8-Port RS-232/422/485 Serial Module

LW-M104-SP8A (DX)

User's Manual

(Release 2.1)

LongWin Computer Inc.

3130 De La Cruz Blvd, #97
Santa Clara, CA 95054
(800) 970-6805
www.LWComputer.com

Copyright:

Disclaimer

LongWin Computers, Incorporated makes no representations or warranties with respect to the contents of this manual or of the associated LongWin products, and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. LongWin shall under no circumstances be liable for incidental or consequential damages or related expenses resulting from the use of this product, even if it has been notified of the possibility of such damages. LongWin reserves the right to revise this publication from time to time without obligation to notify any person of such revisions.

Acknowledgements:

AMI is a trademark of Award American Megatrends Inc.

PS/2 and VGA are trademarks of International Business Machines Corporation.

Intel and Pentium are trademarks of Intel Corporation.

Microsoft Windows® is a registered trademark of Microsoft Corp.

All other product names or trademarks are properties of their respective owners.

Revision History

Revision	Description	Date
1.0	Initial Release	2/18/2008
2.0	Reduce I/O Space from 128 Byte to 64 Byte	5/6/2010
2.1	Save Settings in Flash Memory for DX Version	1/10/2011

For more information on LongWin products or technical support and service, please visit our websites at:

http://www.LWConputer.com

This manual is for the ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module***.

P/N: LW-M104-SP8A-M

February 18, 2008

Packing List

Before you begin installing your module, please make sure that the following materials have been shipped:

- *LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*
- *Disk For Utility, Drivers and Technical Information (Online Download)*
- *Data Sheet*
- *Quick Start Guide*

If any of these items are missing or damaged, contact your distributor or sales representative immediately.

Common Model No. List	Description
LW-M104-SP8A-8D	8 Port RS232/RS485 Serial Module DX Version
LW-M104-SP8A-8N	8 Port RS232/RS485 Serial Module
LW-M104-SP8A-4D	4 Port RS232/RS485 Serial Module DX Version
LW-M104-SP8A-4N	4 Port RS232/RS485 Serial Module

Safety and Static-Electricity

FCC

This device complies with the requirements in part 15 of the FCC rules:

Operation is subject to the following two conditions:

1. This device may not cause harmful interference, and
2. This device must accept any interference received, including interference that may cause undesired operation

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications.

Caution !

Operation of this device in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his/her own expense. The user is advised that any equipment changes or modifications not expressly approved by the party responsible for compliance would void the compliance to FCC regulations and therefore, the user's authority to operate the equipment.

Warning !

There is a danger of a new battery exploding if it is incorrectly installed. Do not attempt to recharge, force open, or heat the battery. Replace the battery only with the same or equivalent type recommended by the manufacturer. Discard used batteries according to the manufacturer's instructions.

Please plug and unplug parts under the condition without power. Make sure power is turned off before setting boards in order to avoid the damage on the sensitive parts caused by power impact.

Acting !

Modern electrical parts are very sensitive to static electricity. Before setting boards, please put them into anti-static pads or bags. It has better to wear anti-static bangle or gloves whenever taking the boards.

Additional Information and Assistance

Visit the LongWin web site at **http://www.LWComputer.com** where you can find the latest information about the product.

Contact your distributor, sales representative, or LongWin's customer service center for technical support if you need additional assistance.

Please have the following information ready before you call:

- *Product name and serial number*
- *Description of your peripheral attachments*
- *Description of your software (operating system, version, application software, etc.)*
- *A complete description of the problem*
- *The exact wording of any error messages*

Contents

List of Figures

List of Tables

Chapter 1

General Information

This chapter gives background information on the LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module.

Information includes:

- *Introduction*
- *Features*
- *Specifications*
- *Physical Dimension*

Chapter 1. General Information

1.1. Introduction

The ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** features lots of functions such as high transmission speed of 921.6 kbps, software Interrupt programming, 16C554 UART compatible, 16-Byte FIFO and more. It is designed to meet the size, power consumption, temperature range, quality, and reliability demands of embedded system applications, and satisfies various industrial and consumer applications. This solid, general purpose digital I/O module works on a single +5 Volt power supply at -40°C to +85°C Extended temperature.

The ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** contains eight independent channels and provides effective RS-485, RS-422 and RS-232 multipoint communication. Each channel may be configured to any mode. It allows for connection to devices utilizing the RS-232/422/485 electrical interface, such as modems, data-entry terminals, plotters, and etc.

RS-422/485 communications commonly use the Half-Duplex mode since they share only a single pair of wires. The ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** supports RS-422/485 in Half-Duplex communications mode with a 2-wire cable connection that use differential balanced transceivers for increased range and noise immunity. The RS-422/485 specification defines a maximum of 32 devices on a single line.

The board was designed for industrial applications and can be installed in PC/104 bus connector.

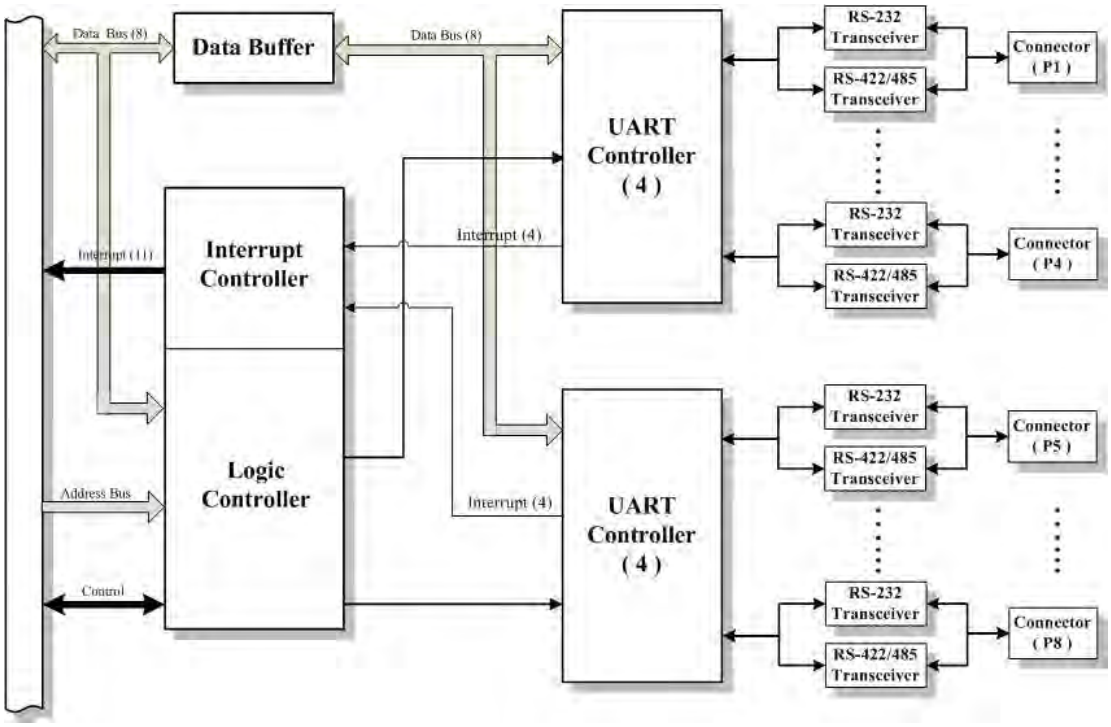

Figure 1. LW-M104-SP8A (DX) Block Diagram

The ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** occupies 128-Byte consecutive I/O locations. The base address is selectable via jumpers.

