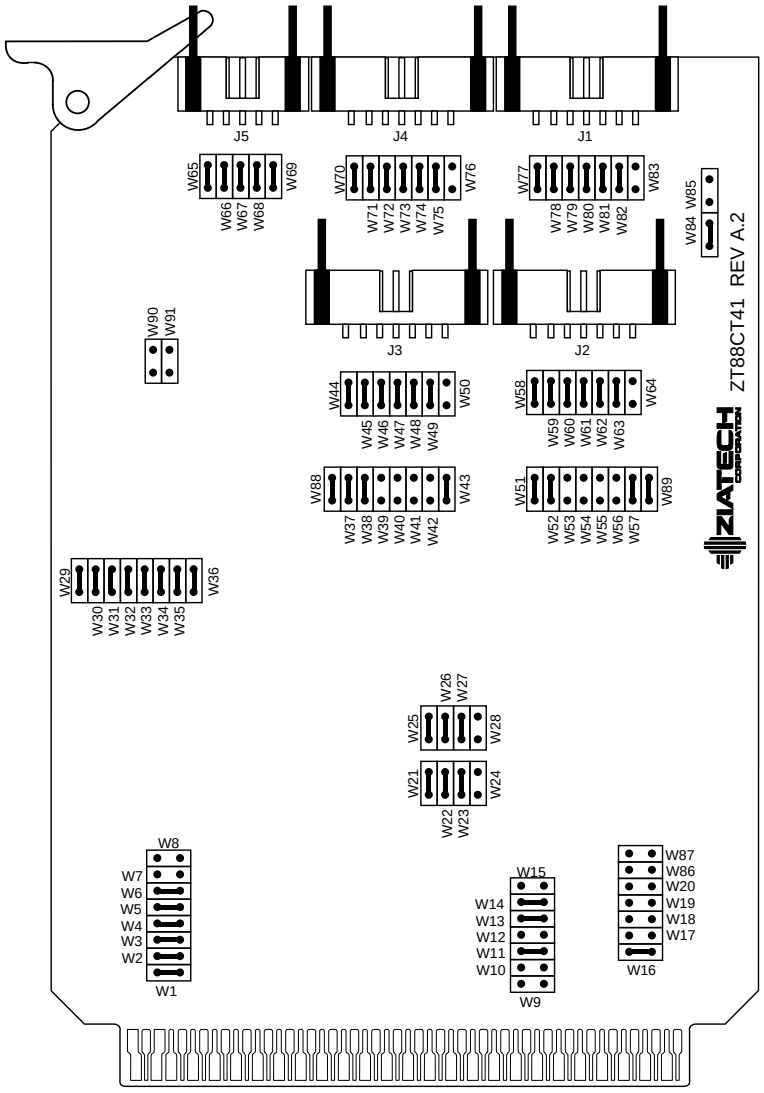


Figure 2–1. ZT 8841 Factory Default Configuration.

Table 2-13 illustrates the default I/O address configuration for serial channels COM1-4. All entries are given for 16-bit I/O addressing.

Table 2-13
Default COM Address Configuration.

COM1	COM2	COM3	COM4	PIC
03E0-03E8h	02E0-02E8h	03E8-03EFh	02E8-02EFh	00A8-00AFh

JUMPER DESCRIPTIONS

The ZT 8841 includes several jumper-selectable options that tailor the operation of the board to the requirements of specific applications. These options are summarized in Appendix A, "Jumper Configurations." Figure 2-1 illustrates the factory default jumper configuration. You may wish to use the diagram on page A-4 to record a custom jumper configuration.

PIN ASSIGNMENTS

The ZT 88CT41 has four serial connectors, J1-4. Each serial channel may be jumpered for DTE (default) or DCE operation (see page A-16). Figures 2-2 and 2-3 illustrate pin assignments and signal flow for RS-232-C operation when used with the ZT 90027 (male) or ZT 90014 (female) cables, respectively.

Serial channels J2 and J3 may be jumpered for RS-422/485 operation. Figure 2-4 illustrates pin assignments for use of either the ZT 90027 (male) or ZT 90014 (female) cable.

Additional pin assignment tables appear on page B-8.

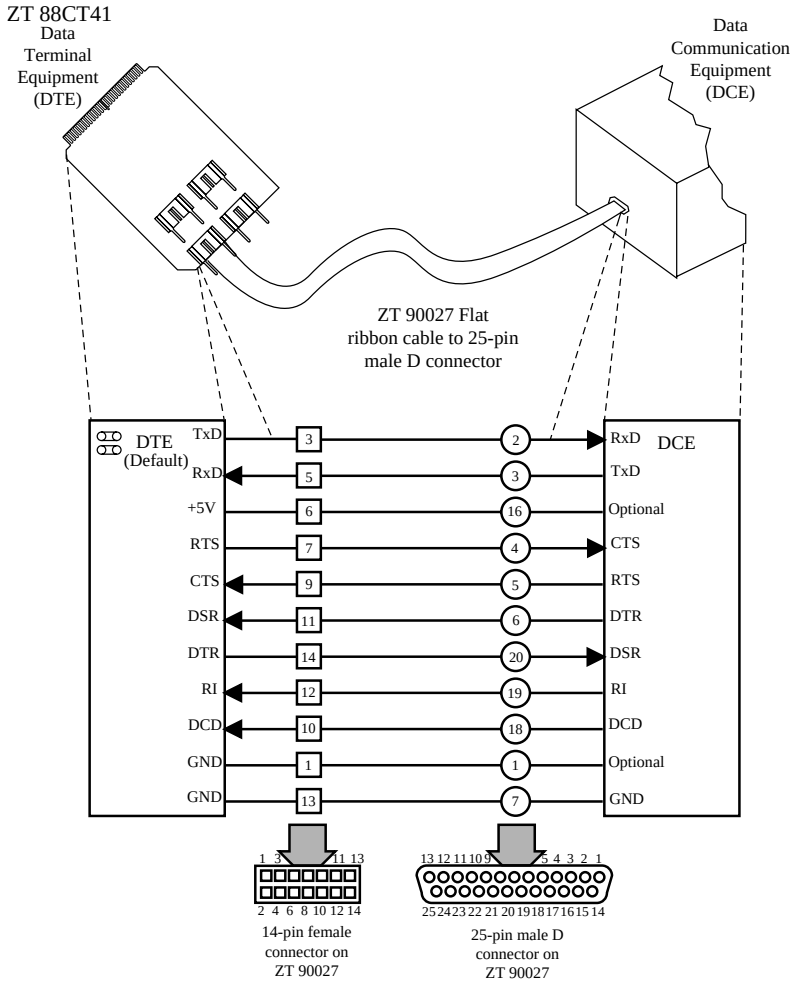


Figure 2–2. RS-232-C DTE and ZT 90027 Diagram.

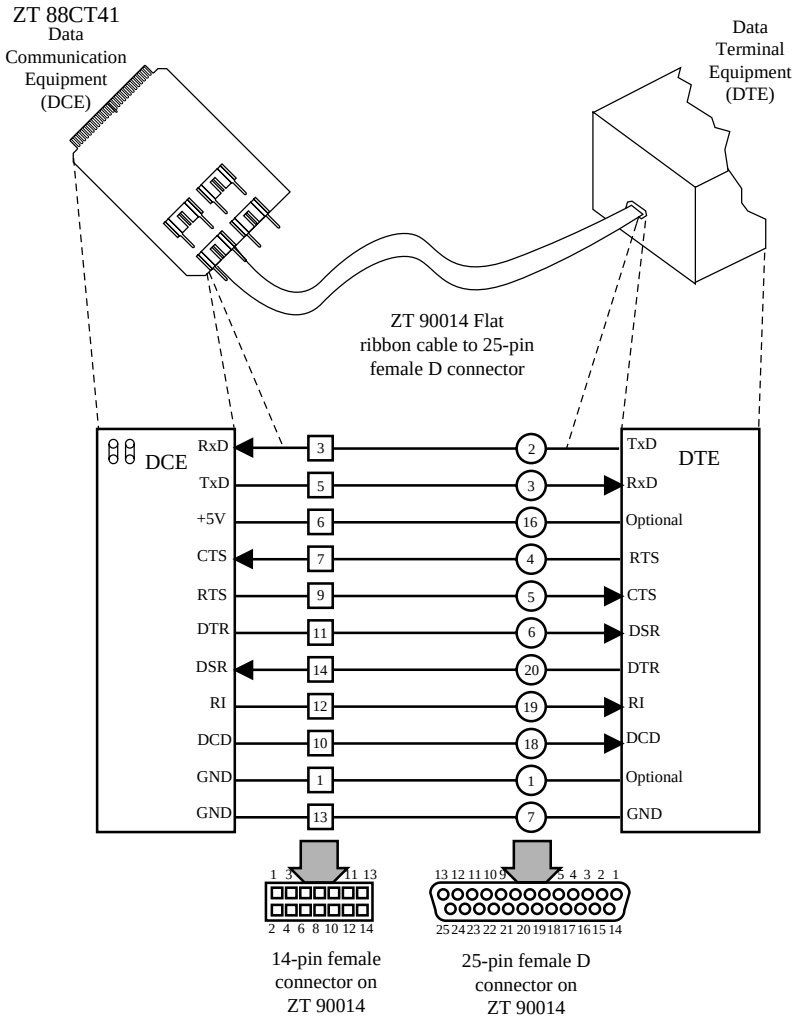


Figure 2–3. RS-232-C DCE and ZT 90014 Diagram.

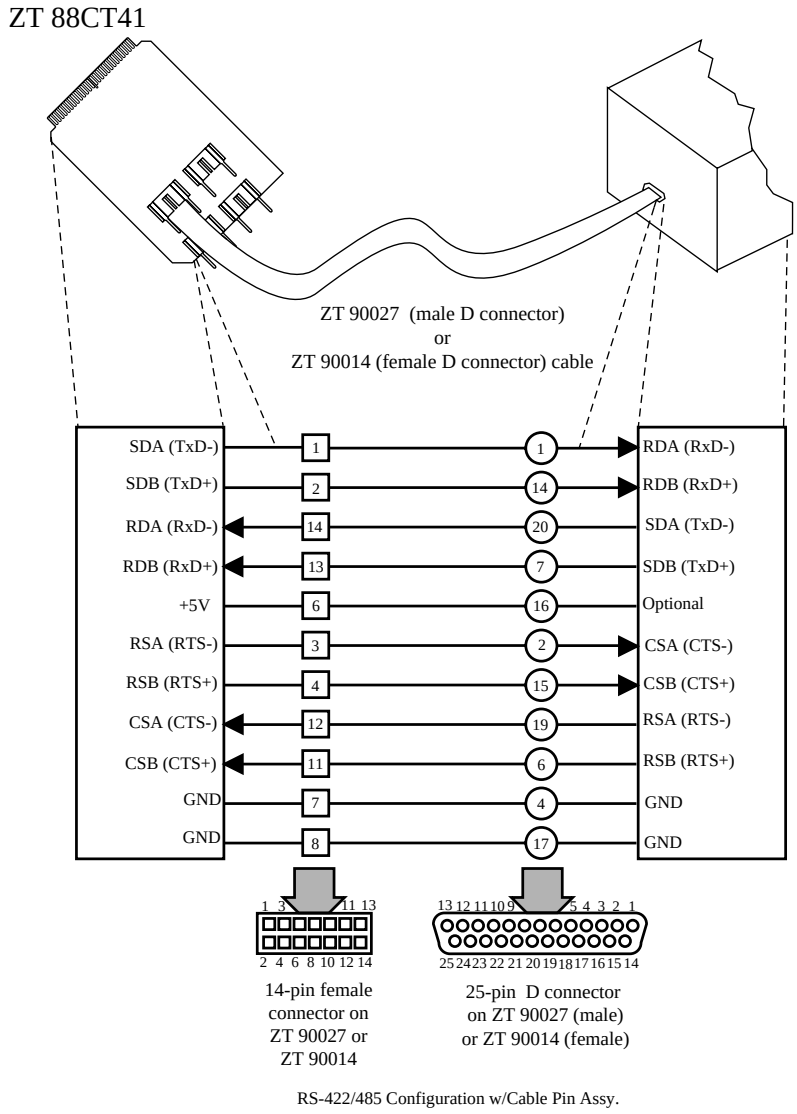


Figure 2–4. RS-422/485 and Cable Diagram.

FURTHER INFORMATION

Further information on interrupts and the Intel 8259A PIC, 82050 UART, 82510 UART, and MCS-51 can be obtained from the following sources:

Interrupts: *Ziatechniques #8* from Ziatech describes how interrupts are used in a system. This publication is suitable for people of all experience levels. All aspects of interrupts are covered for both DOS and non-DOS systems. Contact Ziatech or your local Ziatech Representative for a copy.

8259A PIC: The Intel *Microprocessor And Peripherals Data Book* is the primary reference for the 8259A Programmable Interrupt Controller (PIC). It contains a data sheet on the 8259A and an excellent application note on the 8259A PIC. The application note is *AP59* and will be the most important reference material for the 8259A PIC.

82050 UART: For a data sheet on the Intel 82050 UART, contact your local Intel sales office. For programming purposes, if you cannot locate a 82050 data sheet, then use any handy data sheet on the 8250, 16450, or similar UARTs. The 82050 UART is program compatible with the 8250, 16450, or similar UARTs.

82510 UART: The Intel *Microcommunications Handbook* contains a data sheet and an application note on the 82510 UART. The application note is called *Designing With The 82510 Asynchronous Serial Controller, AP-401*. Low power, FIFO buffer, and interrupt operation are covered along with some programming examples.

MCS-51: The Intel *MCS-51 User's Manual* describes this protocol used to communicate in a master/slave configuration.

III. USER'S REFERENCE

THEORY OF OPERATION 3-1
APPLICATION EXAMPLE 4-1
INTERRUPT CONTROLLER (8259A) 5-1
SERIAL COMMUNICATIONS (82050) 6-1

Chapter 3

THEORY OF OPERATION

Contents **Page**

OVERVIEW 3-2
COMMONLY ASKED QUESTIONS 3-2
ASYNCHRONOUS COMMUNICATIONS 3-5
Standards 3-5
RS-232-C 3-5
RS-422/485 3-5
Configurations 3-6
DTE 3-6
DCE 3-7
INTERRUPT OPERATION 3-8
Fundamentals 3-8
Master PICs 3-9
Slave PICs 3-9
Configurations 3-11
Polled Interrupts 3-11
Backplane Interrupts 3-13
Frontplane Interrupts 3-14
FUNCTIONAL OPERATION 3-15
STD Interface 3-16
I/O Addressing 3-16
Wait-States 3-16
Serial Interface 3-17
Interrupts 3-18
RS-232-C 3-22
RS-422/485 3-23

OVERVIEW

This chapter presents a high-level look at the way the ZT 8841 functions. The chapter first covers commonly asked questions, asynchronous communications, and interrupt operation to provide a basis for the functional operation of the board. STD and serial interfaces are then discussed. Figure 3-7 (page 3-15) illustrates the basic functional diagram of the ZT 8841.

COMMONLY ASKED QUESTIONS

1. **What are the differences between the ZT 8841 and ZT 8840 Quad UART Boards?**

   – The ZT 8840 uses 32 consecutive I/O port addresses and maps the PIC into two unused registers within the UARTs. Only 8-bit I/O decoding is supported by the ZT 8840. The ZT 8841 requires 40 I/O port addresses that are jumper configurable. Several configurations intersperse the UART I/O addresses between the IBM COM compatible I/O addresses 2E8h (COM4), 2F8h (COM2), 3E8h (COM3), and 3F8h (COM1). 16-bit I/O addressing is also supported, allowing the upper six address bits to be jumper selectable.

   – The ZT 8840 operates only in RS-232-C modes while the ZT 8841 supports RS-422/485 operation on two of the four serial channels. Optionally, the RS-422/485 serial channels can be optically isolated for greater noise immunity.

   – The ZT 8840 interrupt structure allows the on-board PIC to operate as a master or slave to other PICs off board. Individual UART interrupts are not available to the backplane or frontplane without going through the on-board PIC. The ZT 8841 on-board PIC can operate as a slave to another off-board PIC. Individual UART interrupts are available on the backplane or frontplane if bypassing the on-board PIC is desired. If the on-board PIC

is used, jumper posts are provided to allow any UART to be connected to any interrupt request input on the PIC.

- The ZT 8840 provided no power on the RS-232-C connectors. The ZT 8841 has jumpers to connect +12 V to the connector. This feature is useful for some serial devices that require power to be supplied (for example, a mouse).
- The ZT 8840 is implemented with bipolar TTL devices. The ZT 88CT41 uses CMOS parts for lower power and extended temperature operation (XTP). Optionally, the ZT 88CT41 can be purchased with enhanced UARTs that have FIFO buffering, 0 to 288 Kbaud rates, and a software power-down mode to conserve even more power. A bipolar version of the ZT 8841 is also available.

2. **What is the difference between the ZT 88CT41 and ZT 8841?**

   - The ZT 88CT41 and ZT 8841 both utilize CMOS devices except for the PAL (Programmable Array Logic) devices used. The 88CT41 utilizes the more expensive CMOS parts, whereas the ZT 8841 uses bipolar parts; the boards are otherwise identical.

3. **What are the benefits of optical isolation?**

   - It provides greater computer immunity to noise that may be picked up on the serial cable. In some environments in which lightning, arc welders, etc. are present, 600 V spikes are prevented from entering the computer.
   - In catastrophic cases in which noise spikes actually destroy the RS-422/485 drivers without destroying the ZT 88CT41, repair costs are much lower. In most cases repair is accomplished by simply replacing the drivers.

4. **Can optical isolation be used at any baud rate?**

   – Yes, if the baud rate on the RS-422/485 channel is between 0 and 56,000 baud. This is the case for the standard ZT 8841.

   – If the enhanced serial option is ordered (82510 UARTs), the baud rate can be between 0 and 288,000 baud. Higher baud rates saturate the optical isolation circuit, preventing reliable data transmission. If you plan to operate above 288,000 baud with the 82510 UARTs, do not use the optical isolation option.

ASYNCHRONOUS COMMUNICATIONS

Data communications between equipment and computers, or between computers and computers, have been facilitated by asynchronous serial protocols. These protocols have become standards widely used in industry today. A brief overview of their differences follows.

Standards

The Electronics Industries Association (EIA) has played a major role in adopting several standards that deal with data transmission. These standards are RS-232-C, RS-422, and RS-485.

RS-232-C

The first of these, RS-232-C, was introduced in 1962 and has been widely used throughout the industry. RS-232-C was developed for single-ended data transmission at relatively slow data rates (20 Kbaud) over short distances (up to 50 feet).

When transmitting at very high data rates, over long distances, and through noisy environments, double-ended transmission offers superior performance. Differential transmission nullifies the effects of ground shifts and noisy signals that appear as common mode voltages on the transmission line. RS-422/485 uses differential drivers.

RS-422/485

RS-422 removes the effects of ground shifts. It allows data rates up to 10 Mbaud (up to 40 feet) and line lengths up to 4000 feet (up to 100 Kbaud).

Drivers designed to meet this standard are well suited for party-line applications in which only one driver is connected to, and transmits on, a bus and up to 32 receivers can receive the data. While party-line applications have many uses, RS-422 devices cannot be used to construct a truly multidrop bus. A multidrop bus consists of multiple

drivers and receivers connected to a single bus in which each driver/receiver can transmit or receive data.

To meet the need for truly multidrop communications, the EIA established RS-485 in 1983. RS-485 meets all the requirements of RS-422. In addition, this new standard allows up to 32 drivers and 32 receivers to be connected to a single bus to form a truly multidrop bus.

The key features of RS-485 are:

- Allows 32 drivers and 32 receivers
- Extended common-mode rejection
- Drivers can withstand bus contention

Configurations

The EIA has defined Data Terminal Equipment (DTE) and Data Communication Equipment (DCE) to allow users of the RS-232-C and RS-422/485 to easily connect equipment. DTE and DCE basically define the pins and the connector sex used for data communications.

DTE

Data Terminal Equipment (DTE) utilizes a male connector that is associated with, but not necessarily physically attached to, the equipment. For RS-232-C, this connector pinout is well defined and is shown on page B-10. For RS-422/485 operation, there is no widely accepted connector pinout.

DCE

Data Communication Equipment (DCE) utilizes a female connector that is associated with, but not necessarily physically attached to, the equipment. For RS-232-C, this connector pinout is well defined and is shown on page B-10. For RS-422/485 operation, there is no widely accepted connector pinout.

INTERRUPT OPERATION

The ZT 8841 can generate interrupts to the CPU if desired. Interrupts are useful when trying to preserve CPU throughput by not requiring the CPU to constantly poll the board for needed service. Service is performed only when the interrupt occurs, as illustrated in Figure 3-1.

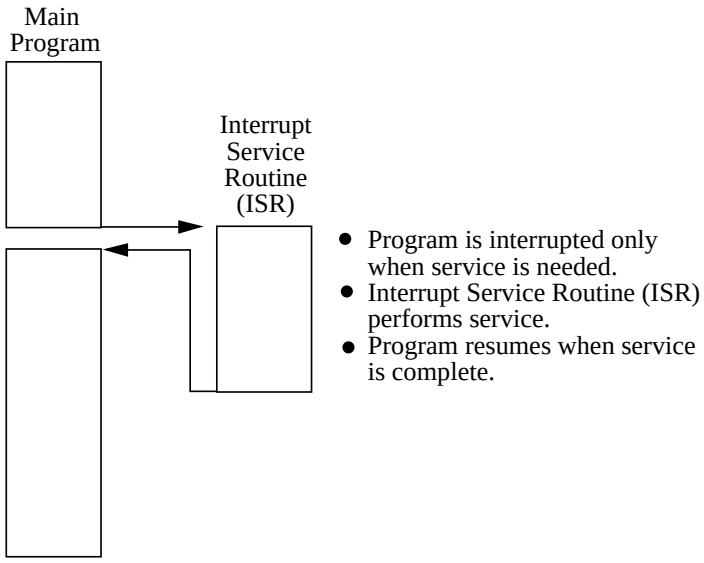


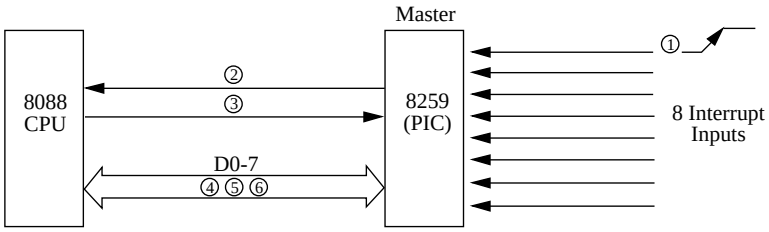
Figure 3–1. Interrupt For Service.

Fundamentals

When service is needed, the interrupt "vectors" the processor from the main program to the Interrupt Service Routine (ISR). The vector is generated from an 8259A Programmable Interrupt Controller (PIC) on either the CPU itself or the ZT 8841.

Master PICs

If the CPU has a PIC on-board (master), the ZT 8841 can either generate an interrupt to the PIC for vector generation or use the on-board PIC (slave) to generate the vector. Figure 3-2 illustrates the interrupt sequence that would occur if one or more UARTs on the ZT 8841 were to generate the interrupt inputs to the master PIC on a CPU. Interrupt sources from the ZT 8841 would originate directly from one or more of the UARTs, bypassing the ZT 8841 PIC.



1. An interrupt input occurs from:
   -On-board counter/timer, clock, etc.
   -Backplane INTRQ*, INTRQ1*, INTRQ2*
   -Frontplane FP1, FP2, etc.
   This interrupt is held active until step 5.
2. Interrupt request sent to the CPU.
3. CPU acknowledges interrupt.
4. Master PIC vectors CPU to interrupt service routine.
5. CPU clears interrupt source.
6. CPU issues End of Interrupt (EOI) to PIC and returns to main program.

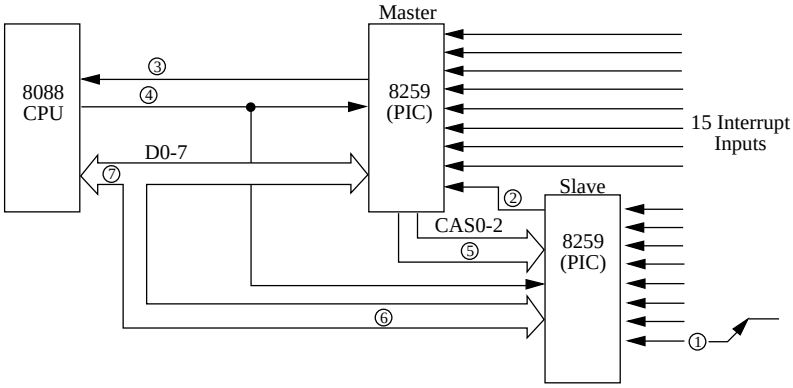
Figure 3–2. Master Interrupt Sequence.