The module provides access to interrupt levels IRQ3-7, 9-12, 14 and 15 on the PC bus for real-time background applications. Using interrupts allows "background" operation, where I/O can be performed while the PC is executing another task, such as running an unrelated applications program. This feature is useful for performing I/O at a controlled rate. Since a counter output can be used to drive the interrupt request pin on the input connector at a periodic rate for a user supplied interrupt service routine that performs whatever function is necessary in response to the interrupt.

The interrupt is enabled and set by software setting the relative register.

1.2. Features

- *8 Standard RS-232/422/485 Serial Channel*
- *High Transmission Speeds Up To 921.6 Kbps*
- *Software Programmable Base Address*
- *Software Programmable Interrupt for Each Port*
- *Flexible Sharing Interrupt*
- *Settings Saved in Flash Memory (DX Version Only)*
- *16C554 Compatible UART*
- *16-Byte FIFO Per Channel*
- *Individual Selection of RS-232/422/485 Modes Per channel*
- *Full Complement of Modem Control Signals*
- *Automatic RTS/CTS (Hardware) Flow Control*
- *Automatic XON/XOFF (Software) Flow Control*
- *Built-in Termination Resistors*
- *16 I/O Base Address Options*
- *-40 ~ 85° C Extended Operating Temperature*
- *+5V Single Power Supply*

1.3. Specifications

The following section provides technical specifications for the ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module***.

Bus Types

- PC/104 (IEEE-996) (ISA Compatible)

Serial Channel

- 8 Standard RS-232/422/485 Serial Channel
- High Transmission Speeds Up To 921.6 Kbps
- Software Programmable Interrupt for Each Port

- 16C554 Compatible UART
- 16-Byte FIFO Per Channel
- Individual Selection of RS-232/422/485 Modes Per channel
- Full Complement of Modem Control Signals
- Character Length: 5,6,7, or 8-bits
- Parity: Even, Odd, or None
- Stop Interval: 1, 1.5, or 2-bits
- Automatic RTS/CTS (Hardware) Flow Control
- Automatic XON/XOFF (Software) Flow Control

RS-422/485

- Half-Duplex Transmission
- Point to Point or Multi-drop
- Built-In Termination Resistors

I/O Address Space and Base Address

- Software Programmable Base Address
- 64-Byte Consecutive Series I/O Addresses Space
- 16 Base Addresses Option (Can be customized)
- 0x100, 0x180, 0x200, 0x280, 0x300, 0x380, 0x400, 0x500, 0x600, 0x700, 0x800, 0x1500, 0x3200, 0x4200, 0x5200, 0x6200 (Hexadecimal)

Interrupt

- Software Programmable Interrupts
- Independent Interrupt Setting for Each Channel
- Flexible Sharing Interrupt
- 11 Interrupt Sources of IRQ3 to IRQ7, IRQ9 to IRQ12, IRQ14 and IRQ15

Serial Port Interface

- Four 20-Pin 0.100" Pitch IDC Connector

Mechanical and Environmental

- Dimensions: 3.6 x 3.8 inch (90 x 96 mm)
- Operating Temperature: -40 ~ 85° C Extended Temperature
- Storage Temperature: -55° to +120° C
- Operating Humidity: 0% ~ 95% Relative Humidity, No Condensing
- Weight: 60 gm

- Shock and Vibration: Tested to MIL-STD 202F, Method 213B, Table 213-I, Condition A (three 50G shocks in each axis) and MIL-STD 202F, Method 214A, Table 214-I, Condition D (11.95B random vibration, 100 Hz to 1000 Hz for 5 minutes per axis).
- PC/104 Expansion Bus: 104-Pin 0.1 Inch Pitch Stackthrough Connector. Electrical Specifications Equivalent to ISA Bus Specification

Power Requirement

- Voltage: +5V +/- 5%
- Current: 0.3A @ 5V (Typical)