Slave PICs

Figure 3-3 illustrates the interrupt sequence that would occur if the ZT 8841 PIC were to generate the interrupt input to the master PIC on a CPU. In either case, the vector actually pertains to one of 256 interrupt "types" supported by the processor in memory.

All 256 interrupt types are located in absolute memory locations 0 through 3FFh, which make up the processor's interrupt vector table. Each type in the interrupt vector table requires four bytes of memory and stores a code segment address and an instruction pointer address. Locations 0 through 3FFh should be reserved for the interrupt vector table alone.

When the processor receives an interrupt vector from the PIC, it multiplies its value by four to acquire the address of the interrupt type. Program execution is then vectored to the interrupt service routine whose address is specified within the interrupt vector table.



1. An interrupt input occurs from:
   -On-board counter/timer, clock, etc.
   -Backplane INTRQ*, INTRQ1*, INTRQ2*
   -Frontplane FP1, FP2, etc.
   This interrupt is held active until step 5.
2. Slave interrupts master.
3. Interrupt request sent to CPU.
4. CPU acknowledges interrupt.
5. Master selects slave PIC.
6. Slave PIC vectors CPU to interrupt service routine.
7. CPU issues End of Interrupt (EOI) to both PICs and returns to main program.

Figure 3–3. Master/Slave Interrupt Sequence.

Configurations

Several possible configurations for the ZT 8841 can be used when using interrupts, depending on whether the CPU or the ZT 8841 PIC generates the interrupt vector. When the CPU PIC generates the vector because off-board slave PICs are not supported (ZT 8806), the interrupt line on the STD bus may be shared. If this is your system configuration, the "polled interrupt" configuration described below will be of interest.

If the CPU allows the ZT 8841 to generate the interrupt vector when cascaded to a master PIC (ZT 8808, ZT 8816), the interrupt can be connected via either the backplane or the frontplane as described on pages 3-13 and 3-14, respectively.

Polled Interrupts

Polled interrupts are useful when the ZT 8841 PIC cannot be used to generate an interrupt vector. For example, if only one backplane interrupt line (INTRQ*) is used for multiple sources, as illustrated in Figure 3-4, the ZT 8841 PIC cannot generate the interrupt vector. In this mode, the ZT 8841 "interrupt" line is actually the logical OR of the four UART interrupt output signals. The PIC on the ZT 8841 performs the OR function, allowing each UART interrupt to be masked or unmasked. The ZT 8841 "interrupt" line can be routed over the backplane or frontplane to signal the CPU when service is needed. Once interrupted, the CPU can interrogate the ZT 8841 PIC to determine which of the four UARTs needs service. Jumper W11 must be removed to enable this mode. For more information on how to program the PIC for polled operation, refer to Intel’s application note AP59 in the *Microprocessor and Peripheral Handbook*.

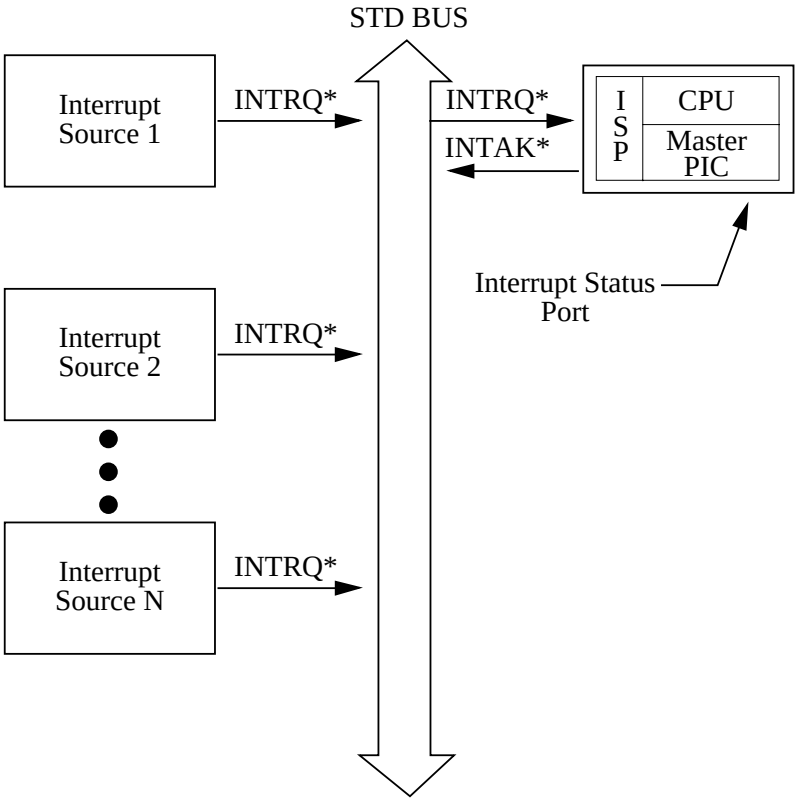


Figure 3–4. Shared STD Bus Interrupt Structure.

Backplane Interrupts

When the ZT 8841 PIC generates the interrupt vector, the interrupt can be routed over the backplane or frontplane. Two dedicated interrupt lines are available on the STD bus in addition to the shared INTRQ* line. These dedicated interrupts are the INTRQ1* and CNTL (INTRQ2*) signals on the backplane. Figure 3-5 illustrates several ZT 8841 "interrupts" being routed over the INTRQ*, INTRQ1*, CNTL (INTRQ2*), and frontplane signals when operating in polled mode with the CPU PIC generating the interrupt vectors. In contrast, Figure 3-6 illustrates the ZT 8841 PIC generating interrupt vectors using the INTRQ*, INTRQ1*, CNTL (INTRQ2*), and frontplane signals.

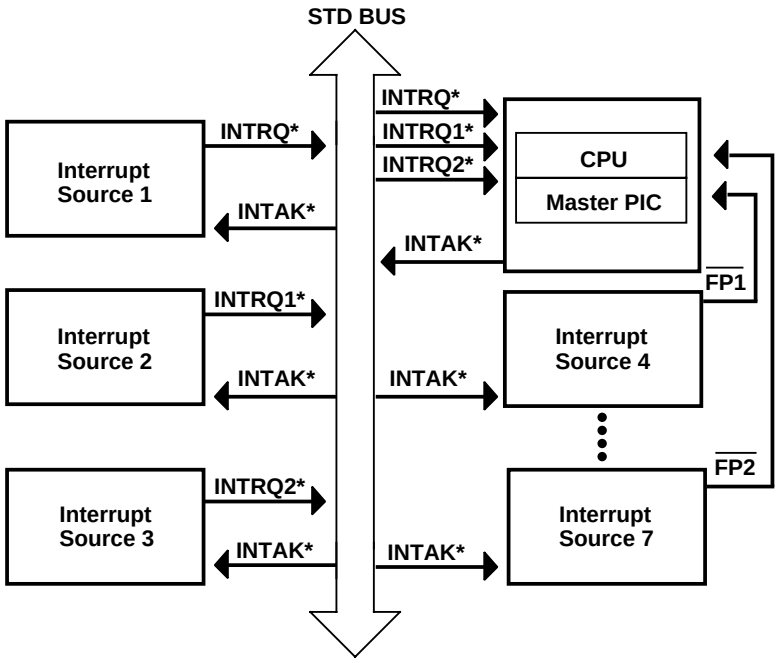


Figure 3–5. Dedicated STD Bus Interrupt Structure.

Frontplane Interrupts

When the ZT 8841 PIC generates the interrupt vector, the interrupt can be routed over the backplane or frontplane. Generally, there are five frontplane interrupt lines, depending on the CPU used. Frontplane interrupts are connected to the CPU by means of a ribbon cable chained from one frontplane interrupt board to another. Figure 3-6 shows several ZT 8841 interrupts being routed over the INTRQ*, INTRQ1*, CNTL (INTRQ2*), and frontplane signals with the ZT 8841 PIC (slave PIC) generating the interrupt vectors.

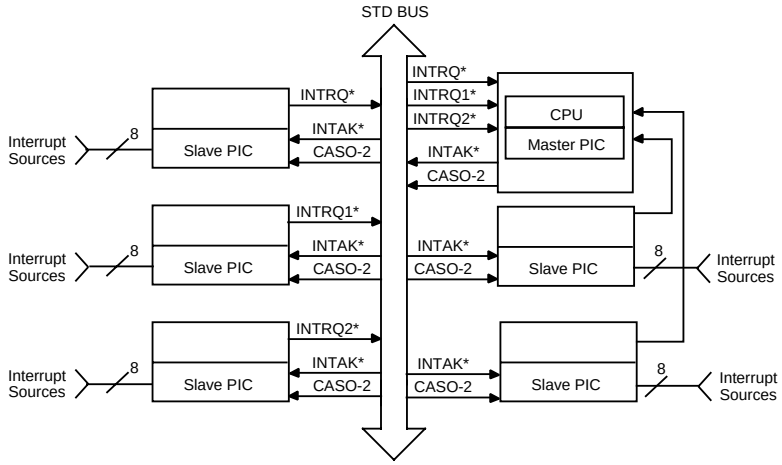


Figure 3–6. Dedicated Frontplane Interrupt Structure.

FUNCTIONAL OPERATION

Previous sections have covered basic asynchronous communications and STD bus interrupt operation. This section describes the ZT 8841 STD Interface and Serial Interface. Refer to Figure 3-7 as you read this section.

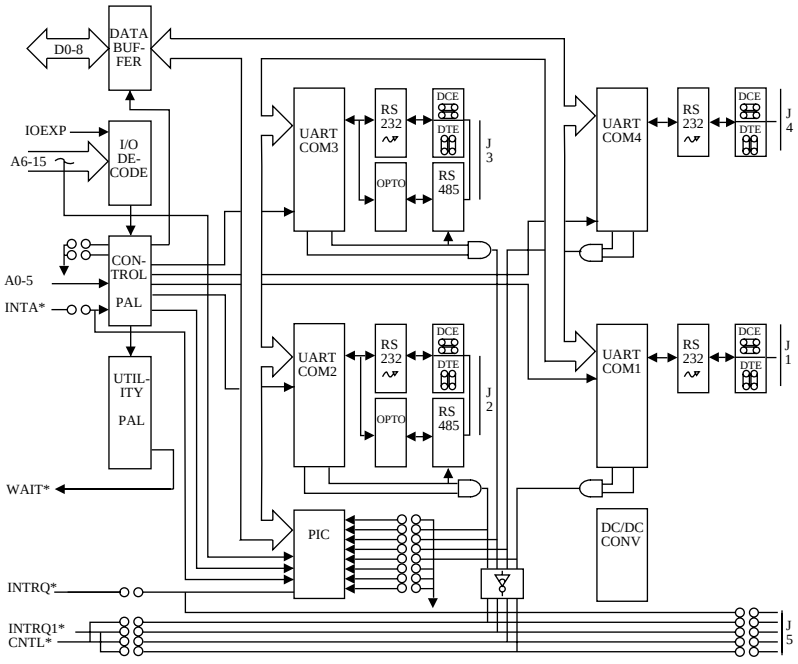


Figure 3–7. ZT 8841 Functional Diagram.

STD Interface

The ZT 8841 interfaces with the STD bus when I/O decode logic detects an I/O address for one of the on-board UARTs or the Programmable Interrupt Controller (PIC). Data is written to or read from the UART or PIC as required by the control PAL and buffered by the data buffer. The control PAL also enables the data buffer if the PIC is to return an interrupt vector.

I/O Addressing

The ZT 8841 requires 40 noncontiguous I/O address ports. These I/O ports are decoded by the ZT 8841 I/O decode logic from either 8- or 16-bit STD I/O addresses as jumper selected. Eight I/O ports are assigned to each of the four UARTs and to the PIC. There are four jumperable I/O address configurations with STD address lines A0-9, as described on page 2-8. Address lines A10-15 can be jumpered for any desired I/O address, as described on page 2-5.

IOEXP can be ignored, decoded high, or decoded low to expand the I/O address map beyond the 64K limit. Default is to ignore IOEXP.

Wait-States

The ZT 8841 is equipped with a wait-state generator circuit for use with some high speed CPUs. One and two wait-states are jumper selectable to avoid unnecessary CPU system wait-states on every bus cycle by not forcing the CPU to insert wait-states. Refer to page A-8 for a description of the wait-state jumpers W14 and W15.

Serial Interface

Standard asynchronous communication for the four independent serial channels is accomplished by means of four programmable 82050 (8250 compatible) Universal Asynchronous Receiver Transmitter (UART) chips. These chips have individual on-board baud rate generators that operate from 50 to 56,000 baud, directly addressable status, control, and data registers, and fully programmable serial interface characteristics. Their programmable characteristics include 5-, 6-, 7-, or 8-bit character length, even or odd parity bit generation/detection, and 1, 1½, or 2 stop-bit generation.

Optionally, the ZT 8841 can be purchased with high performance 82510 UARTs that are 8250 compatible. These UARTs are designed for increased system efficiency by providing DC to 288 Kbaud, complete error detection, separated 4-byte receive and transmit FIFOs, control character recognition, and two independent 16-bit baud rate generators/timers. Enhanced features are configured by 35 control registers while maintaining emulation of the 8250 UARTs used in the IBM PC.

The enhanced serial communications option also includes MCS-51 support. The MCS-51 protocol developed by Intel can be used for multiprocessor communications in a master/slave(s) configuration. See pages 1-8 and 2-26 for more information on MCS-51 operation.

Each UART is selected by the STD I/O address decode logic. Individual UART registers are addressed by STD address lines. Once selected and programmed for operation, interrupts and serial communication can occur.

Interrupts

The following discussion of how interrupts are implemented on the ZT 8841 assumes that the previous "Interrupt Operation" section on page 3-8 has been read. If you have not read that section, please do so now.

Each UART has several conditions that can cause the interrupt output pin to go active. A summary is listed below.

- Receive Error or Break Occurred
- Data Available
- Transmit Date Ready
- Modem Status Change

> **Note:** The Modem Status Change enable bit actually enables interrupts from any of four sources: Clear To Send (CTS), Data Set Ready (DSR), Ring Indicator (RI), or Data Carrier Detect (DCD). Any transition on these input signals causes a Modem Status Change interrupt. Therefore, any inputs that are unused or are not to cause an interrupt must not be left floating (unconnected). Floating inputs can cause unwanted interrupts.

The COM1 and COM4 UART interrupt outputs are enabled by the OUT2 pin of each UART, as illustrated in Figure 3-8.

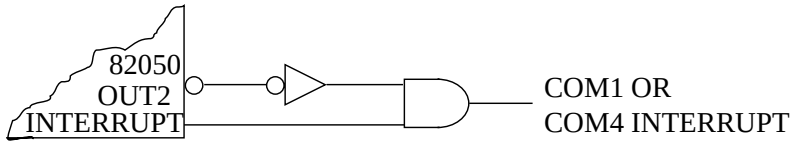


Figure 3–8. COM1, COM4 Interrupt Enable Circuit.

This circuit requires not only that the UART Interrupt Enable register be programmed for an interrupt to occur, but also that the OUT2 pin be enabled in the Modem Control register. This circuit is identical to the way in which interrupts must be enabled on an IBM serial adapter.

The circuit is slightly different for COM2 and COM3, as illustrated in Figure 3-9. OUT2 enables either the interrupt outputs or the RS-422/485 drivers. Do not use OUT2 to enable both interrupts and the RS-422/485 drivers on one COM port. In other words, do not simultaneously install W23 and W24 and do not simultaneously install W27 and W28.

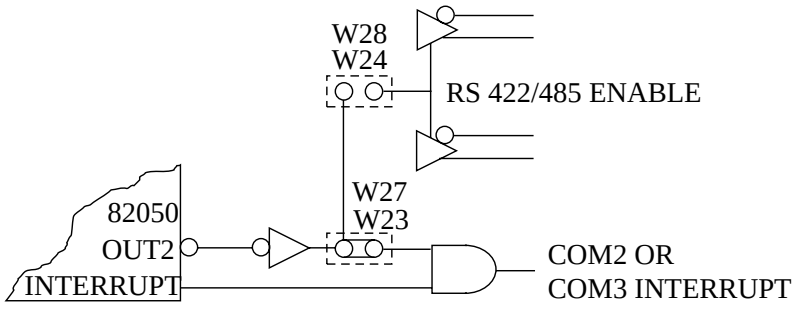


Figure 3–9. COM2, COM3 Interrupt Enable Circuit.

When W27 or W23 are installed (for COM2 and COM3, respectively), interrupts operate in the same manner as described above. When W28 or W24 are installed (COM2 and COM3, respectively), OUT2 no longer enables the UART interrupt. It is used to enable or disable the RS-422/485 drivers for multidrop networks.

With W27 or W23 removed, UART interrupts from COM2 or COM3, respectively, are always externally enabled. Use the UART Interrupt Enable register to mask the desired interrupts to be presented to the Programmable Interrupt Controller (PIC).

Interrupt outputs from COM1-4 are presented to an 8259A PIC, STD backplane interrupt signals on P1, and frontplane interrupt connector J5 (see Figure 3-10).

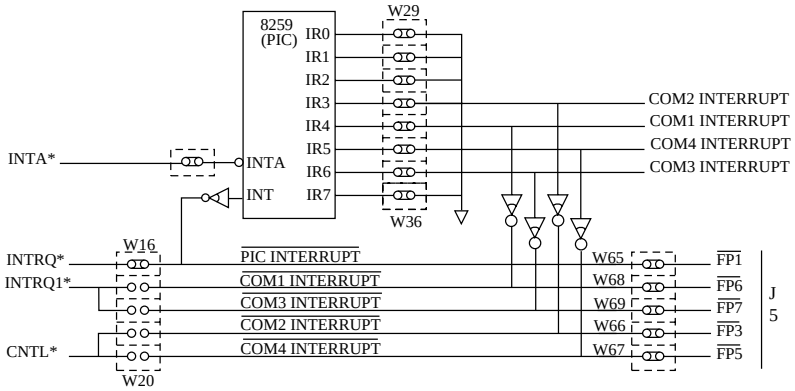


Figure 3–10. COM1-4 Interrupt Structure.

Interrupts generated by the UARTs can be arbitrated by an on-board 8259A PIC configured through software to act as a buffered slave to another master interrupt controller. When operating as a slave to another master, the 8259A cascade address is supplied to the ZT 8841 interrupt controller over STD bus address lines A8-10, per the *STD-80 Series Bus Specification.* This is the case when the STD CPU has an on-board interrupt controller such as the Ziatech ZT 8808 or ZT 8816.

COM1-4 interrupts presented to the PIC are routed through jumpers to various Interrupt Request (IR) inputs. The default jumper configuration is illustrated in Figure 3-10 on page 3-20. If a different assignment is desired, jumper posts W29-W36 can be used to wire-wrap any preferred configuration. The PIC interrupt output can be jumpered either to an STD backplane signal or to a frontplane interrupt signal.

If vectors from the on-board PIC are not desired, each of the COM1-4 interrupt outputs is available from open-collector, low-true buffers. Open-collector buffers can be logically wire-ORed together by wire-wrapping the desired outputs together for STD backplane or frontplane use. Any number of configurations are available with jumpers W16-20 or W65-69 for either STD backplane or frontplane use, respectively.

RS-232-C

Each channel may be easily configured for DTE (default) or DCE operation by means of on-board jumpers. Each channel terminates the following signals to a 14-pin male latching header:

- Transmitted Data (TxD)
- Received Data (RxD)
- Clear To Send (CTS)
- Request To Send (RTS)
- Data Set Ready (DSR)
- Data Terminal Ready (DTR)
- Ring Indicator (RI)
- Data Carrier Detect (DCD)

This arrangement allows easy connections to a flat cable 25-pin RS-232-C D-type connector without conductor scrambling or soldering. Each cable is 1 meter long and is terminated with a female D-type connector for DCE operation.

Refer to page B-10 for connector pin assignments for DCE or DTE operation.

RS-422/485

The RS-485 specification allows up to 32 receivers and 32 drivers on this multidrop network. To prevent more than one transmitter on the party line at any one time, a method for disabling the RS-485 transmitter through software control is necessary.

Two of the four channels may be configured for RS-422/485 operation by means of on-board jumpers. The RS-422/485 channels terminate the following signals to a 14-pin latching male header:

- Transmitted Data (TxD)
- Received Data (RxD)
- Clear To Send (CTS)
- Request To Send (RTS)

When configured for RS-422/485 operation, the operating temperature must be maintained between 0 and +65˚ Celsius because the drivers are not rated for extended temperatures.

The ZT 8841 allows the OUT2 line of the UART to be used for the enable and disable control of the RS-485 transmitter on COM2 and COM3 (see Figure 3-11).

Jumpers W24 and W28 must be installed to route the OUT2 signal to the transmitter. The transmitters are always enabled if these jumpers are removed.

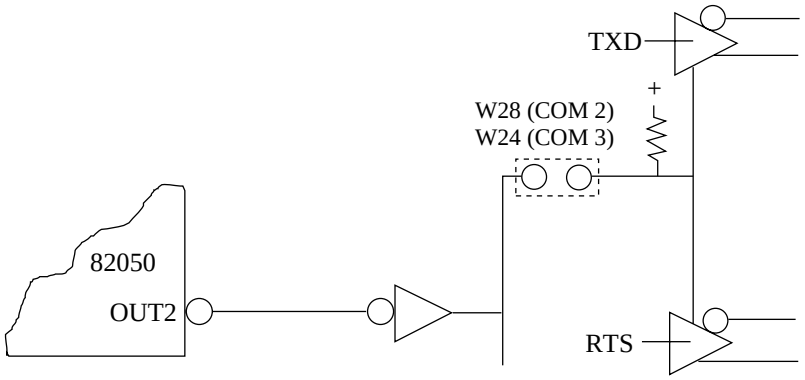


Figure 3–11. COM2, COM3 RS-485 Enable Circuit.

Low Power Operation: If RS-422/485 operation is not utilized and low power operation is desired, remove the RS-422/485 drivers from their sockets (locations 3B and 7B) and remove jumper W85 (W84 should be installed). Removing these parts saves a considerable amount of power. Another power-saving measure is to initialize the ZT 8841's PIC as a slave. The PIC will power up as a master and attempt to drive the cascade address lines. Additional power savings can be achieved by purchasing the ZT 8841 with the enhanced 82510 UARTs. These parts have a power-down mode controllable through software.

Optical Isolation: The ZT 8841 with the optical isolation option prevents up to 600 V spikes from destroying the STD bus computer. Isolation is accomplished only on the two RS-422/485 channels between the UARTs and drivers. This configuration does not prevent destruction of the RS-422/485 drivers, but does save the remaining ZT 8841 and STD bus circuitry. If the RS-422/485 drivers are destroyed, they are socketed for easy replacement at locations 3B and 7B.

Note that ZT 8841 boards with the optical isolation option must utilize both RS-422/485 channels (COM2 and COM3) in order to retain their optical isolation. Using COM2 or COM3 for RS-232-C with optical isolation on the remaining RS-422/485 channel defeats all the benefits of optical isolation.

If the enhanced UARTs are ordered (82510), the baud rate must not exceed 288 Kbaud to keep the optical isolation circuit from saturating. Saturation will cause intermittent operation.

Diagnostics: Diagnostics include data loopback, interrupt, and modem control test capability. Loopback testing is done by internally connecting the output of the transmitter shift register to the receiver shift register to verify the data paths of the UART. The interrupt system can be tested by first setting individual bits in the status register and then either handling the 8259A interrupt vector or checking status in the UART. The CTS and RTS modem control lines are tested by first internally connecting these two lines and then setting and resetting the RTS output and checking that the CTS input follows.

Chapter 6 describes how UART diagnostics are initiated. Refer to this chapter for more information.