1.4. Physical Dimension

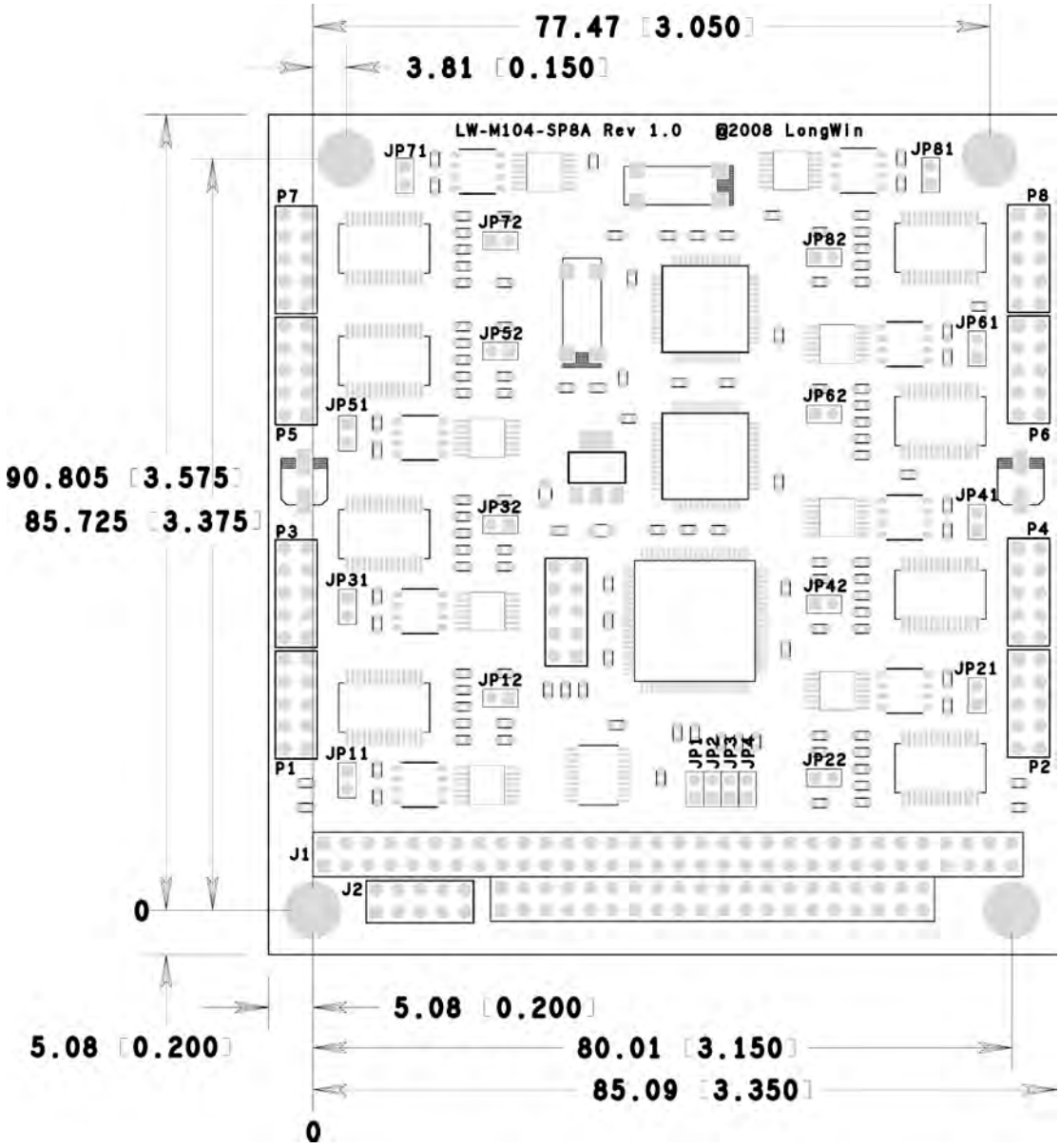

Figure 2. LW-M104-SP8A (DX) Board Dimension (Millimeter)

Chapter 2

Installation

This chapter explains the setup procedures of LW-M104-SP8A (DX) serial module, including instructions on setting jumpers and wiring connectors. Be sure to read all safety precautions before you begin the installation procedure.

Information includes:

- *Jumpers*
- *Connectors*
- *Power Connector (J2 Option)*
- *Serial Interface Connectors*
- *PC/104 Expansion Bus (J1)*

Chapter 2. Installation

2.1. Jumpers

The ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** has some jumpers that allow you to configure your serial module to suit your application. This section of the manual is intended to provide the necessary information to configure the module for the desired mode of operation. The figures and table below list the functions of the various jumper settings.

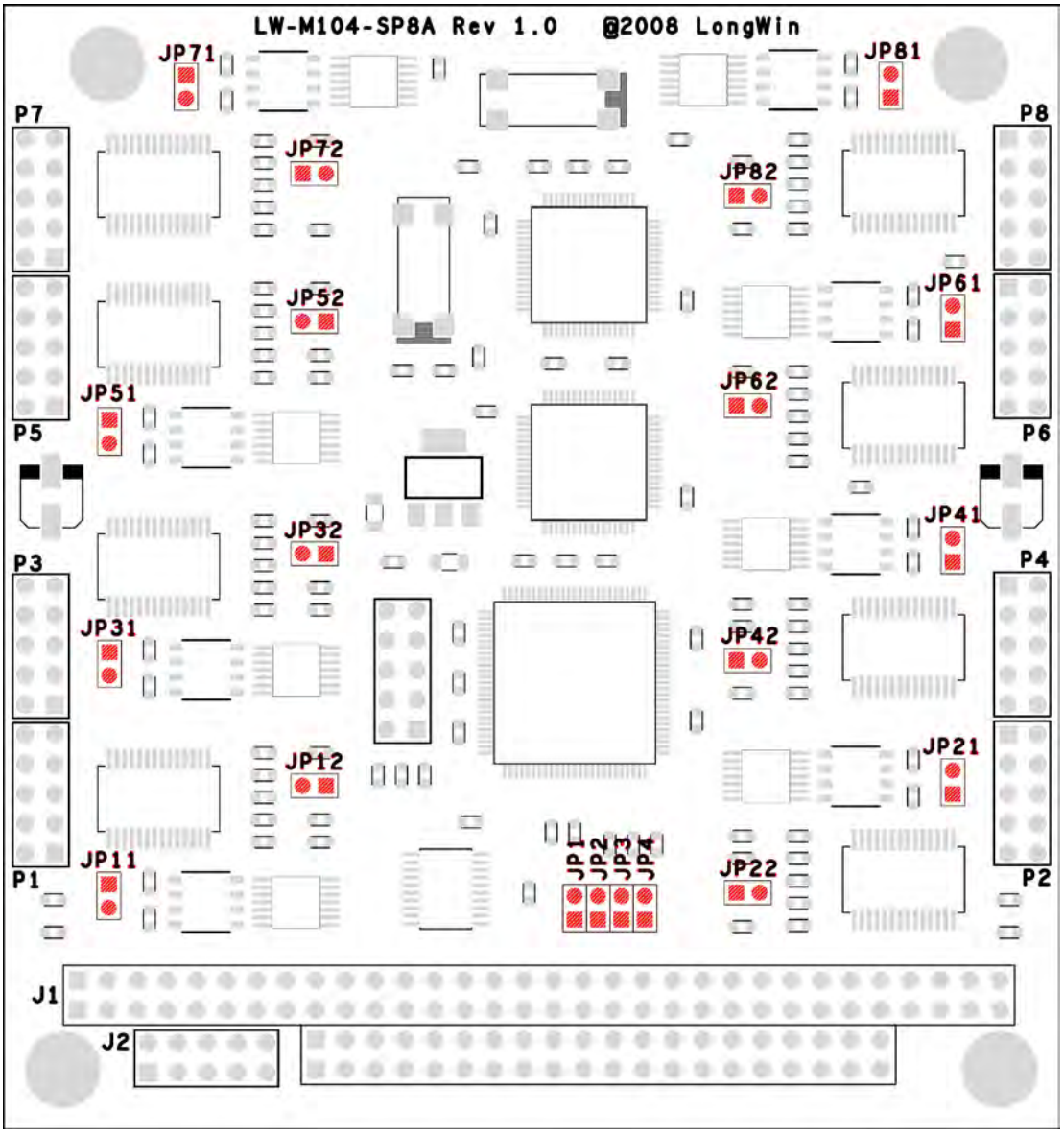

Figure 3. Jumpers Location

2.1.1. I/O Base Address Jumper

The LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module uses 128-Byte consecutive series I/O addresses space for configuration and control of the serial interface. To set desired I/O port address, jumpers must be correctly installed on the board. These jumpers are marked JP1 to JP4. Be careful when selecting the base address as some selections conflict with existing PC ports.

The module has jumpers in default positions so that in most cases it requires no special jumpering for normal operation. You can connect the power and peripherals to operate immediately.

IMPORTANT: Carefully review the *Table 1. I/O Base Address Jumper Setting Table* before selecting the I/O base address. If the addresses of two circuits overlap you will experience unpredictable computer behavior.

Table 1. I/O Base Address Jumper Setting Table

Jumper Reference				Base Address
JP4	JP3	JP2	JP1 *	
Open	Open	Open	Open	0x100
Open	Open	Open	Closed	0x180 **
Open	Open	Closed	Open	0x200
Open	Open	Closed	Closed	0x280 **
Open	Closed	Open	Open	0x300 (Default)
Open	Closed	Open	Closed	0x380 **
Open	Closed	Closed	Open	0x400
Open	Closed	Closed	Closed	0x500 **
Closed	Open	Open	Open	0x600
Closed	Open	Open	Closed	0x700 **
Closed	Open	Closed	Open	0x800
Closed	Open	Closed	Closed	0x1500 **
Closed	Closed	Open	Open	0x3200
Closed	Closed	Open	Closed	0x4200 **
Closed	Closed	Closed	Open	0x5200
Closed	Closed	Closed	Closed	0x6200 **

For DX Version Only:

* When JP1 is set to "Open", the system will power up with one of 8 power-up base address options.

When JP1 is set to "Closed", the system will power up with the last saved base address in flash memory. After setting the base address by software, it will be saved automatically in flash memory.

** Useless for DX version.

2.1.2. Serial Protocol Selection

Each of the serial channels on the ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** can be individually configured as RS-232, or as a two wire RS-422/485 interface. To enable a channel for RS-232, simply set the jumper to "Open"; To enable a channel for two-wire RS-422/485, simple set the jumper to "Closed".

2.1.3. RS-422/485 Termination

In RS-422 or RS-485 networks, termination resistors are normally installed at the endpoints of the cables to minimize reflections on the lines (RS-422 terminates at the receive end only). Typically a 120-ohm resistor is across each RS-422/485. If multiple modules are configured in an RS-485 network, only the boards on each end should have the termination in place.

To enable resistor termination for a channel, install jumpers in the locations ("Closed"). Termination is only needed, and should only be used, at the cable endpoints. Enabling these termination resistors at each end of the cable results in effective impedance of 60Ω. Installing termination resistors at additional points in the network may cause overloading and failure of the line drivers due to the lower impedance caused by multiple resistors in parallel.

2.1.4. Setting Jumper

Jumper JPx1 and JPx2 are used to select the protocol and termination for each serial channel, where "x" is 1 to 8 to represent the channel number. Two configurations are shown for RS-232 and RS-422/485, with and without line termination.

Table 2. RS-232 and RS-422/485 Protocol Selection

Jumper	Function	
	Open	Closed
JPn1	Without Termination (Default)	With Termination
JPn2	RS-232 (Default)	RS-422/485
n = UART Serial Port Channel Number of 1 to 8		

You may configure your module to match the needs of your application by setting jumpers. A jumper is a metal bridge used to close an electric circuit. It consists of two metal pins and a small metal clip (often protected by a plastic cover) that slides over the pins to connect them. To "Closed" a jumper, you connect the pins with the clip. To "Open" a jumper, you remove the clip.

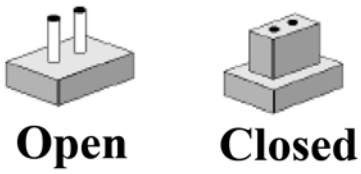

Figure 4. Jumper Setting Illustrating

Jumper pin is 0.1 Inch pitch. The square solder pad identifies as pin 1 of the jumper.

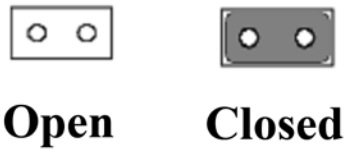

Figure 5. Jumper Schematic Setting Illustrating

A pair of needle-nose pliers may be helpful when working with jumpers. If you have any doubts about the best hardware configuration for your application, contact LongWin before you make any changes.

2.2. Connectors

On board connectors link the ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** to external devices. Generally, you simply need a cable to make connections. Refer to *Figure 6. LW-M104-SP8A (DX) Connectors Location* for the locations of the connectors J1, J2 and P1 to P8. *Table 3. LW-M104-SP8A (DX) Connector List* summarizes the use of the connectors. The interface is described in following section, showing pin assignments, signal definitions. Please refer to relative section for detail.

All the serial interface connectors are dual-row male headers with 0.1 inch pitch for use with flat ribbon IDC female connectors and ribbon cable. Shrouded with center-bump polarized headers are manufacturing options upon required. It will prevent accidentally installing cables backwards.