Chapter 4

APPLICATION EXAMPLE

Contents **Page**

ZT 8841 SOFTWARE DRIVER 4-1
Objectives 4-1
Program Code 4-2

ZT 8841 SOFTWARE DRIVER

Objectives

- Write an 8088 or 80X86 initialization routine for the ZT 8841
- Write 8088 or 80X86 send and receive data routines for the ZT 8841
- Illustrate one way to address the ZT 8841 by means of a 16-bit I/O address

Program Code

```
PAGE ,132
;
;*********************************************************
;*                                                       *
;*               PROGRAMMING ABSTRACT                    *
;*                                                       *
;*********************************************************
;
;                    2/4/89
;                  900 HOURS
;                 ZIATECH CORP.
;               SAN LUIS OBISPO, CA
;
; THIS PROGRAMMING EXAMPLE IS FOR THE ZT 8841 QUAD UART
;    BOARD. INITIALIZATION, SEND AND RECEIVE ROUTINES
;    ARE ILLUSTRATED.
;
;*********************************************************
;*                                                       *
;*                   SYSTEM EQUATES                      *
;*                                                       *
;*********************************************************
;
;     * ZT 8841 STD I/O PORT ADDRESSES *
;
BASE            EQU     00H             ; FACTORY DEFAULT
UART1           EQU     BASE+3E0H       ; UART 1 ADDRESS
UART2           EQU     BASE+2E0H       ; UART 2 ADDRESS
UART3           EQU     BASE+3E8H       ; UART 3 ADDRESS
UART4           EQU     BASE+2E8H       ; UART 4 ADDRESS
PIC             EQU     BASE+0A8H       ; PIC BASE ADDRESS
;
;     * 82050 REGISTER EQUATES BY PORT ADDRESS *
;
;               REG #0 RECV./TRANS REG (R/W)
PORT_REC50      EQU     000H            ; RECV REGISTER
PORT_XMT50      EQU     000H            ; TRANS. REGISTER
PORT_DLALB      EQU     000H            ; IF DLAB=1, LSB DIV.
```

```
;               REG #1 INTERRUPT ENABLE REG (W)
PORT_INTEN      EQU     001H            ; INTERRUPT EN.
PORT_DLAMB      EQU     001H            ; IF DLAB=1, MSB DIV.
ERBI            EQU     01H             ; EN INTR ON RECV
EIRBI           EQU     01H             ; EN INTR ON RECV
EIRBO           EQU     02H             ; EN INTR ON XMT
ELSI            EQU     04H             ; EN LINE STATUS
EDSSI           EQU     08H             ; EN MODEM STATUS
;
;               REG # 2 INTERRUPT STATUS REG (R)
PORT_INSTAT     EQU     002H            ; INTERRUPT STATUS
INTPND          EQU     0FEH            ; INTERRUPT PENDING
RVLNS           EQU     06H             ; RECV LINE INTR
RECDA           EQU     04H             ; RECV DATA AVAIL
TXHRE           EQU     02H             ; XMTR. HOLD EMPTY
MODST           EQU     00H             ; MODEM STATUS INTR
;
;               REG # 3 LINE CONTROL REG (R/W)
PORT_LINEC      EQU     003H            ; LINE CONTROL REG
WL5             EQU     00H             ; 5 BIT WORD
WL6             EQU     01H             ; 6 BIT WORD
WL7             EQU     02H             ; 7 BIT WORD
WL8             EQU     03H             ; 8 BIT WORD
STB             EQU     04H             ; 2 STOP BITS
PEN             EQU     10H             ; PARITY ENABLE
EPS             EQU     20H             ; STICK PARITY
SBK             EQU     40H             ; SET BREAK
DLAB            EQU     80H             ; DIV LATCH ACCESS
;               DIVISOR LATCH VALUES (REG#3, DLAB=1)
BD111           EQU     1047            ; 110 BAUD COUNT
BD301           EQU     384             ; 300 BAUD COUNT
BD122           EQU     96              ; 1200 BAUD COUNT
BD242           EQU     48              ; 2400 BAUD COUNT
BD362           EQU     32              ; 3600 BAUD COUNT
BD482           EQU     24              ; 4800 BAUD COUNT
BD962           EQU     12              ; 9600 BAUD COUNT
BD193           EQU     6               ; 19.2 KBAUD COUNT
;
;               REG #4 MODEM CONTROL REG (W)
PORT_MODC       EQU     004H            ; MODEM CONTROL
DTR             EQU     01H             ; DATA TRML READY
RTS             EQU     02H             ; REQ TO SEND
OUT1            EQU     04H             ; OUTPUT TO LED
DIAG            EQU     10H             ; DIAGNOSTIC MODE
```

```
;               REG #5 LINE STATUS REG (R)
PORT_LINST      EQU     005H            ; LINE STATUS
DR              EQU     01H             ; DATA READY
ORE             EQU     02H             ; OVERRUN ERROR
PE              EQU     04H             ; PARITY ERROR
FE              EQU     80H             ; FRAMING ERROR
BKI             EQU     10H             ; BREAK INTR
THRE            EQU     20H             ; XMTR HOLD EMPTY
TSRE            EQU     40H             ; XMTR SHIFT EMPTY
;
;               REG #6 MODEM STATUS REG (R)
PORT_MODS       EQU     006H            ; MODEM STATUS
DCTS            EQU     01H             ; DELTA CTS
DDSR            EQU     02H             ; DELTA DSR
TERI            EQU     04H             ; TRAIL RING IND.
DSLSD           EQU     08H             ; DELTA RECV SIG.
CTS             EQU     10H             ; CLEAR TO SEND
DSR             EQU     20H             ; DATA SET READY
RI              EQU     40H             ; RING INDICATOR
RLSD            EQU     80H             ; RECV LINE DECT.
;
;*********************************************************
;*                                                       *
;*              MACRO DEFINITIONS                        *
;*                                                       *
;*********************************************************
;
GET     MACRO   SRC
        MOV     DX,UART         ;; GET UART
        ADD     DX,SRC          ;; DETERMINE REG
        IN      AL,DX           ;; GET DATA
        ENDM
;
PUT     MACRO   DST
        MOV     DX,UART         ;; GET UART
        ADD     DX,DST          ;; DETERMINE REG
        OUT     DX,AL           ;; PUT DATA
        ENDM
```

```
;*********************************************************
;*                                                       *
;*                    STACK SEGMENT                      *
;*                                                       *
;*********************************************************
;
; STACK SEGMENT IS LOCATED IN RAM FOR AN ARBITRARY STACK-
;    SIZE.
;
STACK                   SEGMENT         STACK
;
                DW      20 DUP (?)      ; UNINITIALIZED STACK
STACK_TOP       LABEL   WORD            ; OFFSET OF TOS
;
STACK                   ENDS
;
;*********************************************************
;*                                                       *
;*                     DATA SEGMENT                      *
;*                                                       *
;*********************************************************
;
DATA                    SEGMENT
;
UART            DW      ?               ; UART BASE
;
DATA                    ENDS
```

```
;*********************************************************
;*                                                       *
;*                    PROCEDURES                         *
;*                                                       *
;*********************************************************
;
CODE                    SEGMENT    PARA
;
ASSUME  CS:CODE,SS:STACK,DS:DATA,ES:NOTHING
;
;
INIT_UART               PROC
;
;    THIS PROCEDURE IS CALLED TO INITIALIZE A UART. THE
;           FOLLOWING PARAMETERS ARE INITIALIZED:
;
;           8-BIT CHARACTER LENGTH
;           1-START/STOP BIT
;           4800-BAUD
;           NO-PARITY
;           NO-INTERRUPTS
;
; INPUTS:   UART - BASE ADDRESS OF UART TO BE INIT
; OUTPUTS:  NONE
; CALLS:    NONE
; DESTROYS: AX,DX,F/FS
;
            MOV      AL,WL8+DLAB     ; 8-BIT, DLAB
            PUT      PORT_LINEC      ; OUTPUT LINE CONTROL
            MOV      AX,BD482        ; 4800 BAUD
            PUT      PORT_DLALB      ; WRITE LSB
            MOV      AL,AH           ; GET MSB
            PUT      PORT_DLAMB      ; WRITE MSB
            GET      PORT_LINEC      ; RESET DLAB
            AND      AL,7FH          ; ISOLATE DLAB MASK
            OUT      DX,AL           ; WRITE LINE CONTROL
            MOV      AL,0
            PUT      PORT_INTEN      ; ZERO INTERRUPTS
            RET
INIT_UART   ENDP
```

```
SEND_UART   PROC
;
; THIS PROCEDURE IS CALLED TO SEND A CHARACTER.
;
; INPUTS:   UART - BASE ADDRESS OF UART
;           AL - CHARACTER TO BE OUTPUT
; OUTPUTS:  NONE
; CALLS:    NONE
; DESTROYS: DX,F/FS
;
            PUSH     AX               ; SAVE CHAR
            MOV      DX,UART          ; GET UART
            ADD      DX,PORT_LINST    ; STATUS REGISTER
L1:
            IN       AL,DX            ; GET STATUS
            AND      AL,THRE          ; XMTR EMPTY?
            JZ       L1               ; NO, WAIT
            POP      AX               ; YES, RESTORE CHAR
            MOV      DX,UART          ; DATA REGISTER
            OUT      DX,AL            ; OUTPUT CHAR
            RET
SEND_UART   END;
;
RECV_UART   PROC; THIS PROCEDURE IS CALLED TO RECEIVE DATA.
;
; INPUTS:       UART - BASE ADDRESS OF UART
; OUTPUTS:      AL - CHARACTER RECEIVED
; CALLS:        NONE
; DESTROYS:     DX,F/FS
;
                MOV   DX,UART        ;GET UART
                ADD   DX,PORT_LINST  ;STATUS REGISTER
L2:             IN    AL,DX          ;GET STATUS
                AND   AL,DR          ;RCVR FULL?
                JZ    L2             ;NO, WAIT
                MOV   DX,UART        ;DATA REGISTER
                IN    AL,DX          ;INPUT CHAR
                RET
RECV_UART       ENDP
```

```
;*********************************************************
;*                                                       *
;*                    TEST  CODE                         *
;*                                                       *
;*********************************************************
;
; INITIALIZE SEGMENT REGISTER AND STACK POINTER.
;
START:
               MOV     AX,SEG DATA
               MOV     DS,AX
               MOV     AX,SEG STACK
               MOV     SS,AX
               MOV     SP,OFFSET STACK_TOP
;
; * INITIALIZE UART EXAMPLE *
;
               MOV     UART,UART1      ; SET BASE
               CALL    INIT_UART       ; INITIALIZE
;
; * SEND DATA BYTE EXAMPLE *
;
               MOV     AL,55H          ; SEND 55H
               CALL    SEND_UART       ; OUTPUT
;
; * RECEIVE DATA BYTE EXAMPLE *
;
               CALL    RECV_UART       ; INPUT BYTE
;
CODE  ENDS
      END      START
```

Chapter 5

INTERRUPT CONTROLLER (8259A)

Contents **Page**

INTRODUCTION TO THE INTERRUPT CONTROLLER 5-2
Interrupt Controller Operation 5-2
Interrupt Registers and Control Logic 5-6
OPERATION OF THE INTERRUPT CONTROLLER 5-8
Priorities 5-8
Fully Nested Mode 5-8
Special Fully Nested Mode 5-9
Automatic Rotating Mode 5-10
Specific Rotating Mode 5-10
Special Mask Mode 5-11
Poll Mode 5-11
Interrupt Triggering 5-11
Level-Triggered Mode 5-12
Edge-Triggered Mode 5-13
Interrupt Status 5-14
PROGRAMMING THE INTERRUPT CONTROLLER 5-16
End-Of-Interrupt (EOI) Formats 5-16
Programming With ICWs and OCWs 5-18
Initialization Command Words (ICWs) 5-19
ICW1 5-19
ICW2 5-21
ICW3 5-21
ICW4 5-22
Operation Control Words (OCWs) 5-24
OCW1 5-24
OCW2 5-24
OCW3 5-27

INTRODUCTION TO THE INTERRUPT CONTROLLER

The on-board slave 8259A programmable interrupt controller (PIC) handles up to eight vectored priority interrupts. It can expand the number of priority interrupt lines by cascading its interrupt output to a master PIC located on the 8088 or 80X86 CPU board.

Interrupt Controller Operation

Basic functions of the PIC are to determine the priority of an interrupt request, issue a single interrupt request to the 8088 or 80X86 based on that priority, and send a vectored restart address to the 8088 or 80X86 for servicing the interrupting device.

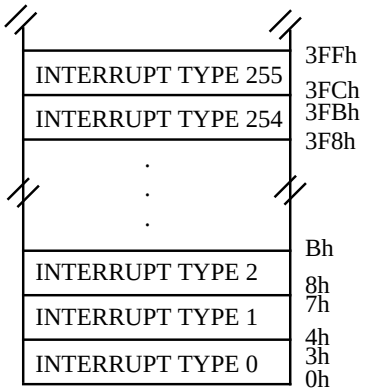
The 8088 or 80X86 enters its interrupt acknowledge cycle when an interrupt request is received and interrupts are enabled. The flag registers are pushed onto the stack (as in a PUSHF instruction) and the IF flag is cleared, disabling interrupts. The contents of both the code segment and the instruction pointer are then also pushed onto the stack.

The stack retains the pre-interrupt flag status and the pre-interrupt program location necessary to return from the service routine. The 8088 or 80X86 then issues the first of two INTA* pulses that signal the 8259A that the CPU board has honored its interrupt request.

The 8259A then vectors program execution to the corresponding service routine between the two INTA* pulses issued by the 8088 or 80X86. Unlike the 8080A or 8085A, the 8259A does not place on the data bus a CALL instruction and the starting address of the service routine. Instead, the first INTA* pulse is used only to signal the 8259A of the honored request. The second INTA* pulse causes the 8259A to place a single interrupt vector byte onto the data bus. Not used as a direct address, this interrupt vector byte pertains to one of 256 interrupt "types" supported by the 8088 or 80X86 memory. Program execution is vectored to the corresponding service routine by the contents of a specified interrupt type.

All 256 interrupt types are located in the absolute memory locations 0 through 3FFh, which make up the interrupt vector table of the 8088 or 80X86 (see Table 5-1 below). Each type in the interrupt vector table stores a code segment address and an instruction pointer address, requiring four bytes of memory. Locations 0 through 3FFh should be reserved for the interrupt vector table alone. Memory locations 00 through 7Fh (types 0-31) are reserved for use by Intel Corporation for Intel hardware and software products. To maintain compatibility with present and future Intel products, these locations should not be used.

Table 5-1
8088 or 80X86 Interrupt Vector.



When the 8088 or 80X86 receives an interrupt-vector byte from the 8259A, it multiplies its value by four to acquire the address of the interrupt type. For example, if the interrupt-vector byte specifies a type of 128 (80h), the vectored address in 8088 or 80X86 memory is 4 x 80h, which equals 200h. Program execution is then vectored to the service routine whose address is specified by the code segment and the instruction pointer values within type 128 located at 200h.

When entering an interrupt service routine, you should save those registers mutually used by the main program and the service routine. The best way to do this is to "PUSH" each register used onto the stack immediately. The service routine can then "POP" each register off the stack in reverse order when complete.

After the service routine is completed, you may reenter the main program by using an IRET (Interrupt Return) instruction, which pops the pre-interrupt instruction pointer, the code segment, and the flags off the stack. The main program resumes where it was interrupted. Flag status, including that of the IF flag, remains the same regardless of changes in the service routine. Interrupts are thus re-enabled automatically upon return from the service routine.

In addition to generating external interrupts from the INTR pin, the 8088 or 80X86 can also invoke interrupts via software. Three interrupt instructions are provided: INT, INT (Type 3), and INTO. INT is a two-byte instruction; the second byte selects the interrupt type. INT (Type 3) is a one-byte instruction that selects interrupt Type 3. INTO is a conditional one-byte interrupt instruction that selects interrupt Type 4 if the OF flag (trap on overflow) is set. All software interrupts vector program execution in the same manner as do hardware interrupts.

An 8259A block diagram is shown on page 5-5. The 8259A consists of eight major blocks: the Interrupt Request register (IRR), the In-Service register (ISR), the Interrupt Mask register (IMR), the Priority Resolver (PR), the cascade buffer/comparator, the data bus buffer, and logic blocks for both control and read/write. The following section discusses the blocks directly related to interrupt handling: the IRR, ISR, IMR, PR, and the control logic.

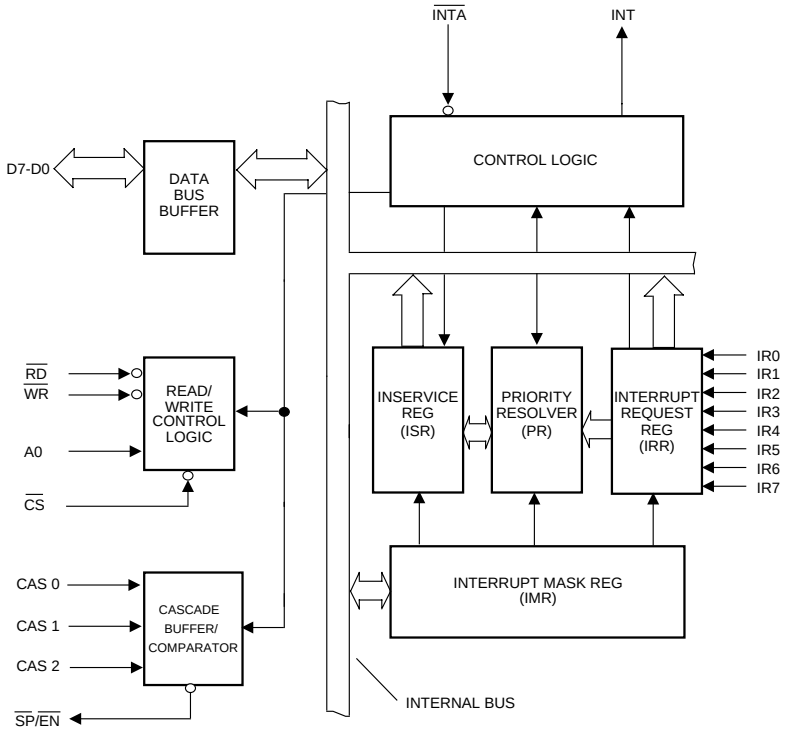


Figure 5–1. 8259A Block Diagram.

Interrupt Registers and Control Logic

Interrupt requests are handled by three "cascaded" registers. The Interrupt Request register (IRR) stores all interrupt levels requesting service; the In-Service register (ISR) stores all levels being serviced; and the Interrupt Mask register (IMR) stores the bits of the interrupt lines to be masked. The Priority Resolver (PR) looks at the IRR, the ISR, and the IMR, and determines if an INT should be issued by the control logic to the processor.

When an interrupt occurs on the IR input, it propagates through the request latch and thence to the PR (assuming the input is not masked). The PR looks at the incoming requests and the currently in-service interrupts to ascertain whether an interrupt should be issued to the processor.

Assuming the request is the only one incoming and no requests are presently in service, the PR then causes the control logic to assert the INT line to the processor (high). When the processor honors the INT pulse, it sends out a sequence of INTA* pulses to the 8259A. During this sequence, the state of the request latch is frozen. Priority is again resolved by the PR to determine the appropriate interrupt vectoring, which is then conveyed to the processor via the data bus.

Immediately following the interrupt acknowledge sequence, the PR sets the corresponding bit in the ISR, which clears the edge sense latch. If edge sensitive triggering is used, clearing the edge sense latch also disarms the request latch. This inhibits a still active IR input from propagating through the priority resolver. The IR input must return to an inactive state, setting the edge sense latch, before another interrupt request can be recognized. If level sensitive triggering is used, however, clearing the edge sense latch does not affect the request latch. The state of the request latch is entirely dependent upon the IR input level. Another interrupt is generated immediately if the IR level is left active after its ISR bit has been reset. An ISR bit is reset with an End Of Interrupt (EOI) command issued in the service routine. End Of Interrupt commands are discussed in greater detail on page 5-16.

Disabling or enabling interrupts can be done by means other than control of the microprocessor's interrupt request pin. The 8259A's IMR (Interrupt Mask register) enhances interrupt control capabilities by allowing individual masking rather than requiring all interrupts to be disabled or enabled at the same time. The IMR is an 8-bit register; bits 0-7 directly correspond to IR0-IR7. Any IR input can be masked by writing to the IMR and setting the appropriate bit. Similarly, any IR input can be enabled by clearing the correct IMR bit.

One example of the usefulness of masking off individual IR inputs is if you wish a portion of a main routine to be interrupted only by specific interrupts. Another might be the capability of disabling higher priority interrupts for a portion of a lower priority service routine. The possibilities are many.

When an interrupt occurs while its IMR bit is set, it is not necessarily forgotten; as stated earlier, the IMR acts only on the output of the IRR. Even with an IR input masked it is still possible to set the IRR. Thus, when resetting an IMR, if its IRR bit is set it will then generate an interrupt. This is providing, of course, that other priority factors are taken into consideration and that the IR request remains active. If the IR request is removed before the IMR is reset, no interrupt will be acknowledged.

OPERATION OF THE INTERRUPT CONTROLLER

Interrupt operation of the 8259A falls under three categories: priorities, triggering, and status. Various modes and commands used in each of these categories are discussed below; programming considerations are discussed later in this chapter (see page 5-18). The 8259A data sheet and AP-59 from Intel are helpful aids.

Priorities

The 8259A can be programmed to operate in one of the following modes:

- Fully Nested Mode
- Special Fully Nested Mode
- Automatic Rotating Mode
- Specific Rotating Mode
- Special Mask Mode
- Poll Mode

Fully Nested Mode

In this mode, PIC input signals are assigned a priority from 0 through 7, with IR0 having the highest priority and IR7 the lowest. This is the normal operation mode of the PIC unless specifically programmed otherwise.

When an interrupt is acknowledged, the highest priority request is available to the CPU. Lower priority interrupts are inhibited; higher priority interrupts generate an interrupt that is acknowledged if the 8088 or 80X86 has enabled its own interrupt. The End-Of-Interrupt (EOI) command from the CPU is required to reset the PIC for the next interrupt.

Special Fully Nested Mode

This mode is used only when the master PIC on the CPU board is cascaded to the ZT 8841 slave PIC. In the cascade mode, if a slave receives a higher priority interrupt request than one that is in service it will not be recognized by the master. Because the master's ISR bit is set, all requests of equal or lower priority are ignored. Higher priority slave interrupts are not serviced in this case until after the master's ISR bit is reset by an EOI command, normally after the completion of the lower priority routine.

The special fully nested mode should be used for a true fully nested structure within a slave 8259A. This mode is programmed in the master only during initialization. In this mode, the master ignores only those interrupt requests of lower priority than that of the set ISR bit and responds to all requests of equal or higher priority. Thus, if a slave receives a higher priority request than the one in service, it is recognized, and the master PIC initiates another interrupt to the 8088 or 80X86.

When exiting the interrupt service routine, the software must check to determine whether another interrupt is pending in the slave before issuing an EOI command to the master. This is done by resetting the appropriate slave ISR bit with an EOI and then reading its ISR. If the ISR contains all zeros, there are no other interrupts from the slave in service, and an EOI command can be sent to the master. If the ISR contains no zeros, an EOI command should not be sent to the master.

Automatic Rotating Mode

In this mode, interrupt priority rotates. Once an interrupt on a given input is serviced, that interrupt assumes the lowest priority. If a number of simultaneous interrupts occur, priority rotates among the interrupts in numerical order. For example, if interrupts IR4 and IR6 request service simultaneously, IR4 will receive the highest priority. After service, the priority level rotates so that IR4 has the lowest priority and IR5 the highest. In the worst case, seven other interrupts are serviced before IR4 again has the highest priority. If IR4 is the only request, it is, of course, serviced promptly. Priority shifts when the PIC receives an End-of-Interrupt (EOI) command.

Specific Rotating Mode

In this mode, the software can change interrupt priority by specifying the lowest priority, which automatically sets the highest. For example, if IR5 is assigned the lowest priority, IR6 assumes the highest. In specific rotating mode, priority can be rotated by writing a Specific Rotate EOI (SEOI) command to the PIC. This command contains the binary coded decimal (BCD) code of the interrupt being serviced; that interrupt is reset as the lowest priority. In addition, the lowest priority interrupt can be fixed at any time by writing a command word to the appropriate PIC.

Special Mask Mode

One or more of the eight interrupt request inputs can be individually masked during the PIC initialization or at any subsequent time. If an interrupt is masked during servicing, lower priority interrupts are inhibited.

There are two ways to enable the lower priority interrupts:

a) Write an End-Of-Interrupt (EOI) command.

b) Set the Special Mask Mode.

The Special Mask Mode is useful when one or more interrupts are masked. If for any reason an input is masked during servicing, lower priority interrupts are disabled. It is possible, however, to enable the lower priority interrupt with the Special Mask Mode. In this mode, the lower priority lines are enabled until the Special Mask Mode is reset. Higher priorities are not affected.

Poll Mode

In this mode, the CPU internal interrupt enable flip-flop is clear (interrupts disabled) and a software subroutine is used to initiate a poll command. In the Poll Mode, the addressed PIC treats an I/O read command as an interrupt acknowledge, sets its in-service flip-flop if there is a pending interrupt request, and reads the priority level. This mode is useful if there is a common service routine for several devices.

Interrupt Triggering

Two classical ways of sensing an active interrupt request are use of a level sensitive input or an edge sensitive input. The 8259A lets you use either method with the edge-triggered mode and the level-triggered mode. Selection of one of these interrupt triggering methods is done during programmed initialization of the 8259A.

Level-Triggered Mode

When in the level-triggered mode, the 8259A recognizes any active (high) level on an IR input as an interrupt request. If the IR input remains active after an EOI command has been issued (resetting its ISR bit), another interrupt is generated. This occurs if the processor INT pin is enabled. Unless repetitious interrupt generation is desired, the IR input must be brought to an inactive state before an EOI command is issued in its service routine. Note that the request on the IR input must remain until after the falling edge of the first INTA* pulse. If, on any IR input, the request goes inactive before the first INTA* pulse, the 8259A responds as if IR7 were active. In any design in which there is a possibility of this occurring, the IR7 default feature can be used as a safeguard. You can do this by using the IR7 routine as a "cleanup routine" to recheck the 8259A status or to merely return program execution to its pre-interrupt location.

The level-triggered mode provides for repetitious interrupt generation. This can be useful when a service routine must be continually executed until the interrupt request goes inactive.

Caution should be exercised when using the automatic EOI mode and the level-triggered mode together. In the automatic EOI mode, an EOI is automatically performed at the end of the interrupt acknowledge sequence. If the processor enables interrupts while an IR input is still high, an interrupt will occur immediately. To avoid this, interrupts should be kept disabled until the end of the service routine or until the IR input returns low.

Edge-Triggered Mode

In the edge-triggered mode, the 8259A will recognize interrupts only if generated by an inactive (low) to active (high) transition on an IR input. The edge-triggered mode incorporates an edge-lockout method of operation. This means that after the rising edge of an interrupt request and the acknowledgement of the request, the positive level of the IR input will not generate further interrupts on this level. You need not worry about quickly removing the request after acknowledgement to avoid generating further interrupts, as might be the case in the level-triggered mode. Before another interrupt can be generated, the IR input must return to the inactive state.

Because of its edge-lockout operation, the edge-triggered mode is best used in those applications in which repetitious interrupt generation is not desired. It is also very useful in systems in which the interrupt request is a pulse (this should take the form of a negative pulse to the 8259A). Another advantage is that it can be used with the automatic EOI mode without the cautions required for the level-triggered mode. The edge-triggered mode simplifies operation in most cases because the duration of the interrupt request at a positive level is not usually a factor.

Interrupt Status

You can interrogate 8259A interrupt status by allowing the internal interrupt registers to be read via software control. This can be useful for interrupt control during service routines. The poll command also provides a method of device monitoring, offering an alternative to the interrupt vector method for those cases requiring more than 64 interrupts.

The contents of the Interrupt Request register, the In-Service register, and the Interrupt Mask register can be read to update your program regarding the present status of the 8259A. This versatile tool in the decision making process of a service routine gives you more control over interrupt operations. Before examining the process of reading the registers, a brief review is in order:

IRR (Interrupt Request Register)	Specifies all interrupts requesting service.
ISR (In-Service Register)	Specifies all interrupt levels being serviced.
IMR (Interrupt Mask Register)	Specifies all interrupt levels that are masked.

To read the contents of the IRR or ISR, first issue the appropriate read register command (read IRR or read ISR) to the 8259A. Apply an RD* pulse (an input instruction) to the 8259A to acquire the contents of the desired register. You need not issue a read register command every time the IRR or ISR is to be read. When the 8259A receives a read register command, it "remembers" which register has been selected. All that is necessary to read the contents of the same register more than once is the RD* pulse and the correct addressing (A0 = 0, explained on the following pages). Upon initialization, the selection of registers defaults to the IRR.

Caution should be exercised when using the read register command in a system that supports several levels of interrupts. If the higher priority routine causes an interrupt between the read register command and the actual input of the register contents, the same register will not automatically be selected upon its return. It is best in such cases to disable interrupts during the operation.

Reading the contents of the IMR differs from reading the IRR or the ISR. Because the IMR can be addressed directly for both reading and writing, a read register command is not necessary. The 8259A requires only an RD* pulse and the correct addressing (A0 = 1, explained on the following pages).

PROGRAMMING THE INTERRUPT CONTROLLER

End-Of-Interrupt (EOI) Formats

You must notify the 8259A when an interrupt service routine is complete. The ISR can then be updated to keep track of interrupt levels being serviced and their relative priorities. Three different End-Of-Interrupt (EOI) formats are available for your use: the non-specific EOI command, the specific EOI command, and the automatic EOI mode. Selection of which to use is dependent upon the interrupt operations you wish to perform.

A **non-specific EOI command** sent from the microprocessor notifies the 8259A that a service routine has been completed without specifying its exact interrupt level. The 8259A automatically determines the interrupt level and resets the correct bit in the ISR.

To utilize the non-specific EOI, the 8259A must operate in a mode in which it can predetermine in-service routine levels. For this reason the non-specific EOI command should be used only when the most recent level acknowledged and serviced is always the highest priority level. When the 8259A receives a non-specific EOI command, it simply resets the highest priority ISR bit, thus confirming to the 8259A that the highest priority routine of the routines in service is finished.

The primary advantage of using the non-specific EOI command is that IR level specification is not necessary. However, special care should be taken when using the non-specific EOI. Specified below are two program conditions in which it is best not used:

- Using the set priority command within an interrupt service routine
- Using a special mask mode

A **specific EOI command** sent from the microprocessor lets the 8259A know when a service routine of a particular interrupt level is completed. Unlike a non-specific EOI command, which automatically resets the highest priority ISR bit, a specific EOI command specifies that an exact ISR bit be reset. One of the eight IR levels of the 8259A can be specified in the command.

The specific EOI command is needed to reset the ISR bit of a completed service routine whenever the 8259A is not able to determine it automatically. An example of this situation is if the priorities of the interrupt levels are changed during an interrupt routine ("Specific Rotation"). In this case, if any other routines are in service at the same time, a non-specific EOI might reset the wrong ISR bit. The specific EOI command is the best choice in this case, or in any case in which confusion of interrupt priorities may exist. You can use the specific EOI command in all conditions of 8259A operation, including those that prohibit non-specific EOI command usage.

When programmed in the **automatic EOI mode**, the microprocessor no longer needs to issue a command to notify the 8259A that it has completed an interrupt routine. The 8259A performs a non-specific EOI automatically at the trailing edge of the last INTA* pulse (the second pulse in 8088 or 80X86 systems).

The obvious advantage of the automatic EOI mode over the other EOI command is that no command has to be issued. This simplifies programming and lowers code requirements within interrupt routines.

Take special care, however, when deciding to use the automatic EOI mode because it disturbs the fully-nested mode. In the automatic EOI mode the ISR bit of a routine in service is reset immediately after it is acknowledged, thus leaving no designation in the ISR that a service routine is being executed. If any interrupt request occurs during this time (and interrupts are enabled), it will be serviced regardless of its priority, low or high. The problem of "over-nesting" may also occur in this situation. Over-nesting occurs when an IR input continually interrupts its own routine, resulting in unnecessary stack pushes that fill the stack in a worst case condition.

Selection of the automatic EOI mode is dependent upon the application. If interrupts are controlled at a predetermined rate, preventing the problems mentioned above, the automatic EOI mode is useful. If interrupts occur sporadically at an indeterminate rate, however, the automatic EOI mode should be used only if the following guidelines are utilized.

When using the automatic EOI mode with an indeterminate interrupt rate, the microprocessor should keep its interrupt request input disabled during execution of service routines. Higher priority interrupt levels will thus be serviced only after the completion of a routine in service. This guideline restores the fully-nested structure in regard to the IRR; a routine in-service, however, cannot be interrupted.

Programming With ICWs and OCWs

Programming the 8259A is accomplished using two types of command words: Initialization Command Words (ICWs) and Operation Control Words (OCWs). All modes and commands explained earlier in this chapter (see page 5-8) are programmable using ICWs and OCWs. ICWs are issued from the processor in a sequential format and are used to set up the 8259A in an initial state of operation. OCWs are issued as needed to vary and control 8259A operation.

Both ICWs and OCWs are sent by the processor to the 8259A via the data bus (8259A CS*=0, WR*=0). The 8259A distinguishes between the different ICWs and OCWs by the state of its A0 pin (controlled by processor addressing), the sequence in which the ICWs are issued, and some dedicated bits among the ICWs and OCWs. Those bits which are dedicated are so indicated by fixed values (0 or 1) in the corresponding ICW or OCW programming formats discussed below. When issuing either ICWs or OCWs, the interrupt request pin of the processor should be disabled. Note that ICW and OCW descriptions

are given for both a master PIC residing on the CPU and a slave PIC residing on the ZT 8841. This has been done to facilitate programming each 8259A in the system for master or slave modes of operation.

Initialization Command Words (ICWs)

Before normal operation can begin, each 8259A in a system must be initialized by a sequence of three to four programming bytes called ICWs (Initialization Command Words). ICWs are used to set up the conditions and modes (master or slave) necessary for proper 8259A operation. Initialization Command Words (ICWs) are shown in Figure 5-2 on page 5-20. Note that ICW and OCW descriptions given in the following sections are for both a master PIC residing on the CPU and a slave PIC residing on the ZT 8841. This has been done to facilitate programming each 8259A in the system for master or slave modes of operation.

ICW1

The first Initialization Command Word (ICW1), required in all modes of operation, consists of the following:

a) Bits 0 and 4 are both 1 and identify the word as ICW1 for an 8088 or 80X86 CPU operation.

b) Bit 1 denotes whether or not the PIC is employed in a multiple PIC configuration. Code bit 1 = 1 for a master PIC if interfaced to a slave PIC on the ZT 8841. Code bit 1 = 1 if the ZT 8841 slave PIC is interfaced to the master PIC via the STD bus.

c) Bits 2, 5, 6, and 7 are "don’t care" and are normally coded as 0.

d) Bit 3 establishes whether the interrupts are requested by a low-to-high transition input. This applies to all input requests handled by the PIC. If bit 3 = 1, a low-to-high transition is required to request an interrupt on any of the eight levels handled by the PIC.

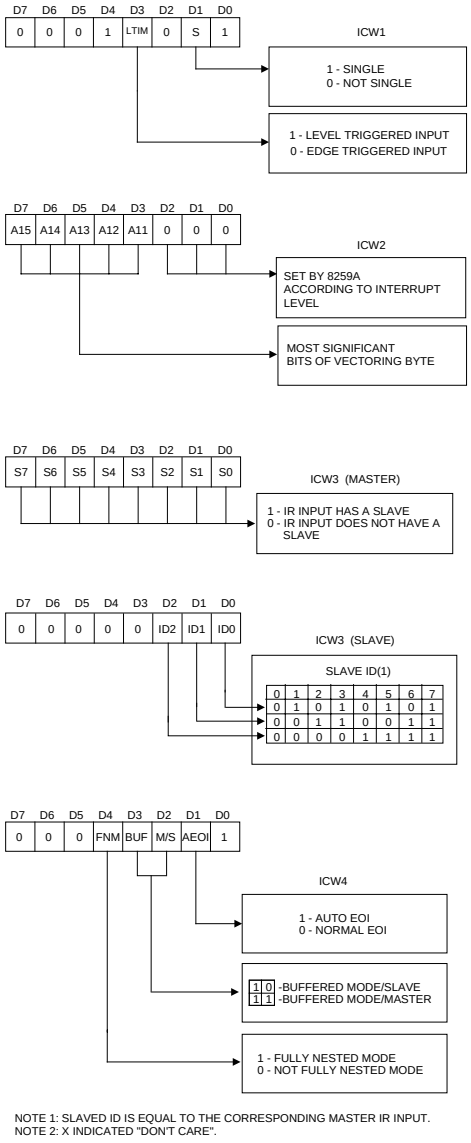


Figure 5–2. 8259A Initialization Command Word Formats.

ICW2

The second Initialization Command Word (ICW2), also required in all modes of operation, consists of the following:

a) For programming as a master PIC with slaves on **all** inputs, write 00h in ICW2. Although in this case ICW2 conveys no information, it is required to prepare the master PIC for either ICW3 or ICW4 (or both) to follow.

b) For programming as a master with slaves on some inputs, write ICW2 to establish the base vector address table in memory. Bits D0-D7 specify the interrupt type desired (0-255). The address of the vector table is obtained by multiplying the type by four. For example, to use the last eight vector types (248-255), program bits D3-D7 with 11111B. The three least significant bits of the vector determine which unique vector in the range 248-255 is selected for interrupt inputs generated on the master by non-slaves. Any interrupt inputs on the master connected to a slave will not generate vector for that corresponding type.

c) For programming as a slave, the operation is similar to b) above except that a different ICW2 must be used to prevent overlapping the slave's vector address table with that of the master.

ICW3

The third Initialization Command Word (ICW3) is required only if bit 1 = 0 in ICW1, specifying that multiple PICs are used (one or more PICs are slaves to the on-board master PIC). The S0-S7 bits correspond to the IR0-IR7 bits of the master PIC. For example, if a slave PIC is connected to the master PIC IR3 input, code bit 3 = 1. The ID0 - ID2 bits are used by the slave to establish a unique identification. The ID code of a particular slave must correspond to the master IR input to which it is connected. For example, if a slave is connected to IR6 on the master, the slave ID0 - ID2 bits should be set to ID0 = 0, ID1 = 1, and ID2 = 1.

ICW4

The fourth Initialization Command Word (ICW4), required for all modes of operation, consists of the following:

a) Bits 0 and 3 are both 1 to identify that the word is ICW4 for an 8088 or 80X86 CPU and that the hardware is configured for buffered operation.

b) Bit 1 programs the End-Of-Interrupt (EOI) function. Code bit 1 = 1 if an EOI is to be automatically executed (hardware). Code bit 1 = 0 if an EOI command is to be generated by software before returning from the service routine.

c) Bit 2 specifies whether ICW4 is addressed to a master PIC or to a slave PIC. For example, code bit 2 = 1 in ICW4 for the master PIC and 0 for the slave PIC.

d) Bit 4 programs the nested or fully-nested mode.

In summary, three or four ICWs are required to initialize the master and each slave PIC. Specifically:

Master PIC - No Slaves:

ICW1, ICW2, ICW4

Master PIC - With Slave(s):

ICW1, ICW2, ICW3, ICW4

Each Slave PIC:

ICW1, ICW2, ICW3, ICW4

To initialize the PICs (master and slave), proceed as follows:

a) Disable system interrupts by executing a CLI (Clear Interrupt Flag) instruction.

b) Initialize master PIC by writing ICWs in the following sequence:

   1. Write ICW1 and ICW2.
   2. If slave PICs are used, write ICW3 and ICW4. If no slave PICs are used, omit ICW3 and write ICW4 only.

c) Initialize each slave PIC by writing ICWs in the following sequence: ICW1, ICW2, ICW4.

d) Enable system interrupts by executing an STI (Set Interrupt Flag) instruction.

Operation Control Words (OCWs)

After initialization, the master and slave PICs can be programmed at any time for various operating modes (master or slave). Operation Control Word (OCW) formats are shown in Figure 5-3 on page 5-25.

Three OCWs are available for programming various modes and commands. Unlike the ICWs, the OCWs do not have to be in any type of sequential order. Rather, they are issued by the processor as needed within a program.

OCW1

OCW1 is used solely for 8259A masking operations. It provides a direct link to the IMR (Interrupt Mask register). The processor can write to or read from the IMR via OCW1. The OCW1 bit definition is as follows:

M0-M7:

The M0-M7 bits are used to control the masking of IR inputs. If an M bit is set to 1, it will mask (disable) the corresponding IR input. A "0" clears the mask, thus enabling the IR input. These bits convey the same meaning when being read by the processor for status update.

OCW2

OCW2 is used for End-Of-Interrupt, automatic rotation, and specific rotation operations. Associated commands and modes of these operations, with the exception of AEOI initialization, are selected using the bits of OCW2 in a combined fashion. Selection of a command or mode should be made with the corresponding table for OCW2 in the OCW programming format rather than on a bit-by-bit basis. However, for completeness of explanation, bit definition of OCW2 is as follows:

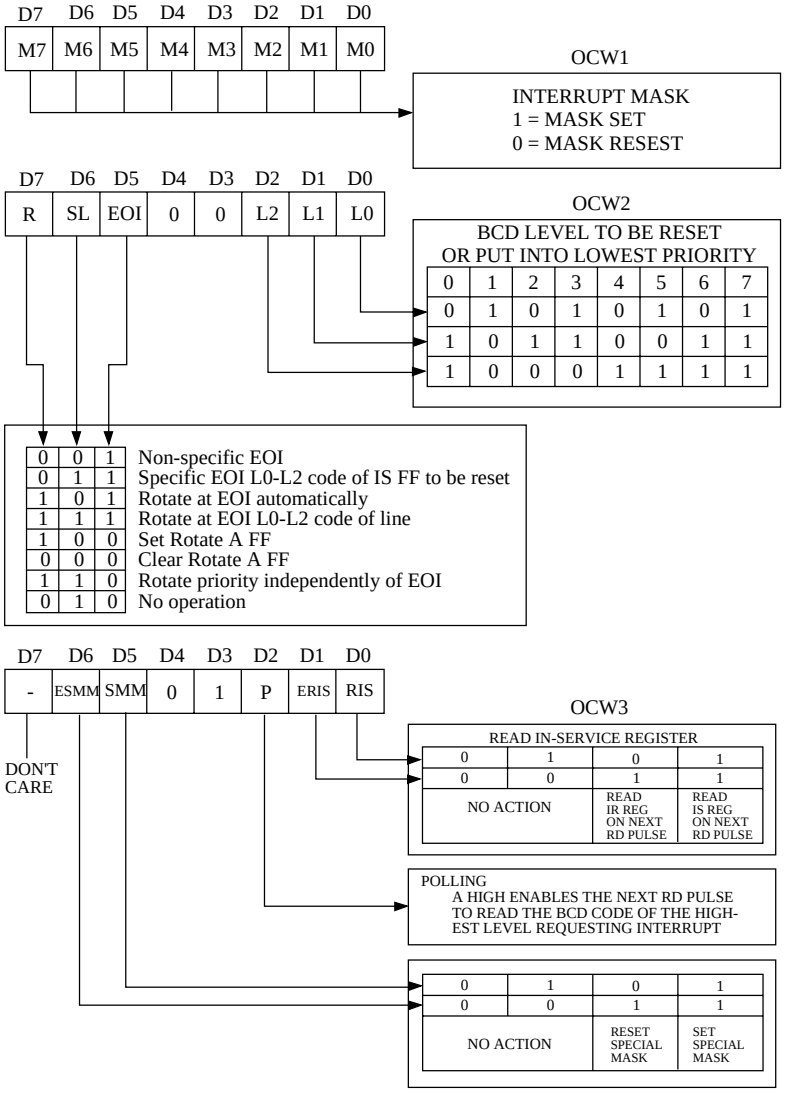


Figure 5–3. 8259A Operation Control Word Formats.

L0-L2:

The L0-L2 bits are used to designate an interrupt level (0-7) to be acted

upon for the operation selected by the EOI, SL, and R bits of OCW2. The level designated will be used either to reset a specific ISR bit or to set a specific priority. The L0-L2 bits are enabled or disabled by the SL bit.

EOI:

The EOI bit is used for all End-Of-Interrupt commands (not an automatic End-Of-Interrupt mode). If set to 1, a form of an End-Of-Interrupt command will be executed depending on the state of the SL and R bits. If EOI is 0, an End-Of-Interrupt command will not be executed.

SL:

The SL bit is used to select a specific level for a given operation. If SL is set to 1, the L0-L2 bits are enabled. The operation selected by the EOI and R bits will be executed on the specified interrupt level. If SL is 0, the L0-L2 bits are disabled.

R:

The R bit is used to control all 8259A rotation operations. If the R bit is set to 1, a form of priority rotation will be executed depending on the state of SL and EOI bits. If R is 0, rotation will not be executed.

OCW3

OCW3 is used to issue various modes and commands to the 8259A. There are two main categories of operation associated with OCW3: interrupt status and interrupt masking. Bit definition of OCW3 is as follows:

RIS:

The RIS bit is used to select the ISR or IRR for the read register command. If RIS is set to 1, ISR is elected. If RIS is 0, IRR is selected. The state of the RIS is honored only if the R bit is 1.

ERIS:

The ERIS bit is used to execute the read register command. If ERIS is set to 1, the read register command is issued and the state of RIS determines the register to be read. If ERIS is 0, the read register command is not issued.

P:

The P bit is used to issue the poll command. If P is set to 1, the poll command is issued. If it is 0, the poll command is not issued. The poll command will override a read register command if set simultaneously.

SMM:

The SMM bit is used to set the special mask mode. If SMM is set to 1, the special mask mode is selected. If it is 0, it is not selected. The state of the SMM bit is honored only if it is enabled by the ESMM bit.

ESMM:

The ESMM bit is used to enable or disable the effect of the SMM bit. If ESMM is set to 1, SMM is enabled. If ESMM is 0, SMM is disabled. This bit is useful to prevent interference of mode and command selections in OCW3.

Chapter 6

SERIAL COMMUNICATIONS (82050)

Contents **Page**

OVERVIEW 6-2
ZT 8841 SPECIFICS 6-3
FUNCTIONAL DESCRIPTION 6-5
- Read/Write Control 6-5
- Receiver 6-5
- Transmitter 6-5
- Modem Control 6-6
- Interrupt Control 6-6

PROGRAMMABLE REGISTERS 6-7
- Transmit and Receive Buffer 6-8
- Line Control 6-8
- Line Status 6-11
- Modem Control 6-13
- Modem Status 6-16
- Divisor Latch 6-18
- Interrupt Identify 6-19
- Interrupt Enable 6-20

OPERATION 6-21
- Reset 6-21
- Serial Data Format 6-22
- Baud Rate 6-23
- Interrupt and Polled Communication 6-25
- RS-422/485 Serial Communications 6-25
- Programming 6-26

OVERVIEW

The Asynchronous Communication Controller (ACC) is the Intel 82050, or equivalent. The ACC is a single asynchronous serial channel that performs serial communication between the ZT 8841 and an external serial device. Major features of the ACC are listed below.

- Full-duplex asynchronous operation
- Programmable baud rates up to 56 Kbaud
- Programmable character format
- Automatic break detect and handling
- Parity, overrun, and framing detection
- Interrupt and polled operation
- Loopback diagnostics
- Double buffering

ZT 8841 SPECIFICS

The four ACC COM serial ports are available at connectors J1-J4. They include Transmit Data (TxD), Receive Data (RxD), and support for a complete selection of the following Modem Control functions:

- Clear To Send (CTS)
- Request To Send (RTS)
- Data Terminal Ready (DTR)
- Data Set Ready (DSR)
- Ring Indicator (RI)
- Data Carrier Detect (DCD)

Each COM serial port is jumper configured to operate as Data Communication Equipment (DCE) or Data Terminal Equipment (DTE) for RS-232-C operation. RS-232-C jumper assignments for connectors J1-4 (COM1-4) are shown for DTE (default) on page A-11 and for DCE on page A-18. RS-422/485 jumper assignments for connectors J2-3 (COM 2-3) are shown on page A-11.

The ACC supports interrupt-driven communications. The interrupt output of each COM serial port is connected to the IR inputs of the 8259A programmable interrupt controller (PIC) as illustrated in Figure 3-10 on page 3-20.

When COM2 (J2) or COM3 (J3) are jumper configured to operate in RS-422/485 mode, the OUT2 bit of the ACC Modem Control register can be jumpered to enable and disable the transmit and the request to send buffers (see pages 6-13 and 3-18).

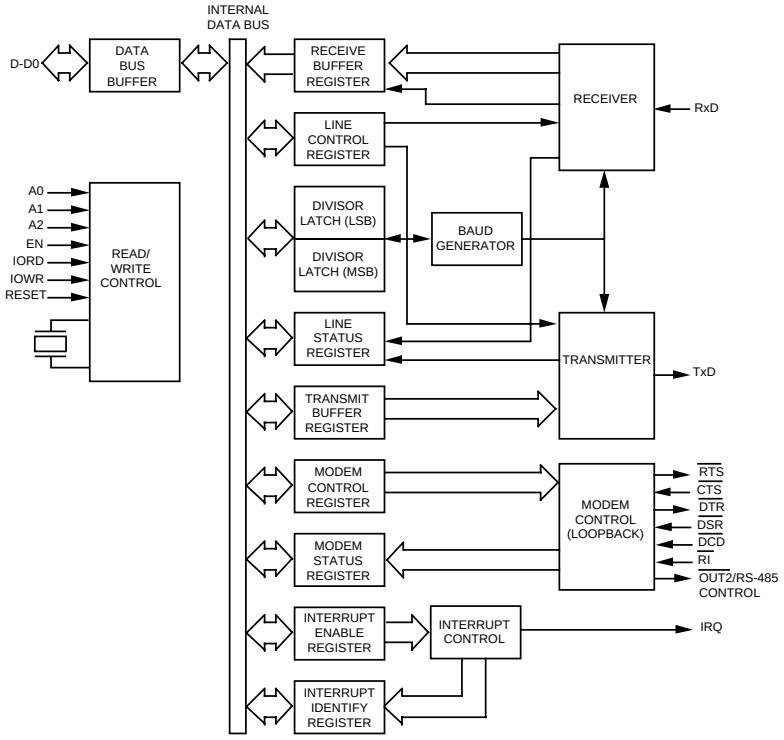


Figure 6–1. ACC Block Diagram.