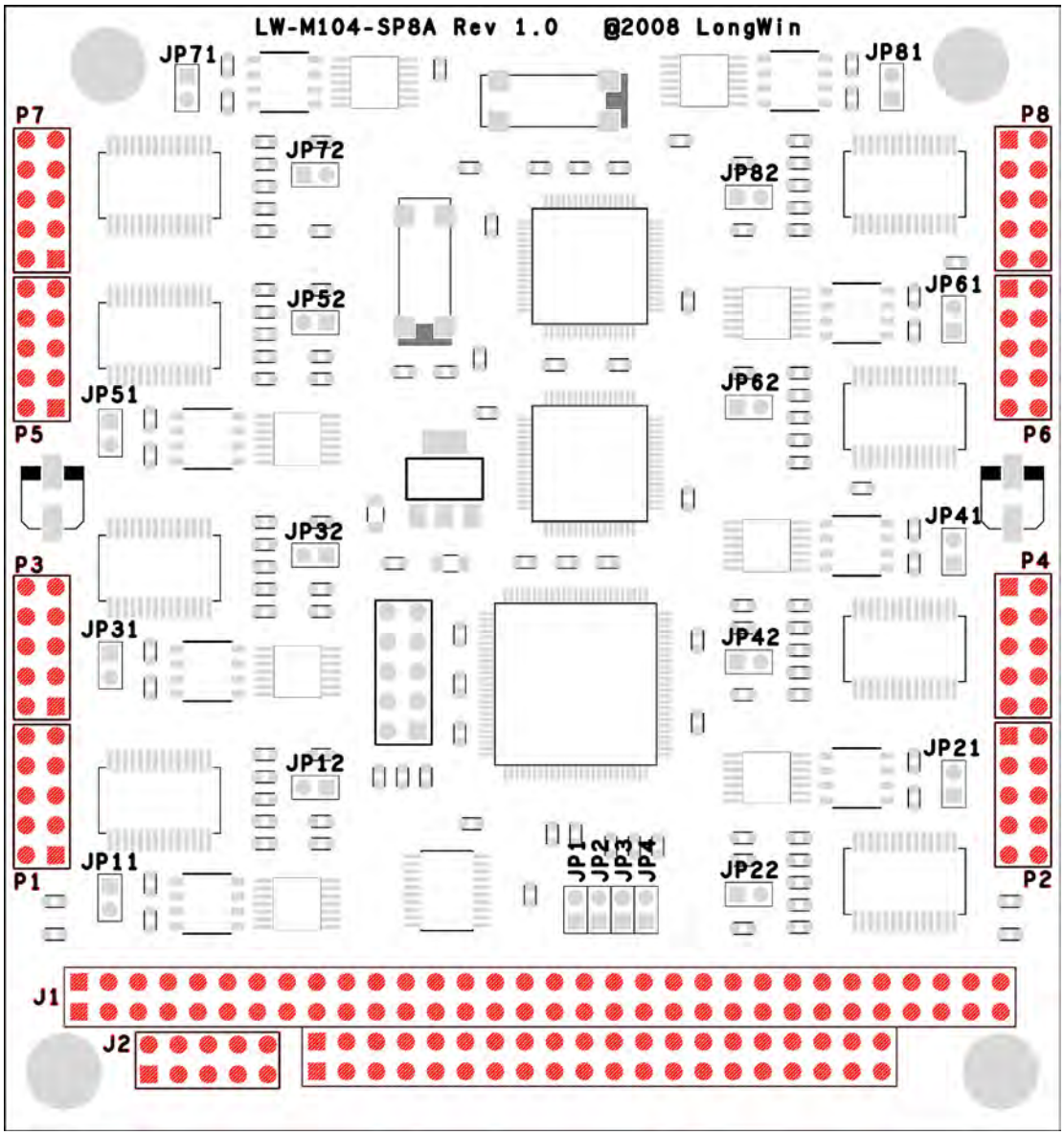

Figure 6. LW-M104-SP8A (DX) Connectors Location

You can design a PCB board assembly, made with female connectors in the same relative positions as the LW-M104-SP8A's headers, to eliminate cables requirements on customized module with straight headers. Precise dimensions for locating connectors are provided upon required.

Table 3. LW-M104-SP8A (DX) Connector List

Reference	Description	Pitch
J1	PC/104 Expansion Connector	0.1 Inch
J2	Power Connector	0.1 Inch
P1 – P8	Serial Interface Connector	0.1 Inch

The PC/104 expansion bus appears on connector J1. It uses a 0.1 Inch 4-row connector called out in the PC/104 specification. J1 has both male and female connections on top and bottom sides, allowing for "Stackthrough" assembly. You can expand the system with other PC/104 compliant expansion modules. These modules stack directly on the connectors.

2.3. Power Connector (J2 Option)

A standard 5x2-pin 0.1 inch pitch connector may be used to supply main power to the ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** and to devices that require. The +5V power supply is used to provide power for stacked external 5V PC/104 module if needed.

Be sure the power plug is wired correctly before applying power to the board! See *Table 4. Power Connector Pin Assignment*.

Table 4. Power Connector Pin Assignment

Signal	Pin		Pin	Signal
Ground	1		2	Key
Ground	3		4	+5V
Ground	5		6	+5V
Ground	7		8	+5V
Ground	9		10	+5V

The ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** requires only single +5VDC (±5%) for operation. The exact power requirement of the ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** depends on several factors, including the installed digital I/O devices, the peripheral connections.

2.4. Serial Interface Connectors

The 8 serial channels are accessed via four right-angle 20-pin IDC connectors of P1/3, P2/4, P5/7 and P6/8. Each connector provides 2 channels of serial ports. Pin assignments for serial port connectors are detailed as followings.

Table 5. Serial Port Connector Pin Assignment P1-8 (RS-485)

Description	Signal	Pin		Pin	Signal	Description
Carrier Detect	CD	1		2	DSR	Data Set Ready
Receive Data	RX (TX/RX+)	3		4	RTS (TX/RX+)	Request to Send
Transmit Data	TX	5		6	CTS	Clear to Send
Data Terminal Ready	DTR	7		8	RI	Ring Indicator
Ground	GND	9		10	Key	

The Pin 1 of connector is identified by a square pad.

2.5. PC/104 Expansion Bus (J1)

The ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** provides a PC/104 expansion bus for additional system functions. This bus offers compact, self-stacking, modular expandability. The PC/104 expansion bus signals are implementing an ISA bus, available on a 104-pin ISA bus connector.

The growing list of PC/104 modules available from LongWin and hundreds of other PC/104 vendors includes such functions as communications interfaces, video frame grabbers, field bus interfaces, digital signal processors (DSPs), data acquisition and control functions, and many specialized interfaces and controllers. In addition, custom application-specific logic boards can also be stacked directly on top of the LW-M104-SP8A (DX) using its PC/104 expansion bus interface as a rugged and reliable interconnect.

The PC/104 expansion bus appears on a 104-pin 0.1 Inch pitch female quad-row header (32 x 2 + 20 x 2). An implementation of the ISA bus appears on J1.

Table 6. PC/104 Expansion Bus Pin Assignment (J1)

Pin	Row A	Row B	Row C	Row D	Pin
1	IOCHK*	GND	GND	GND	0
2	SD7	RESET	SBHE*	MEMCS16*	1
3	SD6	+5V	LA23	IOCS16*	2
4	SD5	IRQ9	LA22	IRQ10	3
5	SD4	-5V	LS21	IRQ11	4
6	SD3	DRQ2	LA20	IRQ12	5
7	SD2	-12V	LA19	IRQ15	6
8	SD1	SRDY*	LA18	IRQ14	7
9	SD0	+12V	LA17	DACK0*	8
10	IOCHRDY	KEY	MEMR*	DRQ0	9
11	AEN	SMEMW*	MEMW*	DACK5*	10
12	SA19	SMEMR*	SD8	DRQ5	11
13	SA18	IOW*	SD9	DACK6*	12
14	SA17	IOR*	SD10	DRQ6	13
15	SA16	DACK3*	SD11	DACK7*	14
16	SA15	DRQ3	SD12	DRQ7	15
17	SA14	DACK1*	SD13	+5V	16
18	SA13	DRQ1	SD14	MASTER*	17
19	SA12	REFRESH*	SD15	GND	18
20	SA11	BCLK	KEY	GND	19
21	SA10	IRQ7			
22	SA9	IRQ6			
23	SA8	IRQ5			
24	SA7	IRQ4			
25	SA6	IRQ3			
26	SA5	DACK2*			
27	SA4	TC			
28	SA3	BALE			
29	SA2	+5V			
30	SA1	OSC			
31	SA0	GND			
32	GND	GND			

Chapter 3

Technical Reference

This chapter explains contains information that permits users to create an embedded system customized to specific requirements.