FUNCTIONAL DESCRIPTION

The ACC is functionally compatible with the industry standard 16450/8250A found in most Personal Computers. The major functional blocks of the ACC are illustrated in Figure 6-1. A description of each functional block follows.

Read/Write Control

The main function of the Read/Write Control block is to supervise the interface between the ACC internal registers and the CPU. This includes enabling one of the ACC programmable registers onto the system data bus based on control signal inputs.

Receiver

The Receiver block converts serial data input on the RxD signal to a parallel format with the start, stop, and parity bits removed. The parallel data is placed in the Receive Buffer and the receiver Data Ready bit in the Line Status register is set to a logical 1. Activating the Data Ready bit generates an interrupt if the receive data interrupt is not masked by the Interrupt Enable register. The Receiver also monitors the incoming data for parity, overrun, and framing errors, and reports any such errors to the Line Status register.

Transmitter

The Transmitter block converts the contents of the Transmit Buffer from a parallel format into a serial string. Start, stop, and parity bits are added as specified by the programmer in the Line Control register. The serial string is then transmitted out of the TxD signal and the Transmitter Holding Register Empty bit is set to indicate that the ACC is ready to accept a new character for transmission. Activating the Transmitter Holding Register Empty bit also generates an interrupt if the transmit data interrupt is not masked by the Interrupt Enable register.

Modem Control

The Modem Control block includes the serial communication handshake, RS-422/485 buffer control, and loopback diagnostic control. The serial handshake signals output by this control block are Request to Send (RTS) and Data Terminal Ready (DTR). Both the RTS and DTR signals are programmable through the Modem Control register. The serial handshake signals monitored by this control block are Clear to Send (CTS), Data Set Ready (DSR), Data Carrier Detect (DCD), and Ring Indicator (RI). These signals are monitored through the Modem Status register. The DTR signal can be jumpered to serve as an enable for the Transmit Data and Request To Send drivers when the ACC is configured for RS-422/485 operation.

Loopback diagnostic control is achievable with the LOOP bit. When set, transmitted data is internally received.

Interrupt Control

The ACC includes a fully prioritized interrupt system to monitor and report operating conditions such as receiver data available and transmitter empty, and error conditions such as overrun, parity, framing, and break. The Interrupt Control block monitors the interrupt sources defined by the Interrupt Enable register and reports any active interrupts through the Interrupt Identify register.

PROGRAMMABLE REGISTERS

The remainder of the functional blocks illustrated in Figure 6-1 are programmable registers. The address of each of the registers is shown in Table 6-1. Please review I/O addressing on page 2-5.

Table 6-1
ACC Register Addressing.

UART +		Register	Operation	Page Number
0000	(DL = 0)	Receive Buffer	Read	6-8
		Transmit Buffer	Write	6-8
	(DL = 1)	Divisor Latch LSB	Read/Write	6-18
0001	(DL = 0)	Interrupt Enable	Read/Write	6-20
	(DL = 1)	Divisor Latch MSB	Read/Write	6-18
0002	(DL = X)	Interrupt Identify	Read	6-19
0003	(DL = X)	Line Control	Read/Write	6-8
0004	(DL = X)	Modem Control	Read/Write	6-13
0005	(DL = X)	Line Status	Read	6-11
0006	(DL = X)	Modem Status	Read/Write	6-16
0007	(DL = X)	Reserved		

Transmit and Receive Buffer

The Transmit Buffer, Receive Buffer, and the Interrupt Enable register share the same addresses as the Divisor Latch. Since the Divisor Latch is programmed only once during system initialization, this should cause no problems. During system initialization, the Divisor Latch is selected by programming the Divisor Latch Access Bit of the Line Control register. The Divisor Latch is then initialized to select the appropriate baud rate. The Divisor Latch Access Bit is reprogrammed to enable the Transmit Buffer, Receive Buffer, and Interrupt Enable register for serial communications.

Line Control

The main function of the Line Control register is to define the format of the serial data. The format for the Line Control register is shown in Figure 6-2. The Word Length Select (WLS) field defines the character length, as shown in Table 6-2.

Table 6-2
Serial Character Length.

Bit 1	Bit 0	Character Length
0	0	5 Bits
0	1	6 Bits
1	0	7 Bits
1	1	8 Bits

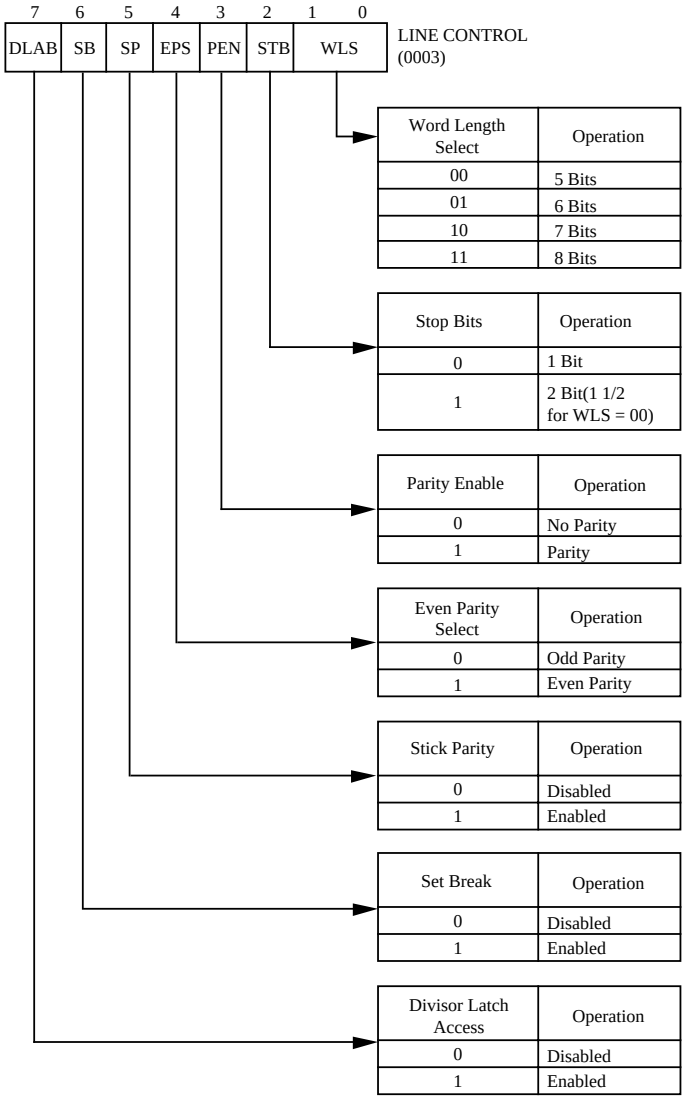


Figure 6–2. Line Control Register.

The number of stop bits added to each character transmitted and removed from each character received is selectable through the STB bit. Programming STB with a logical 0 selects one stop bit and programming STB with a logical 1 selects two stop bits. The exception to this is when the character length is defined as five. In this case, a logical 1 selects one and a half stop bits.

The parity options are programmed through the PEN, EPS, and SP bits. The parity is enabled and disabled through the PEN bit. A logical 0 in PEN disables parity and a logical 1 enables parity. With parity enabled, the ACC adjusts the parity bit of transmitted data to produce an even or an odd number of 1s when added to the character bits. With parity enabled, the ACC also tests the parity of the received data.

The EPS bit selects even or odd parity. Even parity means that there is an even number of 1s in the character data, including the parity bit. Odd parity means that there is an odd number of 1s in the character data, including the parity bit. Odd parity is selected by programming bit four with a logical 0, and even parity is selected by programming bit four with a logical 1. Bit five is the stick parity bit. With bits three and five a logical 1, the parity bit is transmitted and received in the state opposite that of bit four.

The STB bit position enables and disables break control. A logical 0 disables break and a logical 1 enables it. Enabling break forces the TxD output to the marking (negative) state.

The DLAB bit controls access to the ACC divisor latch. Programming bit seven with a logical 1 selects the Divisor Latch and programming bit seven with a logical 0 selects the Transmit Buffer, Receive Buffer, and Interrupt Enable register.

Line Status

The Line Status register provides information to the CPU concerning the data transfer. The Line Status bit map is shown in Figure 6-3. The DR bit indicates the state of the ACC Receive Buffer. A logical 1 in the DR bit signals the availability of a character in the Receive Buffer. The DR bit is automatically reset to a logical 0 when the Receive Buffer is read.

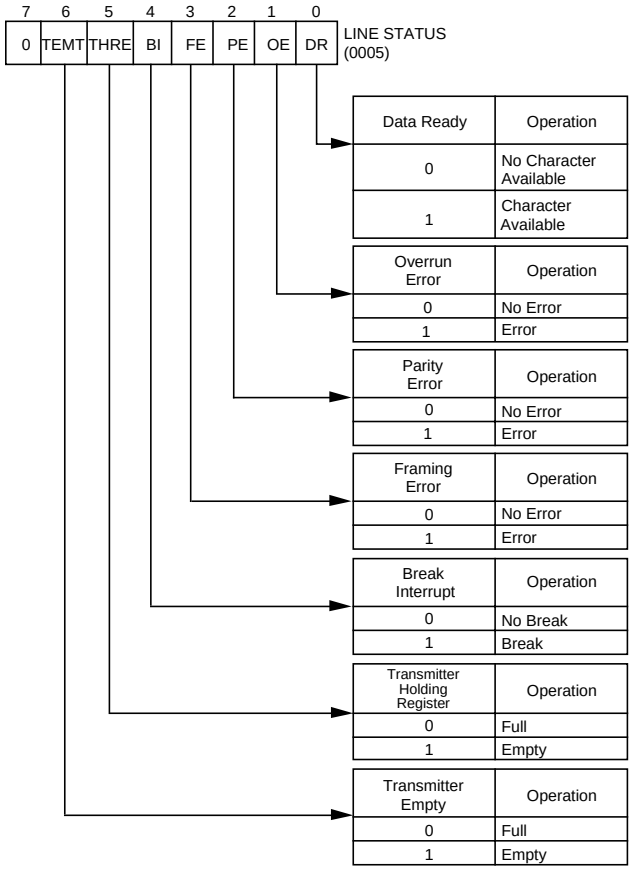


Figure 6–3. Line Status Register.

The OE, PE, and FE bits are transmission error indicators. An overrun error is indicated by a logical 1 in the OE bit, a parity error is indicated by a logical 1 in the PE bit, and a framing error is indicated by a logical 1 in the FE bit. In all cases the error indicator is reset when the Line Status register is read. An overrun error is generated when a character is written into the Receive Buffer before the previous character is read by the CPU. A parity error is generated when the received character does not have the correct even or odd parity. A framing error indicates that the received character does not have a valid stop bit. Any of these error bits are capable of generating an interrupt if enabled through the Interrupt Enable register.

The BI bit is set to a logical 1 when the RxD signal is held in the marking (negative) state for longer than a single character transmission (including start, data, parity, and stop bits). Bit four is reset to a logical 0 when the Line Status register is read. This bit is capable of generating an interrupt if enabled through the Interrupt Enable register.

The THRE and TEMT bits provide Transmitter status. A logical 1 in the THRE bit signals that the Transmit Buffer is empty and ready to receive a new character for transmission. This bit is capable of generating an interrupt if enabled through the Interrupt Enable register. TEMT is similar to bit five except bit six is not set to a logical 1 until both the Transmit Buffer and Transmit Shift register are empty. Both THRE and TEMT bits are automatically reset to a logical 0 with the loading of the next character to be transmitted.

Modem Control

The Modem Control register includes the serial handshake and RS-422/485 driver enable signals. The format of the Modem Control register is illustrated in Figure 6-4. The DTR bit defines the state of the Data Terminal Ready (DTR) signal. Programming the DTR bit with a logical 0 forces DTR to a marking (negative) state, and programming the DTR bit with a logical 1 forces DTR to a spacing (positive) state.

The RTS bit defines the state of the Request to Send (RTS) signal. Programming the RTS bit 0 with a logical 0 forces RTS to a marking (negative) state, and programming bit 0 with a logical 1 forces RTS to a spacing (positive) state.

The OUT2 signal can be jumpered to enable and disable the RS-422/485 transmit data and request to send drivers if the ACC serial port is configured for RS-422/485 operation (COM2 or COM3 only). In this configuration, programming the OUT2 bit with a logical 0 disables the drivers, and programming the OUT2 bit with a logical 1 enables the drivers. The power-on state of OUT2 disables the drivers.

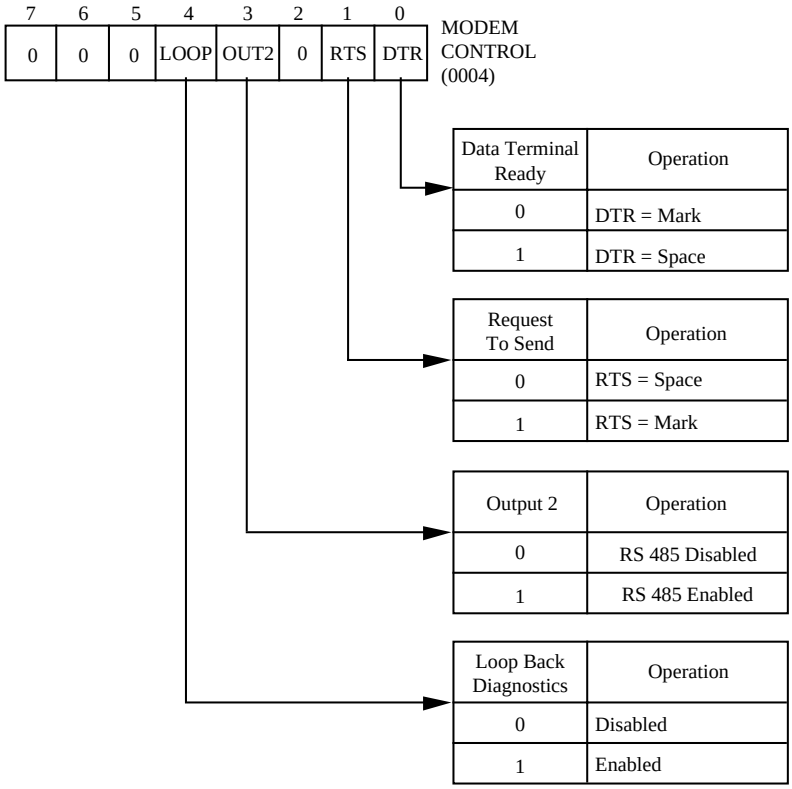


Figure 6–4. Modem Control Register.

The LOOP bit selects the internal loopback operation of the ACC. Programming LOOP with a logical 0 disables loopback, and programming LOOP with a logical 1 enables loopback. With loopback enabled, the following takes place: the RxD input is ignored; the TxD output is externally set to the marking (logical 1) state and internally connected to RxD; the Modem Control inputs (CTS, DSR, DCD, and RI) are ignored; the Modem Control outputs (DTR, RTS, and OUT2) are externally forced to a logical 1 and internally connected to the Modem Control inputs; and the receiver and transmitter interrupts are fully functional. During loopback, the data transmitted is immediately received.

Modem Status

The format of the Modem Status register is shown in Figure 6-5. All eight bits of this register are used to indicate the status of the serial data transfer. The DCTS, DDSR, and DDCD bits are set to a logical 1 to indicate that the CTS, DSR, and DCD bits have changed state since the last time the Modem Status register was read.

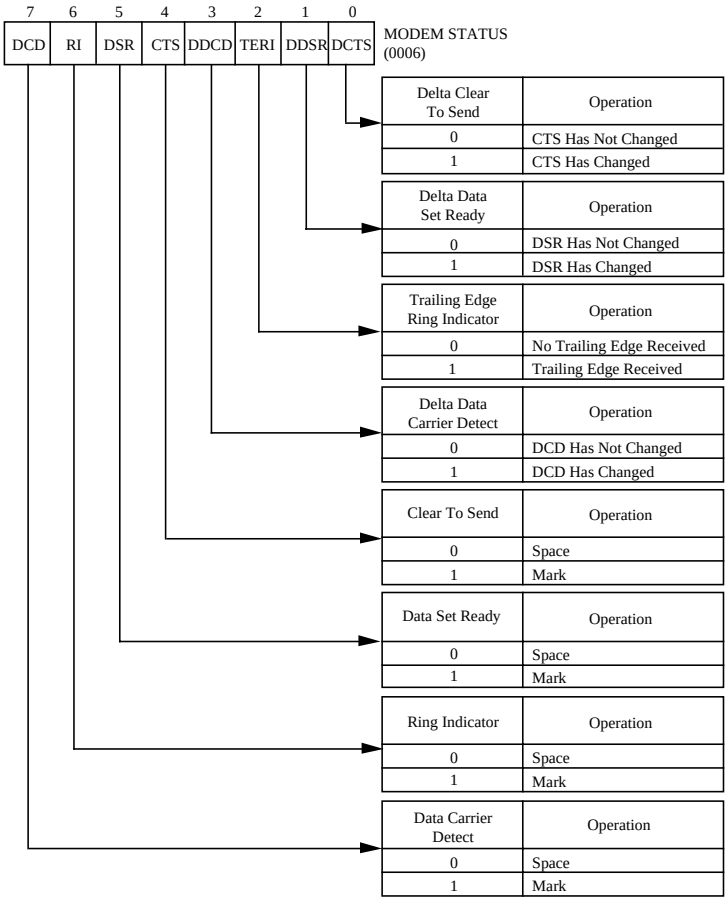


Figure 6–5. Modem Status Register.

The TERI bit signals the trailing edge detection of the Ring Indicator (RI). The TERI bit is a logical 1 if the RI input changed from a spacing (positive) state to a marking (negative) state since the last time the Modem Status register was read.

The CTS bit reflects the state of the Clear To Send (CTS) signal. If CTS is in the marking (negative) state, the CTS bit is a logical 0. Likewise, if CTS is in the spacing (positive) state, the CTS bit is a logical 1. During loopback mode, the CTS bit is internally connected to RTS.

The DSR bit reflects the state of the Data Set Ready (DSR) signal. If DSR is in the marking (negative) state, the DSR bit is a logical 0. Likewise, if DSR is in the spacing (positive) state, the DSR bit is a logical 1. During loopback mode, the DSR bit is internally connected to DTR.

The RI bit reflects the state of the Ring Indicator (RI) signal. If RI is in the marking (negative) state, the RI bit is a logical 0. Likewise, if RI is in the spacing (positive) state, the RI bit is a logical 1.

The DCD bit reflects the state of the Data Carrier Detect (DCD) signal. If DCD is in the marking (negative) state, the DCD bit is a logical 0. Likewise, if DCD is in the spacing (positive) state, the DCD bit is a logical 1.

Divisor Latch

The ACC baud rate is selected by programming the least significant byte (LSB) and most significant byte (MSB) of the Divisor Latch, shown in Figure 6-6. The Divisor Latch is programmed with an integer value that divides the ACC oscillator reference frequency of 18.432 MHz into a value of 16 times the baud rate for both transmission and reception of serial data. The most common baud rate divisors are shown in Table 6-4 on page 6-24.

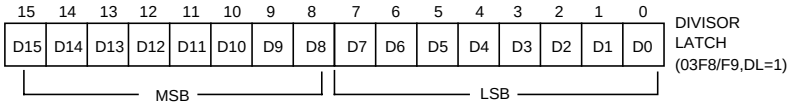


Figure 6–6. Divisor Latch.

Interrupt Identify

The ACC includes 10 sources of interrupts prioritized into four categories. The Interrupt Identify register flags whether or not the ACC has an interrupt pending and, if so, which of the four categories it originated from. The format for the Interrupt Identify register is shown in Figure 6-7.

The IP bit indicates whether or not there is an interrupt pending. A logical 0 in IP signals an active interrupt, and a logical 1 in IP signals no active interrupt. The interrupt sources are prioritized into the following four categories: receiver status, receiver data, transmitter data, and modem status, in that order. The ID field identifies the interrupt category if there is an active interrupt.

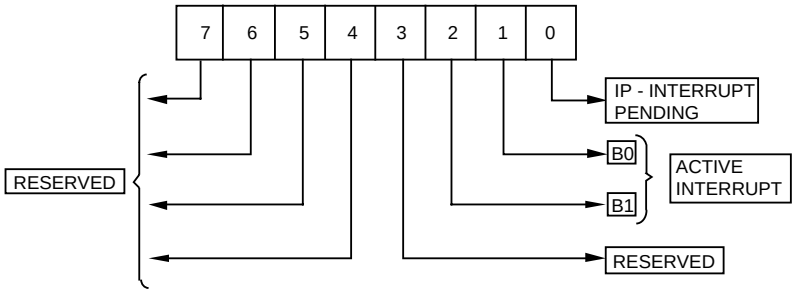


Figure 6–7. Interrupt Identify Register.

Interrupt Enable

The Interrupt Enable register defines which of the four categories of interrupts are enabled and which are not. The bit map for the Interrupt Enable register is illustrated in Figure 6-8. The Interrupt Enable register includes one bit for each of the four categories of interrupts. Setting bits 0 through 3 all to a logical 0 prevents the ACC from generating an interrupt of any kind and inhibits the Interrupt Identify register.

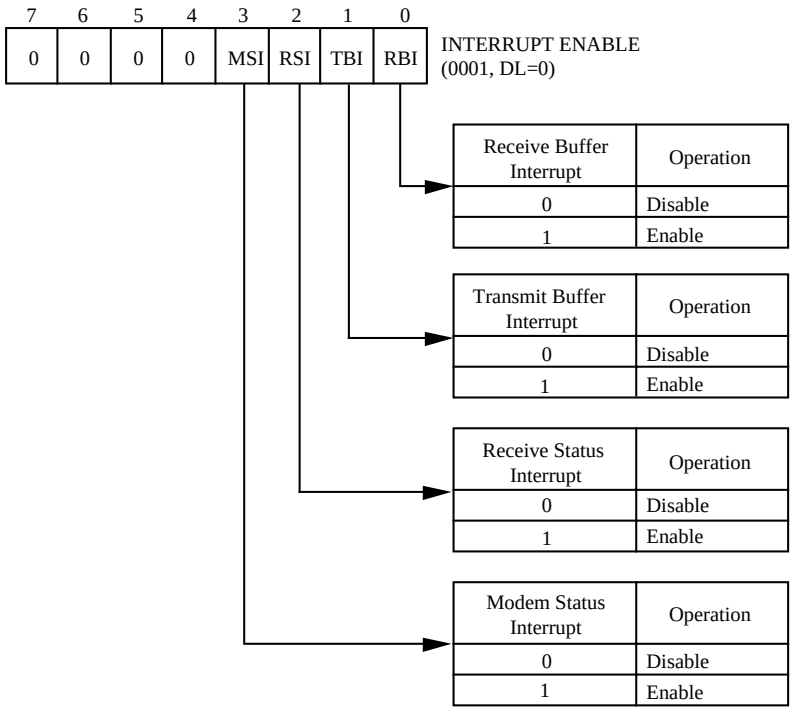


Figure 6–8. Interrupt Enable Register.

OPERATION

Reset

The ACC registers are automatically initialized to a default state after reset. The default state for these registers is shown in Table 6-3.

Table 6-3
ACC Register Defaults.

	Default Bit Value†							
Register	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Transmit Buffer	-	-	-	-	-	-	-	-
Receive Buffer	-	-	-	-	-	-	-	-
Line Control	0	0	0	0	0	0	0	0
Line Status	0	1	1	0	0	0	0	0
Modem Control	0	0	0	0	0	0	0	0
Modem Status	-	-	-	-	0	0	0	0
Divisor Latch (LSB)	-	-	-	-	-	-	-	-
Divisor Latch (MSB)	-	-	-	-	-	-	-	-
Interrupt Identify	0	0	0	0	0	0	0	1
Interrupt Enable	0	0	0	0	0	0	0	0

† Bit positions marked with a dash (-) can default to 1 or 0.

Serial Data Format

The ACC supports asynchronous data transfers. The format for the asynchronous data includes start, stop, and optional parity bits as well as five, six, seven, or eight data bits. This is illustrated in Figure 6-9.

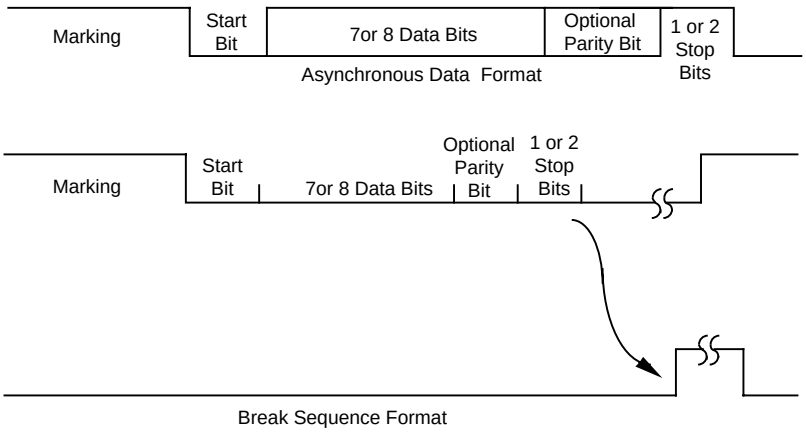


Figure 6–9. ACC Serial Data Format.

The TxD signal is in the "mark" state when data is not being transmitted. The start bit indicates the beginning of the serial data or character bits. The parity bit is optionally inserted during transmission and monitored during reception to test data integrity. The stop bit flags the end of the serial data.

The format for the break sequence is also shown in Figure 6-9. The transmission of the break sequence is programmed in the Line Control register, and the reception of the break sequence is flagged in the Line Status register.

Baud Rate

The ACC includes an internal baud rate generator to divide the 18.432 MHz reference frequency down to a value equal to 16 times the baud rate. The divisor for the reference frequency is a 16-bit integer programmed into the LSB and MSB of the Divisor Latch. The relationship between the baud rate and the count is as follows:

$$\text{Divisor} = 18.432 \times 10^6/(160 \times \text{Baud Rate})$$

Table 6-4 on page 6-24 lists the divisor to be used for the more popular baud rates. It also includes the percent error based on the difference between the exact divisor for a specified baud rate and the divisor obtainable with a 16-bit integer format. To guarantee proper operation, the percent error should never be greater than four.

Table 6-4
ACC Baud Rate Divisors.

Baud Rate	Divisor dec/hex	Percent Error
50	2304/1440	0
75	1536/0960	0
150	768/0480	0
300	384/0240	0
600	192/0120	0
1200	96/0060	0
1800	64/0040	0
2000	58/003A	0.69
2400	48/0030	0
3600	32/0020	0
4800	24/0018	0
7200	16/0010	0
9600	12/000C	0
19200	6/0006	0
38400	3/0003	0
56000	2/0002	2.86

Interrupt and Polled Communication

Serial data can be transferred and received by polling status bits or with interrupts. In a polled mode, the application program monitors the Line Status register Data Ready and Transmitter Holding Register Empty bits to determine when to transfer a character to the ACC or when a character is available.

The ACC is capable of generating interrupts for applications that cannot afford the time wasted in polling for a status change. If enabled in the Interrupt Enable register, a Transmitter Holding Register Empty interrupt is generated when the transmitter is free to receive a character. Also, if enabled in the Interrupt Enable register, a Received Data Available interrupt is generated when a character is received by the ACC.

RS-422/485 Serial Communications

The ZT 8841 includes jumper selectable drivers and receivers for both RS-232-C and RS-422/485 serial communication protocols. The two main advantages of RS-422/485 over RS-232-C are an increased number of drivers and receivers (32 of each, versus one of each for RS-232-C) and increased transmission distance (4000 feet, as compared to 50 feet for RS-232-C).

A terminated twisted pair should be used to protect the integrity of the RS-422/485 signals. A typical twisted pair has a characteristic impedance in the range of 100 to 130 Ω, adequate for data transfer rates up to 10 MHz. The RS-422/485 balanced inputs can be terminated by installing resistors at locations R1 and R2 for COM3 (J3) and at R3 and R4 for COM2 (J2). The resistors must have a power rating of at least 0.125 W and a resistance value equal to the characteristic impedance of the twisted pair.

Appendix B includes the pin numbers and signal descriptions of the RS-232-C and RS-422/485 signals available through connectors J2 and J3. When configured for RS-422/485, Data Terminal Equipment (DTE) is selected by plugging in the serial cable with pin 1 of the cable lined up with pin 1 of the connector. To select Data Communication Equipment (DCE), the serial cable should be plugged in with pin 1 of the cable lined up with pin 14 of the connector.

Programming

The ACC register set is summarized in Table 6-5 for programming reference.

Table 6-5
ACC Register Summary.

Register Address					
Bit No.	0 DLAB = 0	0 DLAB = 0	1 DLAB = 0	2	3
	Receive Buffer (Read Only)	Transmit Buffer (Write Only)	Interrupt Enable Register	Interrupt Identity Register (Read Only)	Line Control Register
	RBR	THR	IER	IIR	LCR
0	Data Bit 0*	Data Bit 0	Enable Receive Buffer Full Interrupt (MSI)	"0" if Interrupt Pending	Word Length Select Bit 0 (WLS0)
1	Data Bit 1	Data Bit 1	Enable Transmit Buffer Empty Interrupt (RSI)	Interrupt ID Bit (0)	Word Length Select Bit 1 (WLS1)
2	Data Bit 2	Data Bit 2	Enable Receive Status Interrupt (TBI)	Interrupt ID Bit (1)	Number of Stop Bits (STB)
3	Data Bit 3	Data Bit 3	Enable Modem Status Interrupt (RBI)	0	Parity Enable (PEN)
4	Data Bit 4	Data Bit 4	0	0	Even Parity Select (EPS)
5	Data Bit 5	Data Bit 5	0	0	Stick Parity
6	Data Bit 6	Data Bit 6	0	0	Set Break
7	Data Bit 7	Data Bit 7	0	0	Divisor Latch Access Bit (DLAB)

*Bit 0 is the least significant bit. It is the first bit serially transmitted or received.

Table 6-5
ACC Register Summary (continued).

Register Address						
Bit No.	4	5	6	7	0 DLAB = 1	1 DLAB = 1
	Modem Control Register	Line Status Register	Modem Status Register	Scratchpad Register	Divisor Latch (LSB)	Divisor Latch (MSB)
	MCR	LSR	MSR	SCR	DLL	DLM
0	Data Terminal Ready (DTR)	Data Ready (DR)	Delta Clear to Send (DCTS)	SCR0	D0	D8
1	Request to Send (RTS)	Overrun Error (OE)	Delta Data Set Ready (DDSR)	SCR1	D1	D9
2	OUT1 (Not available externally)	Parity Error (PE)	Trailing Edge Ring Indicator (TERI)	SCR2	D2	D10
3	OUT2 (Interrupt output enable)	Framing Error (FE)	Delta Data Carrier Detect (DDCD)	SCR3	D3	D11
4	Loop	Break Interrupt (BI)	Clear to Send (CTS)	SCR4	D4	D12
5	0	Transmit Holding Register (THRE)	Data Set Ready (DSR)	SCR5	D5	D13
6	0	Transmit Empty (TEMT)	Ring Indicator (RI)	SCR6	D6	D14
7	0	0	Data Carrier Detect (DCD)	SCR7	D7	D15

IV. APPENDICES

JUMPER CONFIGURATIONS A-1
SPECIFICATIONS B-1
QUICK REFERENCE C-1
CUSTOMER SUPPORT D-1
GLOSSARY E-1

Appendix A

JUMPER CONFIGURATIONS

Contents **Page**

OVERVIEW A-1
JUMPER DESCRIPTIONS A-5

OVERVIEW

The ZT 8841 includes several jumper-selectable options that tailor the operation of the board to the requirements of specific applications.

Table A-1, "Jumper Cross Reference," lists the jumpers associated with each option. It also indicates the pages on which descriptions of these jumpers can be found.

The ZT 8841 jumper options are summarized in Table A-2, which begins on page A-5. A dagger (†) indicates a factory default jumper configuration. The complete factory default jumper configuration is illustrated in Figure A-1 on page A-3.

Figure A-2, "ZT 8841 Customer Jumper Locations," on page A-4 shows jumper pin locations. You may wish to document your customized jumper configuration using this figure.

Table A-1
Jumper Cross Reference.

Function	Jumper #	Pages
COM and PIC I/O addressing	W9, W10	A-6
COM and PIC interrupts	W11, W16-W20, W23, W27, W29-W36, W65-W69	A-7, A-9 A-13, A-15 A-19
Inverter (wire-wrap access)	W90	A-23
I/O port addressing	W1-W8	A-5, A-6
IRQx enable	W86, W87	A-21
RI isolation from CSA	W88, W89	A-22
RS-232-C/RS-485 operation	W21, W22, W24-26, W28, W37-W64, W70-W85	A-10, A-14 A-14 A-10, A-16, A-17 A-16, A-17 A-20
+5 V supply to frontplane	W50, W64, W76, W83	A-17
+5 V (wire-wrap access)	W91	A-23

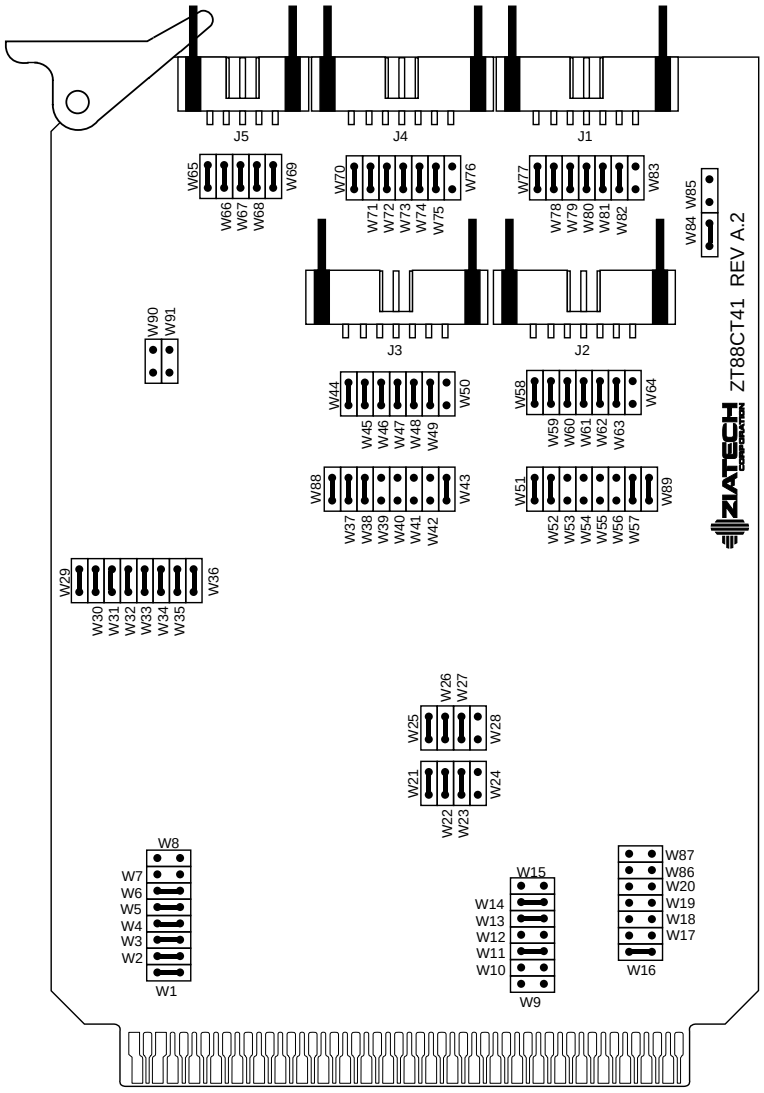


Figure A–1. ZT 8841 Default Jumper Configuration.

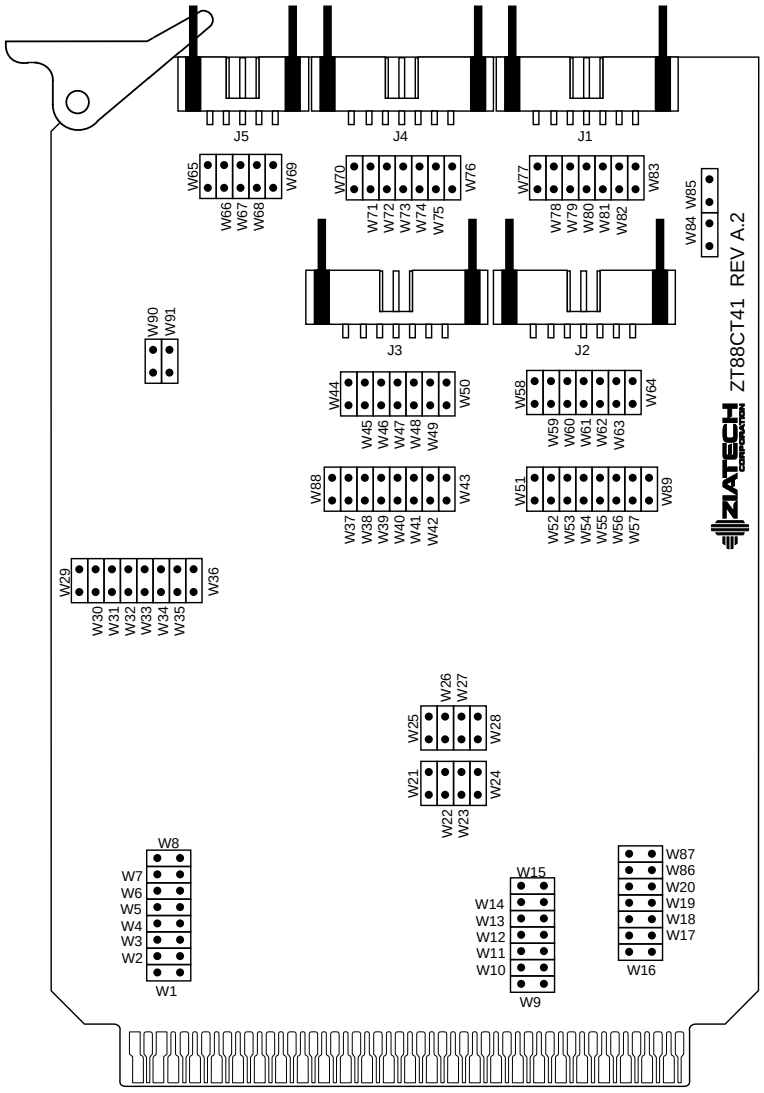


Figure A–2. ZT 8841 Customer Jumper Configuration.

JUMPER DESCRIPTIONS

Table A-2
ZT 8841 Jumper Descriptions.

JUMPER #	DESCRIPTION
W1-W6	Select the value of I/O port address bits A10-15 that make up the upper six bits of the ZT 8841 Base Address. W1-6 correspond to address bits A15-10, respectively. When installed, the address line must be false, and when removed, the address line must be true to select the board. Factory default = W1-6 installed. See pages 2-6 to 2-8 for more information.

Jumper	Address Line
W1 out	A15=1
† W1 in	A15=0
W2 out	A14=1
† W2 in	A14=0
W3 out	A13=1
† W3 in	A13=0
W4 out	A12=1
† W4 in	A12=0
W5 out	A11=1
† W5 in	A11=0
W6 out	A10=1
† W6 in	A10=0

† Factory default jumper configuration.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER # **DESCRIPTION**

W7, W8

W8 enables the ZT 8841 to decode IOEXP. If W8 is installed, W7 will select IOEXP polarity. W7 installed requires IOEXP to be low. W7 removed requires IOEXP to be high. Factory default removes both jumpers to ignore IOEXP.

Jumper	Operation
W8 in, W7 in	IOEXP decoded low true
W8 in, W7 out	IOEXP decoded high true
† W8 out, W7 out	IOEXP ignored

W9, W10

Selects the COM and PIC I/O addresses.†† The selections are:

W10	W9	COM1	COM2	COM3	COM4	PIC
in	in	3F8-3FFh	2F8-2FFh	3E8-3EFh	2E8-2EFh	020-027h
out	in	3E0-3E8h	2F8-2FFh	3E8-3EFh	2E8-2EFh	020-027h
in	out	3E0-3E8h	2F8-2FFh	3E8-3EFh	2E8-2EFh	0A8-0AFh
† out	out	3E0-3E8h	2E0-2E8h	3E8-3EFh	2E8-2EFh	0A8-0AFh

† Factory default jumper configuration.
†† W65-W69 contain spare jumpers that can be used as needed.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W11	Enables the on-board PIC to generate an interrupt vector. When removed, only polled operation of the PIC is possible. Default is W11 installed. See page 3-11 for additional discussion.

Jumper	Operation
† W11 in	PIC interrupt vector generation
W11 out	Polled PIC interrupts

JUMPER #	DESCRIPTION
W12, W13	Are used by Ziatech for testing. W13 should always be installed. W12 should always be removed.

Jumper	Operation
† W12 out	Ziatech test jumper - do not install
† W13 in	Ziatech test jumper - do not remove

† Factory default jumper configuration.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W14, W15	Selects the number of wait states that will be inserted. W14 selects two wait states for 8 MHz operation (factory default). W15 selects one wait state for 5 MHz operation.

Jumper	Operation
† W14 in, W15 out	Selects 2 wait states (8 MHz operation)
W14 out, W15 in	Selects 1 wait state (5 MHz operation)
W14 out, W15 out	Selects 0 wait states

† Factory default jumper configuration.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W16-W20	Enable interrupts from the UARTs (COM1-4) and PIC onto STD bus signals INTRQ*, INTRQ1*, or INTRQ2* (CNTL). W16 selects the PIC interrupt for INTRQ*. W17 and W18 select COM1 and COM3 interrupts, respectively, for INTRQ1*. W19 and W20 select COM2 and COM4 interrupts, respectively, for INTRQ2*. If interrupts are to be routed over the frontplane, see jumpers W65-69. Default configuration is W16 installed. See page 3-18 for additional discussion.

Jumper	Operation
† W16 in	PIC interrupt on INTRQ*
W16 out	No PIC interrupt on INTRQ*
W17 in	COM1 interrupts on INTRQ1*
† W17 out	No COM1 interrupts on INTRQ1*
W18 in	COM3 interrupts on INTRQ1*
† W18 out	No COM3 interrupts on INTRQ1*
W19 in	COM2 interrupts on INTRQ2*
† W19 out	No COM2 interrupts on INTRQ2*
W20 in	COM4 interrupts on INTRQ2*
† W20 out	No COM4 interrupts on INTRQ2*

† Factory default jumper configuration (other combinations are possible using wire wrap).

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W21, W22, W25, W26, W37-W43, W51-W57	Select RS-232-C or RS-422/485 operation for COM2 and COM3. COM3 RS-232-C operation (default): remove W39-W42 and install W21, W22, W37, W38, and W43. COM2 RS-232-C operation (default): remove W53-W56 and install W25, W26, W51, W52, and W57. See Figure A-3. COM3 RS-422/485 operation: remove W21, W22, W37, W38, and W43 and install W39-W42. COM2 RS-422/485 operation: remove W25, W26, W51, W52, and W57 and install W53-W56. See Figure A-4. Refer also to the description of jumpers W84 and W85 on page A-20 and jumpers W24 and W28 on page A-14.

Jumper	COM3	COM2	Operation
† in	W21,22,37,38,43	W25,26,51,52,57	RS-232-C
† out	W39-42	W53-56	RS-232-C
in	W39-42	W53-56	RS-422/485
out	W21, 22,37,38,43	W25,26,51,52,57	RS-422/485

† Factory default jumper configuration.

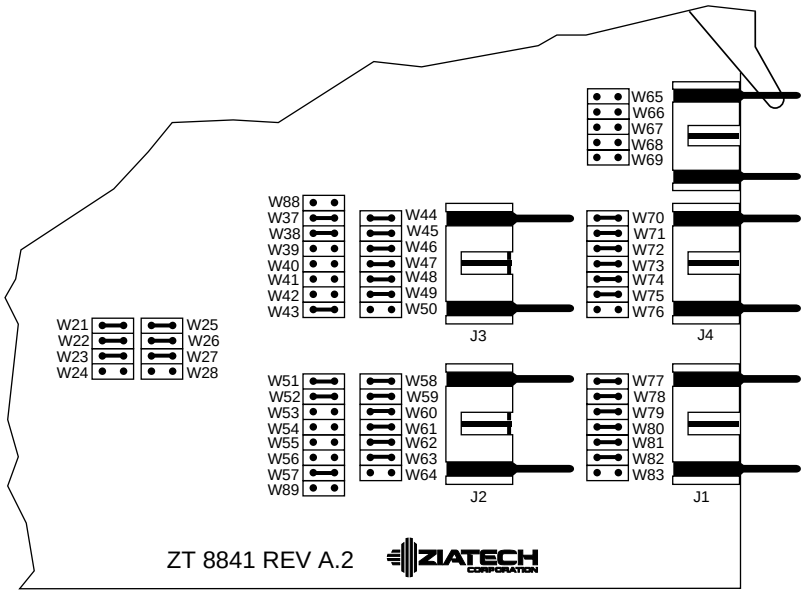


Figure A–3. COM1-4 RS-232-C DTE (Default).

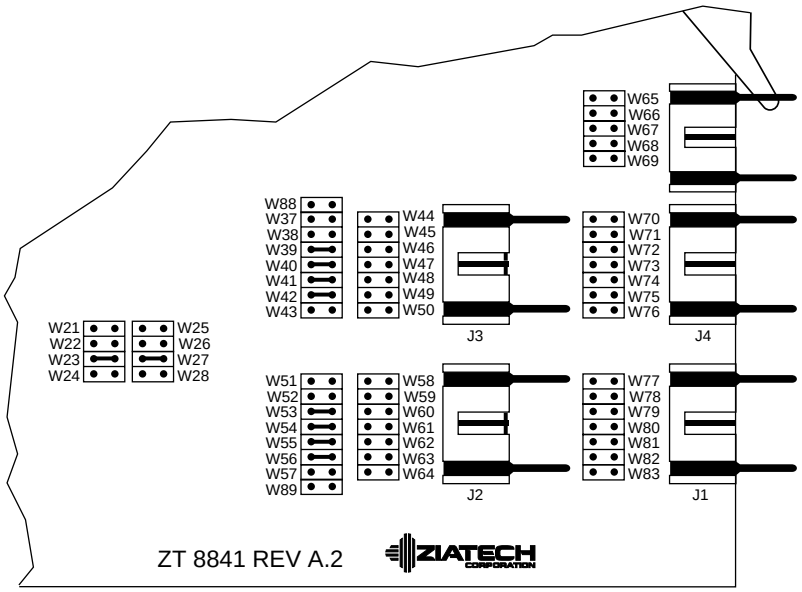


Figure A–4. COM2-3 RS-422/485 Operation.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W23, W27	Enable the interrupt from COM3 or COM2, respectively, to be gated by the OUT2 pin on the UART. When W23 and W27 are removed, the COM3 and COM2 interrupts will always be routed to the PIC and to the back/front plane interrupt routing jumper blocks (W16-20, W65-69).

Jumper	Operation
† W23 in	COM3 INT gated w/ UART OUT2
W23 out	COM3 INT not gated
† W27 in	COM2 INT gated w/ UART OUT2
W27 out	COM2 INT not gated

† Factory default jumper configuration.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W24, W28	Allow the OUT2 pin on the COM3 and COM2 UARTs to control the RS-422/485 transmitters for RS-485 multidrop operation. W24 and W28 should be removed for RS-422 non-multidrop operation and for RS-232-C operation of COM3 and COM2.

Jumper	Operation
W24 in	COM3 RS-485 multidrop
† W24 out	COM3 RS-422 non-multidrop
W28 in	COM2 RS-485 multidrop
† W28 out	COM2 RS-422 non-multidrop

† Factory default jumper configuration.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W29-W36	Select the PIC Interrupt Request (IR) line used for each of the UARTs. Default installs W33 for COM1 on IR4, W32 for COM2 on IR3, W35 for COM3 on IR6, and W34 for COM4 on IR5. Extra jumper positions are for wire-wrapping a custom configuration. See page 3-21 for additional information.

Jumper	Operation
† W29 in	IR0 selected for user expansion
W29 out	IR0 not selected
† W30 in	IR1 selected for user expansion
W30 out	IR1 not selected
† W31 in	IR2 selected for user expansion
W31 out	IR2 not selected
† W32 in	IR3 selected for COM2
W32 out	IR3 not selected
† W33 in	IR4 selected for COM1
W33 out	IR4 not selected
† W34 in	IR5 selected for COM4
W34 out	IR5 not selected
† W35 in	IR6 selected for COM3
W35 out	IR6 not selected
† W36 in	IR7 selected for user expansion
W36 out	IR7 not selected

† Factory default jumper configuration.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W44-W49, **W58-W63,** **W70-W75,** **W77-W82**	Configure the COM ports for DTE or DCE RS-232-C operation. COM3: install jumpers W44-W49 (default). Remove W44-W49 for RS-422/485 operation. COM2: install jumpers W58-W63 (default). Remove W58-W63 for RS-422/485 operation. COM1: install jumpers W77-82. COM4: install jumpers W70-75. These jumpers can be rotated 90˚ for DCE configuration on any of the serial channels. See Figure A-5 on page A-18 for COM1-COM4 configured for RS-232-C DCE operation. COM1-COM4 correspond to ZT 8841 connectors J1-J4.