Information includes:

- *I/O Address*
- *UART Register Detail*
- *Baud Rate Selection*
- *Interrupt*
- *Asynchronous Communications*
- *RS-232 Protocol*
- *RS-422/485 Protocol*
- *PC/104 Stack Expansion*

Chapter 3. Technical Reference

3.1. I/O Address

The board occupies 64-Byte consecutive I/O locations... The module base address can be selected by jumper. If in doubt of where to assign the base address, refer to the *Table 1. I/O Base Address Jumper Setting Table* for your reference.

The 128 consecutive series I/O addresses space is shown as following *Table 7. Serial Channel I/O Address Map*.

Table 7. Serial Channel I/O Address Map

Address	Function
I/O Base Address	UART Channel 1 Base Address (Port P1)
I/O Base Address + 0x08	UART Channel 1 Base Address (Port P2)
I/O Base Address + 0x10	UART Channel 1 Base Address (Port P3)
I/O Base Address + 0x18	UART Channel 1 Base Address (Port P4)
I/O Base Address + 0x20	UART Channel 1 Base Address (Port P5)
I/O Base Address + 0x28	UART Channel 1 Base Address (Port P6)
I/O Base Address + 0x30	UART Channel 1 Base Address (Port P7)
I/O Base Address + 0x38	UART Channel 1 Base Address (Port P8)

The module is an I/O-mapped device that is easily configured from any language and can easily perform digital I/O access through the module's I/O ports. This is especially true if the form of the data is byte wide. All references to the I/O ports would be in absolute port addressing. However, a table could be used to convert the byte data ports to a logical reference.

The simplest I/O address mapping has the eight serial channels appearing as eight bytes each, next to each other, based at 128-byte boundary in the I/O map.

3.2. UART Register Detail

Each UART have 12 byte registers in a continuous address space for monitoring and control. These registers are shown in *Table 8. UART* Registers Address Mapping.

Table 8. UART Registers Address Mapping

Address *	Read	Write
Base	Receive Holding Register (RHR)	Transmit Holding Register (THR)
Base + 8x(n-1) + 1	Interrupt Enable Register (IER)	
Base + 8x(n-1) + 2	Interrupt Status Register (ISR)	FIFO Control Register (FCR)
Base + 8x(n-1) + 3	Line Control Register (LCR)	
Base + 8x(n-1) + 4	Modem Control Register (MCR)	
Base + 8x(n-1) + 5	Line Status Register (LSR)	Interrupt Number for COMn **
Base + 8x(n-1) + 6	Modem Status Register (MSR)	N/A
Base + 8x(n-1) + 7	Scratchpad Register (SPR)	
Base + 0x06		Base Address (Low Byte) ***
Base + 0x0E		Base Address (High Byte) ***

- n = Serial Port Channel Number of 1 to 8

** [D3:D0] = Interrupt Number of .3-7, 9-12, 14-15.

*** Must write the 2-Byte Base Address in Sequence, Low Byte first, then High Byte.

Table 9. Reset State for Register

Register	Reset State
IER	IER[7:0] = 0
ISR	ISR[7:1] = 0; ISR[0] = 1
LCR	LCR[7:0] = 0
MCR	MCR[7:0] = 0
LSR	LSR[7] = 0; LSR[6:5] = 1; LSR[4:0] = 0
MSR	MSR[7:4] = Input Signals; MSR[3:0] = 0
MCR	FCR[7:0] = 0

3.2.1. Transmit (THR) and Receive (RHR) Holding Registers

The serial transmitter section consists of an 8-bit Transmit Hold Register (THR) and Transmit Shift Register (TSR). The status of the THR is provided in the Line Status Register (LSR). Writing to the THR transfers the contents of the data bus (D7 to D0) to the THR, providing that the THR or TSR is empty. The THR empty flag in the LSR register will be set to logic 1 when the transmitter is empty or when data is transferred to the TSR. Note that a write operation can be performed when the THR empty flag is set (logic 0 = FIFO full; logic 1 = at least one FIFO location available).

The serial receive section also contains an 8-bit Receive Holding Register (RHR). Receive FIFO by reading the RHR register. The receive section provides a mechanism to prevent false starts. On the falling edge of a start or false start bit, an internal receiver counter starts counting clocks at the 16x clock rate. After 7½ clocks, the start bit time should be shifted to the center of the start bit. At this time the start bit is sampled, and if it is still logic 0 it is validated. Evaluating the start bit in this manner prevents the receiver from assembling a false character. Receiver status codes will be posted in the LSR.

3.2.2. Interrupt Enable Register (IER)

The Interrupt Enable Register (IER) masks the interrupts from receiver ready, transmitter empty, line status and modem status registers.

Table 10. Interrupt Enable Register (IER) bits Description

Bit	Symbol	Description
7:4	IER[7:4]	Reserved; set to '0'.
3	IER[3]	Modem Status Interrupt. Logic 0 = Disable the modem status register interrupt (Default) Logic 1 = Enable the modem status register interrupt
2	IER[2]	Receive Line Status Interrupt. Logic 0 = Disable the receiver line status interrupt (Default) Logic 1 = Enable the receiver line status interrupt
1	IER[1]	Transmit Holding Register Interrupt. This interrupt will be issued whenever the THR is empty, and is associated with LSR[1]. Logic 0 = Disable the transmitter empty interrupt (Default) Logic 1 = Enable the transmitter empty interrupt
0	IER[0]	Receive Holding Register Interrupt. This interrupt will be issued when the FIFO has reached the programmed trigger level, or is cleared when the FIFO drops below the trigger level in the FIFO mode of operation. Logic 0 = Disable the receiver ready interrupt (Default) Logic 1 = Enable the receiver ready interrupt

IER versus Receive FIFO interrupt mode operation

When the receive FIFO (FCR[0] = logic 1), and receive interrupts (IER[0] = logic 1) are enabled, the receive interrupts and register status will reflect the following:

- The receive data available interrupts are issued to the external CPU when the FIFO has reached the programmed trigger level. It will be cleared when the FIFO drops below the programmed trigger level.
- FIFO status will also be reflected in the user accessible ISR register when the FIFO trigger level is reached. Both the ISR register status bit and the interrupt will be cleared when the FIFO drops below the trigger level.
- The data ready bit (LSR[0]) is set as soon as a character is transferred from the shift register to the receive FIFO. It is reset when the FIFO is empty.