Jumper	Operation
† W44-W49 in	COM3 RS-232-C DTE
W44-W49 out	COM3 RS-422/485 DTE
† W58-W63 in	COM2 RS-232-C DTE
W58-W63 out	COM2 RS-422/485 DTE
† W77-W82 in	COM1 RS-232-C DTE
W77-W82 out	COM1 disabled
† W70-W75 in	COM4 RS-232-C DTE
W70-W75 out	COM4 disabled

† Factory default jumper configuration; rotate 90˚ for DCE operation.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W50,W64, **W76,W83**	Supply +5 V to J3, J2, J4, and J1, respectively. Default configuration removes these jumpers. **Warning:** Do not connect W49 to W50, W75 to W76, W63 to W64, or W82 to W83. The +5 V power will damage the RS-232-C drivers if any of these connections are made.

Jumper	Operation
W50 in	Supplies +5 V to J3 pin 6
† W50 out	No power supplied to J3
W64 in	Supplies +5 V to J2 pin 6
† W64 out	No power supplied to J2
W76 in	Supplies +5 V to J4 pin 6
† W76 out	No power supplied to J4
W83 in	Supplies +5 V to J1 pin 6
† W83 out	No power supplied to J1

† Factory default jumper configuration.

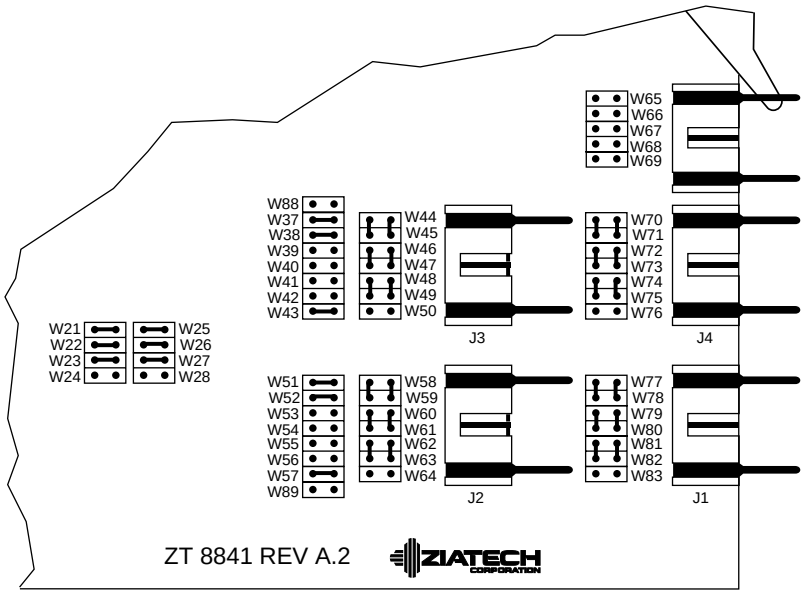


Figure A–5. COM1-COM4 RS-232-C DCE Operation.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W65-W69	Connect interrupts from the UARTs (COM1-4) and PIC to frontplane connector J5. These interrupts can be wire-wrapped to other places on the board if desired.

Jumper	Source	Name	J5 Pin #
† W65 in	8259A	$\overline{FP1}$	2
† W66 in	COM2	$\overline{FP3}$	4
† W67 in	COM4	$\overline{FP5}$	6
† W68 in	COM1	$\overline{FP6}$	8
† W69 in	COM3	$\overline{FP7}$	10

† Factory default jumper configuration.

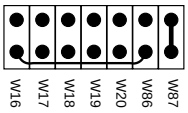
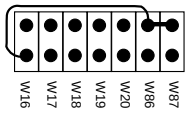
Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W84, W85	Are installed on non-optically isolated versions of the ZT 88CT41 to connect ground and +5 V to the RS-422/485 drivers and receivers. Default installs W84 (RS-232-C operation enabled) and removes W85 (RS-422/485 operation disabled). Remove W85 if the RS-422/485 drivers are not being used and if low power or extended temperature operation is desired. See page 3-24 for more information.

Jumper	Operation
† W84 in	RS-232-C operation enabled
W84 out	RS-232-C operation disabled
W85 in	RS-422/485 operation enabled
† W85 out	RS-422/485 operation disabled

† Factory default jumper configuration.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W86, W87	Enable IRQx. The STD 32 IRQx signal can be configured for active high or active low operation. Active high IRQx operation: install W87 and wire-wrap the bottom pin of W86 to the bottom pin of W16. See Figure A-6.  *Figure A–6. Active High IRQx.* Active low IRQx operation: Connect the top pins of W86 and W87 together with a jumper connector. Wire-wrap the top pin of W86 to the bottom pin of W16. See Figure A-7.  *Figure A–7. Active Low IRQx.*

Note: IRQx is supported only in an STD 32 card cage with a ZT 89CT39 arbitration card or master controller present in slot X.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W88	Isolates the RI signal on connector J3. Remove W88 to isolate the RI signal on J3 from CSA when operating in RS-485 mode.

Jumper	Operation
† W88 in	RI not isolated from CSA on J3
W88 out	Isolates RI on J3 from CSA

JUMPER #	DESCRIPTION
W89	Isolates the RI signal on connector J2. Remove W89 to isolate the RI signal on J2 from CSA when operating in RS-485 mode.

Jumper	Operation
† W89 in	RI not isolated from CSA on J2
W89 out	Isolates RI on J2 from CSA

† Factory default jumper configuration.

Table A-2
ZT 8841 Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W90	Provides wire-wrap access to an inverter. With board oriented component side up and edge connectors to the left, W90's left pin is connected to the input of a 74ACT04 inverter. W90's right pin is connected to the output. (Factory default: no jumper installed in W90.)

W90 Pin	Operation
Left	74ACT04 inverter input
Right	74ACT04 inverter output

W91	Provides wire-wrap access to power and ground. With board oriented component side up and edge connectors to the left, W91's left pin is tied to +5 V. W91's right pin is tied to ground. (Factory default: no jumper installed in W91.)

W91 Pin	Operation
Left	+5 V power source
Right	Ground

Appendix B

SPECIFICATIONS

Contents **Page**

INTRODUCTION B-1
ELECTRICAL SPECIFICATIONS B-2
Absolute Maximum Ratings B-2
DC Operating Characteristics B-2
ZT 8841 Power Requirements B-2
ZT 88CT41 Power Requirements B-2
STD Load Characteristics B-4
ENVIRONMENTAL SPECIFICATIONS B-7
Operating Temperature B-7
Storage Temperature B-7
MECHANICAL SPECIFICATIONS B-8
Connectors B-8
Cable Specifications B-12
Card Dimensions & Weight B-15

INTRODUCTION

This appendix provides electrical, environmental, and mechanical specifications. Where appropriate, specifications for both the ZT 8841 and ZT 88CT41 boards are given.

ELECTRICAL SPECIFICATIONS

The ZT 8841 absolute maximum, operating, and STD load characteristics are outlined below:

Absolute Maximum Ratings

Supply Voltage, Vcc ..... 0 to 7 V
Supply Voltage, AUX +V ..... 0 to 15 V
Supply Voltage, AUX -V ..... 0 to -15 V

DC Operating Characteristics

ZT 8841 Power Requirements

Operating w/82050 UARTs: +5 VDC ±5% @ 0.270 A typ.

Operating w/82510 UARTs: +5 VDC ±5% @ 0.270 A typ.

ZT 88CT41 Power Requirements

Powered Standby With 82050 UARTs: +5 VDC ±5% @ 0.095 A typ (processor HALTed)

Powered Standby With 82510 UARTs: +5 VDC ±5% @ 0.065 A typ (processor HALTed and 82510 UARTs in power-down mode)

Power Req. (RS-232)	Min.	Typ.	Max.
Supply Voltage, Vcc	4.75 V	5.00 V	5.25 V
Supply Current, Vcc = 5.0 V	-	92 mA	200 mA
Aux Voltage, Vaux+	10.8 V	12.0 V	13.2 V
Aux Current, Vaux+ = 12.0 V	-	30 mA	60 mA
Aux Voltage, Vaux-	-13.2 V	-12.0 V	-10.8 V
Aux Current, Vaux- = -12.0 V	-	30 mA	60 mA

Power Req. (RS-485)	Min.	Typ.	Max.
Supply Voltage, Vcc †	4.75 V	5.00 V	5.25 V
Supply Current, Vcc = 5.0 V	-	162 mA	345 mA

† If RS-232 is not used, ±12 V are not required.

STD Load Characteristics

Both the ZT 8841 and the ZT 88CT41 are TTL compatible on the STD bus.

The unit load is a convenient method for specifying input and output drive capability of STD bus cards. In STD bus systems, one unit load is equal to one LSTTL load as follows:

- Max. high level input current: 20 μA.
- Max. low level input current: -400 μA.

The STD bus unit load reflects input current requirements at worst case conditions over recommended supply voltage and ambient temperature ranges. An output rated at 60 unit loads can drive 60 STD bus cards having input rated at one unit load. Table B-1 includes load values for the STD-80 connections. Table B-2 includes load values for STD 32 connections.

Note: The ZT 8841 and the ZT 88CT41 use CMOS devices that have high and low input currents of ±10 μA.

Table B-1
STD Bus Loading, P Connector.

PIN (CIRCUIT SIDE) / OUTPUT DRIVE / INPUT LOAD / MNEMONIC							PIN (COMPONENT SIDE) / OUTPUT DRIVE / INPUT LOAD / MNEMONIC
+5 VDC [A]			2	1			+5 VDC [A]
GND			4	3			GND
DCPDN*			6	5			VBAT [NC]
D7 (A23)	1	59	8	7	59	1	D3 (A19)
D6 (A22)	1	59	10	9	59	1	D2 (A18)
D5 (A21)	1	59	12	11	59	1	D1 (A17)
D4 (A20)	1	59	14	13	59	1	D0 (A16)
A15	1		16	15		2	A7
A14	1		18	17		1	A6
A13	1		20	19		1	A5
A12	1		22	21		1	A4
A11	1		24	23		1	A3
A10	1		26	25		1	A2
A9	2		28	27		1	A1
A8	2		30	29		1	A0
RD*	1		32	31		1	WR*
MEMRQ*			34	33		1	IORQ*
BHE			36	35		1	IOEXP
MCSYNC*			38	37	60		INTRQ1*
STATUS 0*	1		40	39		1	STATUS 1*
BUSRQ*			42	41			BUSAK*
INTRQ*		60	44	43		1	INTAK*
NMIRQ*			46	45	60		WAITRQ*
PBRESET* [NC]			48	47		1	SYSRESET*
INTRQ2* (CNTRL*)		60	50	49			CLOCK*
PCI [B]			52	51			PCO [B]
AUX GND			54	53			AUX GND
AUX-V			56	55			AUX+V

Notes to the STD Bus Loading Diagram
[A] is the logic supply drain = .27 Amps typ.
[B] PCI and PCO are shorted together.
[NC] No Connection.

Table B-2
STD Bus Loading, E Connector.

PIN (CIRCUIT SIDE) / OUTPUT DRIVE / INPUT LOAD / MNEMONIC							PIN (COMPONENT SIDE) / OUTPUT DRIVE / INPUT LOAD / MNEMONIC
RSVD			E2	E1			GND
XA23			E4	E3			XA19
XA22			E6	E5			XA18
XA21			E8	E7			XA17
XA20			E10	E9			XA16
RSVD			E12	E11			NOWS*
+5 VDC			E14	E13			+5 VDC
MREQX*			E16	E15			MAKx*
GND			E18	E17			GND
D31			E20	E19			D27
D30			E22	E21			D26
D29			E24	E23			D25
D28			E26	E25			D24
GND			E28	E27			D23
D15			E30	E29			D22
D14			E32	E31			D21
D13			E34	E33			D20
D12			E36	E35			GND
D11			E38	E37			D19
D10			E40	E39			D18
D9			E42	E41			D17
D8			E44	E43			D16
MASTER16*			E46	E45			GND
AENx*			E48	E47	60		IRQx
BE3*			E50	E49			BE1*
BE2*			E52	E51			BE0*
GND			E54	E53			MEM16*
W-R			E56	E55			M-IO
DMAIOR*			E58	E57			DMAIOW*
EX8*			E60	E59			IO16*
START*			E62	E61			CMD*
EX32*			E64	E63			EX16*
T-C			E66	E65			EXRDY
+5 VDC			E68	E67			LOCK*
DREQx*			E70	E69			DAKx*
MSBURST*			E72	E71			SLBURST*
XA31*			E74	E73			XA27*
XA30*			E76	E75			XA26*
XA29*			E78	E77			XA25*
XA28*			E80	E79			XA24*

ENVIRONMENTAL SPECIFICATIONS

Operating Temperature

ZT 8841: 0˚ through +65˚ Celsius @ < 95% relative humidity at 40˚ Celsius, non-condensing

ZT 88CT41 (RS-232-C): -40˚ through +85˚ Celsius @ < 95% relative humidity at 40˚ Celsius, non-condensing

ZT 88CT41 (RS-422/485): 0˚ through +65˚ Celsius @ < 95% relative humidity at 40˚ Celsius, non-condensing

Storage Temperature

-65˚ through +150˚ Celsius @ < 95% relative humidity at 40˚ Celsius, non-condensing

MECHANICAL SPECIFICATIONS

Connectors

The ZT 8841 has one card-edge connector (P) and five right angle frontplane connectors (J1-J5). The connectors are described below:

P: The P connector is a 56-pin (dual 28-pin) printed circuit board edge connector with fingers on 0.125 inch centers. The mating connector is a Viking 3VH28/1CNK5 or equivalent for the solder tail, or a Viking 3VH28/1CND5 or equivalent for a three-level wire-wrap. Pin assignments are shown on page B-5.

E: The E connector extends the P connector to interface the ZT 8841 to the STD 32 bus. This connector combines with the P connector to make a 114-pin (dual 57-pin) card-edge connector with fingers on 0.062 inch centers. The mating connector is a Viking S3VT68/5DP12 or equivalent for the solder tail, or a Viking S3VT68/5DE12 or equivalent for the card extender version. Pin assignments for the E connector are shown on page B-6.

J1-4: Connectors J1-4 are 14-pin (dual 7-pin) male latching headers with 0.1 inch spacing. The mating connector is a T&B Ansley #622-1430 or equivalent. J1-4 pin assignments are given in Figure B-1 on page B-9, along with pin numbers for a 25-pin D-type ribbon cable connector. Tables B-3 and B-4 show pin assignments for RS-232-C and RS-422/485, respectively. Serial channels COM1-COM4 are accessed through connectors J1-J4, respectively.

J5: Connector J5 is a 10-pin (dual 5-pin) male header with 0.1 inch spacing. The mating connector is a T&B Ansley #622-1030 or equivalent. J5 pin assignments for frontplane interrupts are shown in Table B-5.

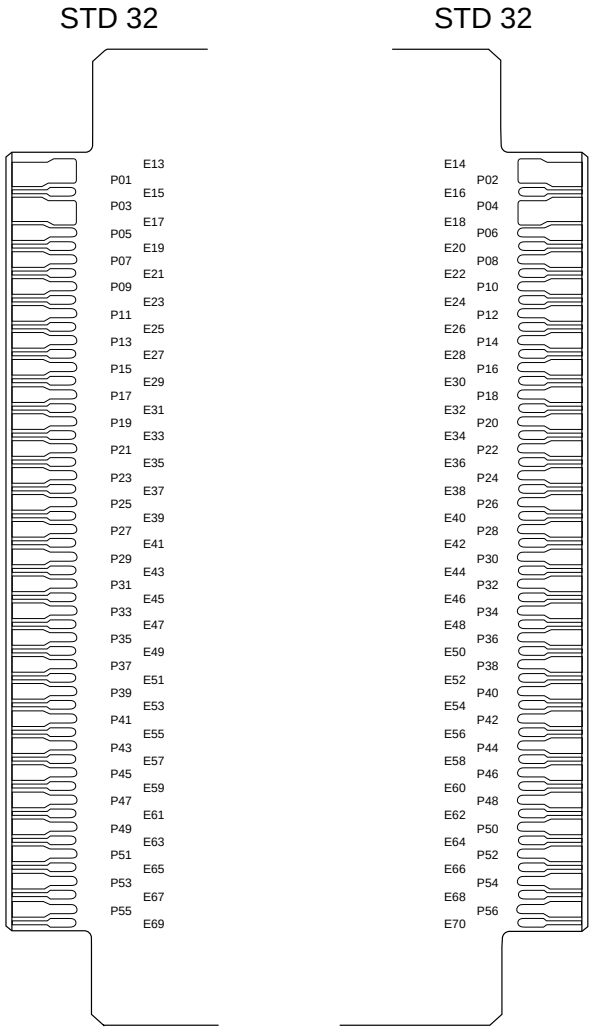


Figure B–1. P/E Connector Pinout.

Table B-3
J1-J4 RS-232-C Pin Assignments.

ZT 8841			ZT 90014 ZT 90027		OTHER EQUIPMENT		
DCE OPERATION	DTE OPERATION	J1-4 PIN #	P1 PIN #	J1 PIN #	PIN #	DCE OPERATION	DTE OPERATION
PG (GND)	PG (GND)	1	1	1	1	PG (GND)	PG (GND)
RXD (IN)	TXD (OUT)	3	3	2	2	RXD (IN)	TXD (OUT)
TXD (OUT)	RXD (IN)	5	5	3	3	TXD (OUT)	RXD (IN)
+5V	+5V	6	6	16	16	+5V	+5V
CTS (IN)	RTS (OUT)	7	7	4	4	CTS (IN)	RTS (OUT)
RTS (OUT)	CTS (IN)	9	9	5	5	RTS (OUT)	CTS (IN)
DCD (OUT)	DCD (IN)	10	10	18	18	DCD (OUT)	DCD (IN)
DTR (OUT)	DSR (IN)	11	11	6	6	DSR (OUT)	DSR (IN)
RI (OUT)	RI (IN)	12	12	19	19	RI (OUT)	RI (IN)
SG (GND)	SG (GND)	13	13	7	7	SG (GND)	SG (GND)
DSR (IN)	DTR (OUT)	14	14	20	20	DTR (IN)	DTR (OUT)

Note: If the ZT 8841 is configured as DCE, the other equipment must be configured as DTE. In like manner, if the ZT 8841 is configured as DTE, other equipment must be configured as DCE.

P is the 14-pin end of the ZT 90014 and ZT 90027 cables that connects to connectors J1-J4 on the ZT 88CT41.

J1 is the 25-pin D-type shell connector on the other end of these two cables. Table B-3 above shows the pin-to-pin conversion.

Table B-4
J1-J4 RS-422/485 Pin Assignments.

ZT 8841		ZT 90014 ZT 90027		OTHER EQUIPMENT	
SIGNAL	J1-4 PIN #	P1 PIN #	J1 PIN #	PIN #	SIGNAL
SDA (TXD-)	1	1	1	1	SDA
SDB (TXD+)	2	2	14	14	SDB
RSA (RTS-)	3	3	2	2	RSA
RSB (RTS+)	4	4	15	15	RSB
GND	7	7	4	4	GND
GND	8	8	17	17	GND
CSB (CTS+)	11	11	6	6	CSB
CSA (CTS-)	12	12	19	19	CSA
RDB (RXD+)	13	13	7	7	RDB
RDA(RXD-)	14	14	20	20	RDA

Note: P is the 14-pin end of the ZT 90014 and ZT 90027 cablesthat attaches to ZT 8841 connectors J1-J4.

J1 is a 25-pin D-type shell connector located at the opposite end of the cables.

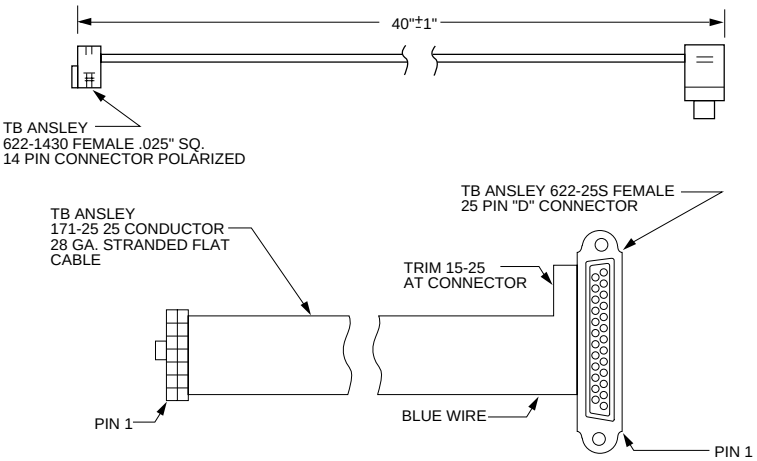
Table B-5
J5 Pin Assignments.

J5 Pin #	Name	Description
2	$\overline{FP1}$	8259A Interrupt
4	$\overline{FP3}$	COM2 Interrupt
6	$\overline{FP5}$	COM4 Interrupt
8	$\overline{FP6}$	COM1 Interrupt
10	$\overline{FP7}$	COM3 Interrupt
1,3,5 7,9	GND	Ground

Cable Specifications

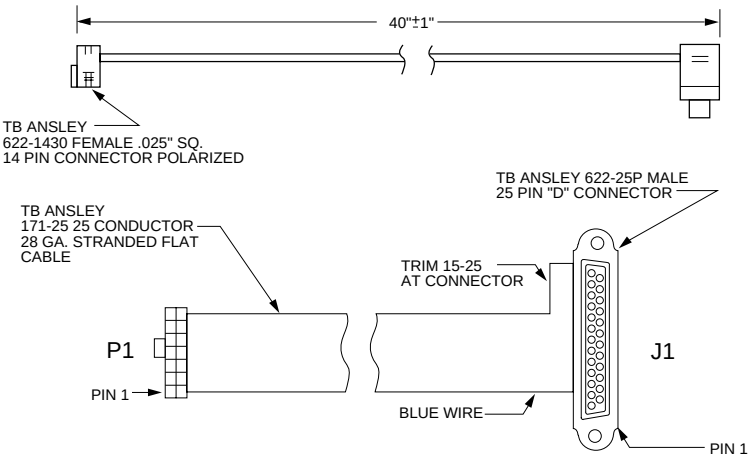
In compliance with EIA Standard RS-232-C for DCE operation, Ziatech offers a 1 meter flat ribbon serial cable compatible with the ZT 8841 (part number ZT 90014). This cable is terminated with an RS-232-C female 25-pin subminiature D-type connector for freestanding or panel mount use. A cable drawing appears on page B-13, and the cable pin list is given on page B-10.

For DTE (PC compatible) operation, Ziatech offers a 1 meter flat ribbon serial cable compatible with the ZT 8841 (part number ZT 90027). This cable is terminated with an RS-232-C male 25-pin subminiature D-type connector for freestanding or panel mount use. A cable drawing appears on page B-14, and the cable pin list is given on page B-10.



P1	J1	P1	J1
1	1	8	17
2	14	9	5
3	2	10	18
4	15	11	6
5	3	12	19
6	16	13	7
7	4	14	20

Figure B–2. ZT 90014 Rev. 0 Cable Drawing.



P1	J1	P1	J1
1	1	8	17
2	14	9	5
3	2	10	18
4	15	11	6
5	3	12	19
6	16	13	7
7	4	14	20

Figure B–3. ZT 90027 Rev. A Cable Drawing.

Card Dimensions & Weight

Width: 113.8 mm (4.5 inches)

Height: 165.1 mm (6.5 inches)

Weight: 156 g (5.5 oz)

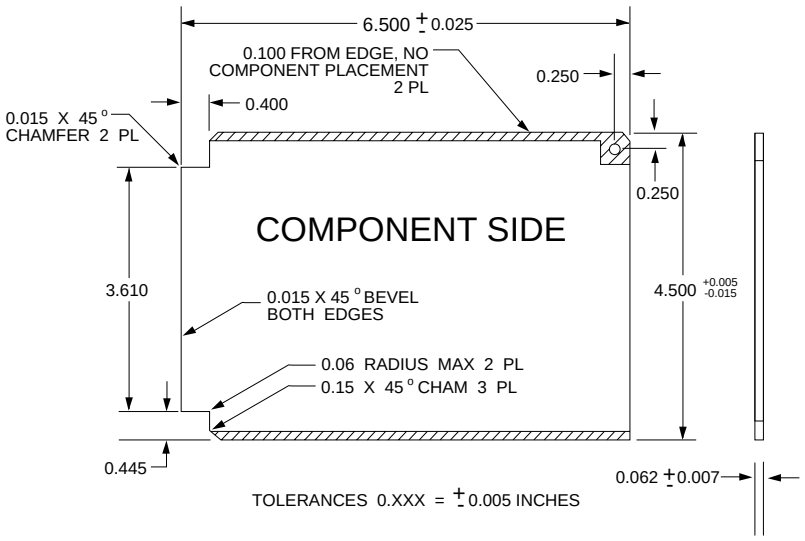


Figure B–4. ZT 8841 Card Dimensions.