IER versus Receive/Transmit FIFO polled mode operation

When FCR[0] = logic 1, resetting IER[0:3] enables the UART in the FIFO polled mode of operation. Since the receiver and transmitter have separate bits in the LSR, either or both can be used in the polled mode by selecting respective transmit or receive control bit(s).

- LSR[0] will be a logic 1 as long as there is one byte in the receive FIFO.
- LSR[1:4] will provide the type of errors encountered, if any.
- LSR[5] will indicate when the transmit FIFO is empty.
- LSR[6] will indicate when both the transmit FIFO and Transmit Shift Register are empty.
- LSR[7] will indicate any FIFO data errors.

3.2.3. FIFO Control Register (FCR)

This register is used to enable the FIFOs, clear the FIFOs, set the receive FIFO trigger levels, and select the DMA mode.

DMA Mode 0 (FCR bit 3 = 0)

Set and enable the interrupt for each single transmit or receive operation, and is similar to the 16C454 mode. Transmit Ready (TXRDY) will go to a logic 0 whenever an empty transmit space is available in the Transmit Holding Register (THR). Receive Ready (RXRDY) will go to a logic 0 whenever the Receive Holding Register (RHR) is loaded with a character.

DMA Mode 1 (FCR bit 3 = 1)

Set and enable the interrupt in a block mode operation. The transmit interrupt is set when there are one or more FIFO locations empty. The receive interrupt is set when the receive FIFO fills to the programmed trigger level. However, the FIFO continues to fill regardless of the programmed level until the FIFO is full. RXRDY remains logic 0 as long as the FIFO fill level is above the programmed trigger level.

Table 11. FIFO Control Register (FCR) Bits Description

Bit	Symbol	Description
7:6	FCR[7:6]	RCVR Trigger. These bits are used to set the trigger level for the receive FIFO interrupt. An interrupt is generated when the number of characters in the FIFO equals the programmed trigger level. However, the FIFO will continue to be loaded until it is full.
5:4	FCR[5:4]	Not Used; Initialized to logic 0
3	FCR[3]	DMA Mode Select. Logic 0 = Set DMA mode '0' (Default) Logic 1 = Set DMA mode '1' **Transmit Operation in Mode '0'**: When the UART is in the 16C450 mode (FIFOs disabled; FCR[0] = logic 0) or in the FIFO mode (FIFOs enabled; FCR[0] = logic 1; FCR[3] = logic 0), and when there are no characters in the transmit FIFO or Transmit Holding Register, the TXRDY pin will be logic 0. Once active, the TXRDY pin will go to logic 1 after the first character is loaded into the Transmit Holding Register. **Receive Operation in Mode '0'**: When the UART is in mode '0' (FCR[0] = logic 0), or in the FIFO mode (FCR[0] = logic 1; FCR[3] =

		Logic 0) and there is at least one character in the receive FIFO, the RXRDY pin will be a logic 0. Once active, the RXRDY pin will go to logic 1 when there are no more characters in the receiver. **Transmit Operation in Mode '1'**: When the UART is in FIFO mode (FCR[0] = logic 1; FCR[3] = logic 1), the TXRDY pin will be a logic 1 when the transmit FIFO is completely full. It will be logic 0 if one or more FIFO locations are empty. **Receive Operation in Mode '1'**: When the UART is in FIFO mode (FCR[0] = logic 1; FCR[3] = logic 1) and the trigger level has been reached, or a Receive Time-out has occurred, the RXRDY pin will go to a logic 0. Once activated, it will go to a logic 1 after there are no more characters in the FIFO.
2	FCR[2]	XMIT FIFO Reset. Logic 0 = No FIFO transmit reset (Default) Logic 1 = Clears the contents of the transmit FIFO and resets the FIFO counter logic (the Transmit Shift Register is not cleared or altered). This bit will return to logic 0 after clearing the FIFO.
1	FCR[1]	RCVR FIFO Reset. Logic 0 = No FIFO receive reset (Default) Logic 1 = Clears the contents of the receive FIFO and resets the FIFO counter logic (the Receive Shift Register is not cleared or altered). This bit will return to logic 0 after clearing the FIFO.
0	FCR[0]	FIFO Enable. Logic 0 = Disable the transmit and receive FIFO (Default) Logic 1 = Enable the transmit and receive FIFO. This bit must be a 1 when other FCR bits are written to, or they will not be programmed.

Table 12. RCVR Trigger Levels

FCR[7]	FCR[6]	RX FIFO Trigger Level
0	0	1
0	1	4
0	0	8
0	1	14

3.2.4. Interrupt Status Register (ISR)

The UART provides four levels of prioritized interrupts to minimize external software interaction. The Interrupt Status Register (ISR) provides the user with four interrupt status bits. Performing a read cycle on the ISR will provide the user with the highest pending interrupt level to be serviced. No other interrupts are acknowledged until the pending interrupt is serviced. Whenever the Interrupt Status Register is read, the interrupt status is cleared. However, it should be noted that only the current pending interrupt is cleared by the read. A lower level interrupt may be seen after re-reading the interrupt status bits.

Table 13. Interrupt Source

Priority Level	ISR Bit						Description
	5	4	3	2	1	0	
1	0	0	0	1	1	0	LSR (Receiver Line Status Register)

2	0	0	0	1	0	0	RXRDY (Receive Data Ready)
3	0	0	1	1	0	0	RXRDY (Receive Data time-out)
4	0	0	0	0	1	0	TXRDY (Transmitter Holding Register Empty)
5	0	0	0	0	0	0	MSR (Modem Status Register)

Table 14. Interrupt Status Register (ISR) Bits Description

Bit	Symbol	Description
7:6	ISR[7:6]	FIFOs Enabled. These bits are set to logic 0 when the FIFO is not being used. They are set to logic 1 when the FIFOs are enabled. Logic 0 or Cleared = Default
5:4	ISR[5:4]	Reserved; Set to 0.
3:1	ISR[3:1]	INT priority bits 2 to 0. These bits indicate the source for a pending interrupt at interrupt priority levels 1, 2, and 3. Logic 0 or Cleared = Default
0	ISR[0]	INT Status. Logic 0 = An interrupt is pending and the ISR contents may be used as a pointer to the appropriate interrupt service routine Logic 1 = No interrupt pending (Default)

3.2.5. Line Control Register (LCR)

The Line Control Register is used to specify the asynchronous data communication format. The word length, the number of stop bits, and the parity are selected by writing the appropriate bits in this register.

Table 15. Line Control Register (LCR) Bits Description

Bit	Symbol	Description
7	LCR[7]	Divisor Latch Enable. The internal baud rate counter latch and Enhance Feature mode enable. Logic 0 = Divisor latch disabled (Default) Logic 1 = Divisor latch enabled
6	LCR[6]	Set Break. When enabled, the Break control bit causes a break condition to be transmitted (the TX output is forced to a logic 0 state). This condition exists until disabled by setting LCR[6] to logic 0. Logic 0 = No TX break condition (Default) Logic 1 = Forces the transmitter output (TX) to a logic 0 for alerting the remote receiver to a line break condition
5	LCR[5]	Set Parity. If the parity bit is enabled, LCR[5] selects the forced parity format. Programs the parity conditions. Logic 0 = Parity is not forced (Default) LCR[5] = Logic 1 and LCR[4] = Logic 0: Parity bit is forced to a logical 1 for the transmit and receive data LCR[5] = Logic 1 and LCR[4] = Logic 1: Parity bit is forced to a logical 0 for the transmit and receive data
4	LCR[4]	Even Parity. If the parity bit is enabled with LCR[3] set to a logic 1, LCR[4] selects the even or odd parity format. Logic 0 = Odd parity is generated by forcing an odd number of logic

		1s in the transmitted data. The receiver must be programmed to Check the same format (normal default condition). Logic 1 = Even parity is generated by forcing an even number of Logic 1s in the transmitted data. The receiver must be programmed to check the same format.
3	LCR[3]	Parity Enable. Parity or no parity can be selected via this bit. Logic 0 = No parity (Default) Logic 1 = A parity bit is generated during the transmission, receiver checks the data and parity for transmission errors
2	LCR[2]	Stop Bits. The length of stop bit is specified by this bit in conjunction with the programmed word length. Logic 0 or Cleared = Default
1:0	LCR[1:0]	Word Length bits 1, 0. These two bits specify the word length to be transmitted or received. Logic 0 or Cleared = Default

Table 16. LCR[5] Parity Selection

LCR[5]	LCR[4]	LCR[3]	Parity Selection
X	X	0	No Parity
0	0	1	Odd Parity
0	1	1	Even Parity
1	0	1	Force Parity "1"
1	1	1	Force Parity "0"

Table 17. LCR[2] Stop Bit length

LCR[2]	Word Length (bits)	Stop Bit Length (bit times)
0	5, 6, 7, 8	1
1	5	1½
1	6, 7, 8	2

Table 18. LCR[1:0] Word Length

LCR[1]	LCR[0]	Word Length (bits)
0	0	5
0	1	6
1	0	7
1	1	8

3.2.6. Modem Control Register (MCR)

This register controls the interface with the modem or a peripheral device.

Table 19. Modem Control Register (MCR) Bits Description

Bit	Symbol	Description

7:6	MCR[7:6]	Reserved; Set to '0'.
5	MCR[5]	Autoflow Control Enable.
4	MCR[4]	Loop-Back. Enable the local loop-back mode (diagnostics). In this mode the transmitter output (TX) and the receiver input (RX), CTS, DSR, CD, and RI are disconnected from the UART I/O pins. Internally the modem data and control pins are connected into a loop-back data configuration. In this mode, the receiver and transmitter interrupts remain fully operational. The Modem Control Interrupts are also operational, but the interrupts' sources are switched to the lower four bits of the Modem Control. Interrupts continue to be controlled by the IER register. Logic 0 = Disable loop-back mode (Default) Logic 1 = Enable local loop-back mode (diagnostics)
3	MCR[3]	OP2, INTx enable. Used to control the modem CD signal in the loop-back mode. Logic 0 = Forces INTA to INTD outputs to the 3-state mode during the 16 mode (normal default condition). In the loop-back mode, sets OP2 (CD) internally to logic 1. Logic 1 = Forces the INTA to INTD outputs to the active mode during the 16 mode. In the loop-back mode, sets OP2 (CD) internally to a logic 0.
2	MCR[2]	OP1. This bit is used in the Loop-back mode only. In the loop-back mode, this bit is used to write the state of the modem RI interface signal via OP1.
1	MCR[1]	RTS Logic 0 = Force RTS output to a logic 1 (Default) Logic 1 = Force RTS output to a logic 0 Automatic RTS may be used for hardware flow control by enabling MCR[5].
0	MCR[0]	DTR Logic 0 = Force DTR output to a logic 1 Default) Logic 1 = Force DTR output to a logic 0

3.2.7. Line Status Register (LSR)

This register provides the status of data transfers between the UART and the CPU.

Table 20. Line Status Register (LSR) Bits Description

Bit	Symbol	Description
7	LSR[7]	FIFO Data Error. Logic 0 = No error (Default) Logic 1 = At least one parity error, framing error or break indication is in the current FIFO data. This bit is cleared when LSR register is read.
6	LSR[6]	THR and TSR Empty. This bit is the Transmit Empty indicator. This bit is set to a logic 1 whenever the Transmit Holding Register and the Transmit Shift Register are both empty. It is reset to logic 0 whenever either the THR or TSR contains a data character. In the FIFO mode, this bit is set to '1' whenever the transmit FIFO and Transmit Shift Register are both empty.
5	LSR[5]	THR Empty. This bit is the Transmit Holding Register Empty indicator. This bit indicates that the UART is ready to accept a new character for transmission. In addition, this bit causes the UART to issue an interrupt to CPU when the THR interrupt enable is set. The THR bit is set to logic

		1 when a character is transferred from the Transmit Holding Register into the Transmitter Shift Register. The bit is reset to logic 0 concurrently with the loading of the transmitter holding register by the CPU. In the FIFO mode, this bit is set when the transmit FIFO is empty; it is cleared when at least 1 byte is written to the transmit FIFO.
4	LSR[4]	Break Interrupt. Logic 0 = No break condition (Default) Logic 1 = The receiver received a break signal (RX was a logic 0 for one character frame time). In the FIFO mode, only one break character is loaded into the FIFO.
3	LSR[3]	Framing Error. Logic 0 = No framing error (Default) Logic 1 = Framing error. The receive character did not have a valid stop bit(s). In the FIFO mode, this error is associated with the character at the top of the FIFO.
2	LSR[2]	Parity Error. Logic 0 = No parity error (Default) Logic 1 = Parity error. The receive character does not have correct Parity information and is suspect. In the FIFO mode, this error is associated with the character at the top of the FIFO.
1	LSR[1]	Overrun Error. Logic 0 = No overrun error (Default) Logic 1 = Overrun error. A data overrun error occurred in the Receive Shift Register. This happens when additional data arrives while the FIFO is full. In this case, the previous data in the shift register is overwritten. Note that under this condition, the data byte in the Receive Shift Register is not transferred into the FIFO, therefore the data in the FIFO is not corrupted by the error.
0	LSR[0]	Receive Data Ready. Logic 0 = No data in Receive Holding Register or FIFO (Default) Logic 1 = Data has been received and is saved in the Receive Holding Register or FIFO

3.2.8. Modem Status Register (MSR