Appendix C

QUICK REFERENCE

Contents **Page**

OVERVIEW C-1
I/O MAP C-1
J1-J4 SERIAL PIN ASSIGNMENTS C-5
J5 INTERRUPT PIN ASSIGNMENTS C-7

OVERVIEW

The I/O map tables and connector pin assignments in this appendix are taken from the body of the manual for easy access.

I/O MAP

Note: A dagger (†) indicates the standard default jumper location. W65-69 contain spare jumpers that can be used.

Table C-1
Base Address Jumpers.

JUMPER #	DESCRIPTION
W1-W6	Select the value of I/O port address bits A10-A15 that make up the upper six bits of the ZT 8841 Base Address. W1-6 correspond to address bits A15-10, respectively. When installed, the address line must be false, and when removed, the address line must be true to select the board. Factory default = W1-6 installed. See pages 2-6 to 2-8 for more information.

Jumper	Address Line
W1 out	A15=1
† W1 in	A15=0
W2 out	A14=0
† W2 in	A14=1
W3 out	A13=0
† W3 in	A13=1
W4 out	A12=0
† W4 in	A12=1
W5 out	A11=0
† W5 in	A11=1
W6 out	A10=0
† W6 in	A10=1

† Factory default jumper configuration.

Table C-1
Base Address Jumpers (continued).

JUMPER #	DESCRIPTION
W9, W10	Select the COM and PIC I/O addresses based on address bits A3-A9 that make up the lower seven bits of the board's Base Address. ††The selections are:

W10	W9	COM1	COM2	COM3	COM4	PIC
in	in	3F8-3FFh	2F8-2FFh	3E8-3EFh	2E8-2EFh	020-027h
out	in	3E0-3E8h	2F8-2FFh	3E8-3EFh	2E8-2EFh	020-027h
in	out	3E0-3E8h	2F8-2FFh	3E8-3EFh	2E8-2EFh	0A8-0AFh
† out	out	3E0-3E8h	2E0-2E8h	3E8-3EFh	2E8-2EFh	0A8-0AFh

† Factory default jumper configuration.

†† W65-W69 contain spare jumpers that can be used as needed.

For example, if COM1 is desired to have a Base Address of C3F8h, remove W1 and W2 and install W3-6, then install both W9 and W10. Use your Base value in the following table to access the individual registers. Note that each UART and PIC has its own unique Base Address depending on the combination of W9 and W10 selected in conjunction with W1-6.

Table C-2
I/O Registers.

UART BASE +	**I/O Read Register**	**I/O Write Register**
0000	Receive Buffer	Transmitter Buffer
0001	Interrupt Enable	Interrupt Enable
0002	Interrupt Ident.	---
0003	Line Control	Line Control
0004	Modem Control	Modem Control
0005	Line Status	Line Status
0006	Modem Status	---
0007	Scratchpad	Scratchpad
PIC BASE +	**I/O Read Register**	**I/O Write Register**
0000	IRR, ISR, IL	ICW1, OCW2, OCW3,
0001	IMR	ICW2,3,4, OCW1

J1-J4 SERIAL PIN ASSIGNMENTS

Connectors J1-J4 can be used for RS-232-C communications (COM1-4). Two RS-232-C connectors (J2, J3) can be used for RS-422/485 communications (COM2,3). Depending on the jumper configuration described in Appendix A, the pinout is determined by the following tables:

Table C-3
J1-J4 RS-232-C Pin Assignments.

ZT 8841			ZT 90014 ZT 90027		OTHER EQUIPMENT		
DCE OPERATION	DTE OPERATION	J1-4 PIN #	P1 PIN #	J1 PIN #	PIN #	DCE OPERATION	DTE OPERATION
	PG (GND)	1	1	1	1	PG (GND)	PG (GND)
	TXD (OUT)	3	3	2	2	RXD (IN)	TXD (OUT)
	RXD (IN)	5	5	3	3	TXD (OUT)	RXD (IN)
	+5V	6	6	16	16	+5V	+5V
	RTS (OUT)	7	7	4	4	CTS (IN)	RTS (OUT)
	CTS (IN)	9	9	5	5	RTS (OUT)	CTS (IN)
	DCD (IN)	10	10	18	18	DCD (OUT)	DCD (IN)
	DSR (IN)	11	11	6	6	DSR (OUT)	DSR (IN)
	RI (IN)	12	12	19	19	RI (OUT)	RI (IN)
	SG (GND)	13	13	7	7	SG (GND)	SG (GND)
	DTR (OUT)	14	14	20	20	DTR (IN)	DTR (OUT)

Note: If the ZT 8841 is configured as DCE, the other equipment must be configured as DTE. In like manner, if the ZT 8841 is configured as DTE, other equipment must be configured as DCE.

P is the 14-pin end of the ZT 90014 and ZT 90027 cables that connects to connectors J1-J4 on the ZT 88CT41.

J1 is the 25-pin D-type shell connector on the other end of these two cables. Table B-3 above shows the pin-to-pin conversion.

Table C-4
J2-J3 RS-422/485 Pin Assignments.

ZT 8841		ZT 90014 ZT 90027		OTHER EQUIPMENT	
SIGNAL	J1-4 PIN #	P1 PIN #	J1 PIN #	PIN #	SIGNAL
SDA (TXD-)	1	1	1	1	SDA
SDB (TXD+)	2	2	14	14	SDB
RSA (RTS-)	3	3	2	2	RSA
RSB (RTS+)	4	4	15	15	RSB
GND	7	7	4	4	GND
GND	8	8	17	17	GND
CSB (CTS+)	11	11	6	6	CSB
CSA (CTS-)	12	12	19	19	CSA
RDB (RXD+)	13	13	7	7	RDB
RDA(RXD-)	14	14	20	20	RDA

Note: P is the 14-pin end of the ZT 90014 and ZT 90027 cables that attaches to ZT 8841 connectors J1-J4.

J1 is a 25-pin D-type shell connector located at the opposite end of the cables.

J5 INTERRUPT PIN ASSIGNMENTS

Table C-5
J5 Interrupt Pin Assignments.

J5 Pin #	Name	Description
2	$\overline{FP1}$	8259A Interrupt
4	$\overline{FP3}$	COM2 Interrupt
6	$\overline{FP5}$	COM4 Interrupt
8	$\overline{FP6}$	COM1 Interrupt
10	$\overline{FP7}$	COM3 Interrupt
1,3,5 7,9	GND	Ground

Appendix D

CUSTOMER SUPPORT

Contents **Page**

OVERVIEW D-1
ZT 8841 REVISIONS D-2
Revision History D-2
ZT 8841 Revision 0.1 . . . . . D-2
ZT 88CT41 Revision A — 4/23/91 . . . . . D-2
ZT 88CT41 Revision A.1 — 8/7/91 . . . . . D-3
ZT 88CT41 Revision A.2 — 12/11/91 . . . . . D-3
RELIABILITY D-4
TECHNICAL ASSISTANCE D-6
RETURNING THE ZT 8841 FOR SERVICE D-7

OVERVIEW

This appendix offers technical assistance for ZT 8841 users. Also included in this appendix are the ZT 8841 revision history, a discussion of reliability considerations, the Ziatech warranty, and information about returning products for service.

ZT 8841 REVISIONS

Revision History

Functional changes that affect a product's backward compatibility are denoted by incrementing the major (letter) revision level (for example, revision A incrementing to revision B).

Changes that have no effect on backward compatibility, such as increased functionality and PCB modifications to eliminate wire connections, are indicated by incrementing the minor (dot) revision level (for example, revision A.1 incrementing to revision A.2).

ZT 8841 Revision 0.1

The ZT 8841 Revision 0.1 was the first production release of the ZT 8841.

ZT 88CT41 Revision A — 4/23/91

The ZT 8841 Revision A (4/23/91) incorporated the following changes for ZT 88CT41 Revision A boards:

- Functionality was increased with the addition of jumpers W86 through W91 (see pages A-21 through A-23 for descriptions).
- Pinouts of connectors J1 through J4 were modified. Pin 6 is no longer connected to +12 V but is now tied to +5 V.
- Power requirements for opto-isolated RS-485 systems were changed; +12 V is no longer required.
- The RS-485 receiver circuit was improved. 560 Ω pull-up/pull-down resistors have been added to the RS-485 receivers to prevent their lines from floating when their drivers are disabled. These socketed resistors are designated RP-6 for J3 and RP-7 for J2.

If multiple ZT 88CT41s are linked together, only one board should have the resistors loaded. RP-6 and RP-7 should be removed from all other boards.

– Revision A replaces the STD-80 card-edge connector with an STD 32 P/E connector. This allows access to the new IRQx signal and added power connections. The P/E connector is 100% backward compatible with the old STD-80 connector and can be used trouble-free in older systems.

ZT 88CT41 Revision A.1 — 8/7/91

The ZT 8841 Revision A.1 (8/7/91) incorporated no functional changes.

ZT 88CT41 Revision A.2 — 12/11/91

The ZT 8841 Revision A.2 (12/11/91) incorporated no functional changes.

RELIABILITY

Ziatech has taken extra care in the design of the ZT 8841 to ensure reliability. The three major ways in which reliability is achieved are:

1. The product was designed in top-down fashion, utilizing the latest in hardware and software design techniques, so that unwanted side effects and unclean interactions between parts of the system have been eliminated.
2. Ziatech tests each board by exercising its functions, burns it in under power, and retests it to ensure that the infant mortality phase is passed before the product is shipped.
3. Ziatech maintains a lifetime data base on each ZT 8841 and on the parts used on each board. Any negative trends in reliability are spotted and Ziatech's suppliers are informed and/or changed.

ZIATECH WARRANTY

Warranty information for Ziatech products is available at Ziatech's website, located at http://www.ziatech.com.

TECHNICAL ASSISTANCE

If you have a technical question, please call Ziatech's Customer Support Service at one of the following numbers:

Corporate Headquarters: (805) 541-0488
(805) 541-5088 (FAX)

If you have a sales question, please contact your local Ziatech Sales Representative.

RETURNING THE ZT 8841 FOR SERVICE

If a product is damaged in shipment, contact Ziatech Corporation. Call (805) 541-0488 for a Return Material Authorization (RMA) number. Under no circumstances should the product be returned without an RMA number. We will need the following information to expedite the shipment of the repaired board or a replacement to you:

1. Your company name & address for invoice
2. Shipping address & phone number
3. Product I.D. number
4. Purchase order number for service charge (if not under warranty)

If a product must be returned for service, determine if the unit is under warranty and obtain an RMA number as described above. Pack the ZT 8841 in **anti-static** material and ship in a sturdy cardboard box with enough packing material to adequately cushion the board. Any product retirned to Ziatech improperly packed will immediately void the warranty for that particular product. Mark the RMA number clearly on the outside of the box before returning.

Appendix E

GLOSSARY

baud rate The transfer rate of serial data, expressed in bits per second.

cascade address Used by the Master PIC to direct a Slave PIC to output a vector. On the STD bus this address is multiplexed over address lines A8-10 during the interrupt acknowledge cycle performed by the processor.

CMOS Complementary Metal Oxide Semiconductor. Provides low power density and high noise immunity. The ZT 88CT41 uses TTL compatible CMOS devices to ensure compatibility with non-CMOS boards.

CNTL Control. This signal serves as an auxiliary timing signal or as an interrupt signal on the STD bus. When used as an interrupt signal, Ziatech calls this signal INTRQ2 as defined in the Bus Practice section of the *STD 80 Series Bus Specification*, version 2.3.

DCE Data Communication Equipment. Defined by the RS-232-C specification, denotes the sex and pin assignments associated with the serial connector.

DDP Device Driver Package. This software driver package can be used to support the ZT 8841. Complete buffering and software handshake protocols have been implemented.

DTE Data Terminal Equipment that denotes the sex and pin assignments associated with the serial connector. Defined by the RS-232-C specification.

EIA Electronics Industries Association. A standards group that has defined the RS-232-C and RS-422/485 serial communications.

FIFO First In First Out. FIFO capability will allow a UART to buffer up to four bytes of serial data before interrupting the processor, thereby decreasing the number of interrupts from four to one. This feature is included in the Intel 82510 UARTs that can be ordered as options on the ZT 8841.

IOEXP Input/Output Expansion. An STD-80 control signal. Determines the validity of an I/O access to the board.

INTRQ* An interrupt request on the STD bus. Used to interrupt the processor when service is needed (see *INTRQ1* and *CNTL*).

INTRQ1* An interrupt signal on the STD bus defined by version 2.3 of the *STD 80 Series Bus Specification* (STD-8088 Bus Specification).

ISR Interrupt Service Routine. A software routine used to service the device that caused the hardware interrupt.

LSB Least Significant Bit.

LSTTL Low power Schottky Transistor Transistor Logic.

master PIC The master Programmable Interrupt Controller takes interrupt inputs directly from interrupting devices and/or from slave PICs. The master interrupts the processor and provides a vector for interrupting devices. If a slave PIC is interrupting, then the master directs which slave PIC should output a vector by issuing a cascade address.

MCS-51 A protocol in which a master on a serial network can address a slave in order to send or receive data. This protocol is built into the Intel 82510 UART that can be purchased as an option on the ZT 8841.

MSB Most Significant Bit.

multidrop Multiple connections made to a common wire.

optical isolation Optical isolation prevents 600 V noise spikes on the RS-422/485 serial cable from entering the computer. This is an option to the ZT 8841.

PAL Programmable Array Logic. Uses a fixed OR array fed by the output of a programmable AND array to produce "sum of products" logic expressed in a Boolean transfer function. On the ZT 8841, PALs decode address bits for UART selection and for wait states (see *WAITRQ**).

PIC Programmable Interrupt Controller. Typically an Intel 8259A device that prioritizes eight inputs and generates one interrupt output. When the interrupt output is acknowledged, the 8259A generates a vector that points to the interrupt service routine. Also see *vector, ISR, master PIC, slave PIC.*

RS-232-C Serial data protocol defined by the EIA for single-ended communications. Maximum specified length is 50 feet at a baud rate of 20,000 bits per second.

RS-422/485 Serial data protocol defined by the EIA for differential communications. Maximum specified length is 4,000 feet.

Schottky TTL Provides high speed, low noise, and compatibility with standard TTL.

slave PIC The slave Programmable Interrupt Controller takes interrupt inputs directly from interrupting devices. When the master PIC issues a cascade address that matches the slave PIC, the slave PIC will output a vector pointing to the interrupt service routine.

TTL Transistor Transistor Logic.

UART Universal Asynchronous Receiver Transmitter. Converts parallel data to or from serial data. The rate of conversion is software selectable and defined in the number of bits per second (see *baud rate*). The UARTs used on the ZT 8841 are either Intel 82050 or 82510 devices that are IBM 8250 compatible.

vector As used here, vector refers to the interrupt number returned by the Programmable Interrupt Controller (PIC) that is multiplied by four for the processor to obtain the interrupt service routine (ISR) address.

WAITRQ* This is a backplane signal, asserted by the ZT 8841 to force the STD processor to wait. The ZT 8841 needs one wait state for 5 MHz operation and two wait states for 8 MHz operation. Wait states are jumper selectable.

XTP Extended Temperature operation. When ordered, the ZT 88CT41 can operate in ambient temperatures of -40 to +85˚ Celsius.

INDEX

- A -

ambient temperature requirements 2-4, B-2
application notes
  interrupts 2-26
  MCS-51 2-26
  PIC (8259A) 2-26
  UART (82510) 2-26
asynchronous communication controller (ACC) 6-1
automatic rotating mode 5-10

- B -

backplane interrupts 3-13, 3-20, A-9
Base Address C-1
baud rate 6-23, E-1
  82050 serial port 6-23
block diagram 1-5
board Base Address A-5
board dimensions and weight B-15

- C -

cables
  ZT 90014 RS-232-C female 1-4, B-12
  ZT 90027 RS-232-C male 1-4, B-12
cable specifications B-12
card dimensions and weight B-15

cascade address 1-9, 3-21, E-1
CMOS E-1
CNTL (INTRQ2*) 3-13, A-9, E-1
COM address configurations 2-8, A-6, C-1, C-3
COM interrupt enabling 3-19
commonly asked questions 3-2
configuration, default 2-19, A-1
connectors
  J5, frontplane interrupt 3-20, A-19, B-8, C-7
  J1-J4, serial 1-4, B-8, C-5
  J2-J3, serial C-6
cooling the ZT 8841 2-4

- D -

damage during shipping 2-2
data communication equipment (DCE) 1-4, 3-7, 3-22, A-16, E-1
data terminal equipment (DTE) 1-4, 2-19, 3-6, 3-22, A-16, E-2
DC operating characteristics B-2
DDP 2-10, E-2
default Base Address 2-8, C-1
default COM address configurations 2-21, C-1
default jumper configuration 2-19, A-3
diagnostics 3-25, 6-6
dimensions and weight of ZT 8841 . . . . . B-15

- E -

edge-triggered mode 5-13
EIA standards 3-5, E-2
electrical specifications B-2
Electronics Industries Association E-2
enhanced serial communications
  FIFO 1-8, 2-26, E-2
  MSC-51 1-8
  UART (82510) 1-8, 2-26, 3-17

environmental specifications B-7
EOI 5-16

- F -

features of the ZT 8841 1-3
First In First Out (FIFO) 1-3, 1-8, 2-26, E-2
frontplane interrupts 3-14, 3-20, A-19, B-12, C-7
fully nested mode 5-8
functional block diagram 1-5
functional operation 3-15

- G -

getting started 2-2
glossary E-1

- H -

humidity requirements 2-4, B-2

- I -

initial conditions 2-19
initialization command words (ICWs) 5-19
INTA* A-7
interrupt controller 5-2
  modes of operation 5-8
interrupt registers and control logic 5-6
interrupts
  8259A PIC A-15
  application notes 2-26
  backplane 3-13, 3-20, A-9
  cascade 3-21
  causes 3-18
  CNTL* signal 3-20
  conditions 3-18
  enable 3-19

frontplane 3-14, 3-20, A-19, C-7
fundamentals 3-8
INTRQ* signal 3-20
INTRQ1* signal 3-20
PIC (8259A) 1-2, 1-9, 3-9, 3-21
polled 3-11, 3-21, A-7
service routine 3-8, E-3
status 5-14
triggering 5-11
types 3-10
vectored 3-8, 3-10, 3-13, A-7
introduction to the ZT 8841 1-1
INTRQ* 3-13, E-2
INTRQ1* 3-13, E-2
INTRQ*, INTRQ1*, INTRQ2* (CNTL) A-9
inverter (74ACT04) A-23
I/O addressing 2-19, 3-2, 3-16, A-5
Base Address 2-8
I/O registers 2-8, C-4
port selection 2-8
IOEXP 2-19, 3-16, A-6, E-2
IRQx A-21

- J -

J5, frontplane interrupt connector 3-20, A-19, B-8, C-7
J1-J4, serial connectors 1-4, B-8, C-5
jumpers (see also entries under W)
configuration 2-8, A-1
customer jumper locations A-4
factory default configuration 2-19, A-3

- L -

level-triggered mode 5-12
load characteristics (STD) B-4

low power operation 3-24
LSTTL E-3

- M -

MCS-51 2-26
  application notes 2-26
  protocol 1-8, E-3
mechanical specifications B-8
microprocessor compatibility 2-5
modem control 6-6
modes
  automatic rotating mode 5-10
  fully nested mode 5-8
  poll mode 5-11
  special fully nested mode 5-9
  special mask mode 5-11
  specific rotating mode 5-10
mounting the ZT 8841 2-4
mouse operation 3-3
multidrop 3-6, A-14, E-3

- O -

operating requirements 2-4, B-2
operating temperature B-7
operation control words (OCWs) 5-24
optical isolation 1-7, 3-24, E-3
  benefits 3-3
  operating speed 3-4

- P -

PAL E-4
PCI, PCO 2-19
PIC (8259A) 1-2, 1-9, 2-8, 2-26, 3-9, A-6, A-7, C-3, E-4
  application note 2-26

cascade address 3-21
slave mode 3-21
polled interrupts 3-11
poll mode 5-11
power dissipation requirements 2-4
power requirements B-2
priorities (operation of the interrupt controller) 5-8
processor compatibility 1-2, 2-5
product definition 1-2
programmable interrupt controller (PIC) 1-9, 2-26, 3-9, 5-2, E-4
programming 5-16
registers 2-8

- Q -

quick reference C-1

- R -

receiver 6-5
reliability of the ZT 8841 . . . . . D-4
requirements, system 2-4, B-2
reset 6-21
Return Material Authorization (RMA) D-7
revision history D-2
RS-422/485 . . . . . 1-3, 1-6, E-4
drivers and receivers 1-6, 1-7, 3-23
features 3-6
implemented signals 1-7
operating temperature 3-23
optical isolation 1-7, 3-24
pinout C-6
power supply A-20
serial communications 6-25
standard 3-5
RS-232-C 1-3, 1-6, E-4
cables B-12

DCE 3-22, A-16
drivers and receivers 1-6
DTE 3-22, A-16
implemented signals 1-6
pinout C-5
standard 3-5
vs. RS-422/485 3-5

- *S* -

Schottky TTL E-4
serial communications (see also I/O) 1-2, 1-4, 1-8, 3-5, 3-17, 6-1
baud rate 6-23
data format 6-22
interrupt communication 6-25
interrupt control 6-6
modem control 6-6
operation 6-21
polled communication 6-25
programmable registers 6-7
programming 6-26
receiver 6-5
reset 6-21
RS-422/485 6-25
transmitter 6-5
serial controller (ACC) 6-1
serial data format 6-22
service to the ZT 8841 . . . . . D-7
shipping damage 2-2
software driver 4-1
objectives 4-1
program code 4-2
special fully nested mode 5-9
special mask mode 5-11
specific rotating mode 5-10
STD-80 compatibility 2-5

STD interface 3-16
STD load characteristics B-4
storage temperature B-7
supply current requirements 2-4, B-2
supply voltage requirements 2-4, B-2
system requirements 2-4, B-2

- T -

technical assistance D-6
theory of operation 3-2
transmitter 6-5
TTL E-5

- U -

UART (82050) 2-8, 2-26, 3-17, B-2
UART (82510) 1-8, 2-26, 3-17, 3-24, B-2, E-2, E-3
unpacking 2-2

- V -

vector 3-9, 3-11, 3-13, E-5
vectored interrupts 3-8, 3-13

- W -

W1-W6 ..... A-5
W7-W8 ..... A-6
W9-W10 ..... A-6, C-3
W11 ..... A-7
W12-W13 ..... A-7
W14-W15 ..... A-8
W16-W20 ..... A-9
W21-W22, W25-W26 ..... A-10
W23-W24, W27-W28 3-19
W23-W27 ..... A-13

W24-W28 . . . . . A-14
W29-W36 . . . . . A-15
W37-W43 . . . . . A-10
W44-W49 . . . . . A-16
W50, W64, W76, W83 . . . . . A-17
W51-W57 . . . . . A-10
W58-W63 . . . . . A-16
W64 . . . . . A-17
W65-W69 . . . . . A-19
W70-W75 . . . . . A-16
W76 . . . . . A-17
W77-W82 . . . . . A-16
W83 . . . . . A-17
W84-W85 . . . . . A-20
W86-W87 . . . . . A-21
W88 . . . . . A-22
W89 . . . . . A-22
W90 . . . . . A-23
W91 . . . . . A-23
WAITRQ* A-8
wait states A-8
warranty D-5
weight and dimensions of ZT 8841 . . . . . B-15
what's in the box 2-3

- X -

XTP 2-4, 3-3, E-5

- Z -

ZT 8841
  compared to ZT 88CT41 . . . . . 1-1, 3-3, B-2
  vs. ZT 8840 3-2
ZT 88CT41 2-4
  CMOS technology 1-3
  compared to ZT 8841 . . . . . 3-2, B-2

power requirements B-2
revision history D-2
ZT 90014, RS-232-C female cable 1-4, B-12
ZT 90027, RS-232-C male cable 1-4, B-12
ZT 8840 vs. ZT 8841 3-2

ZT M8841
May 20, 1994
For Revision A.2

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP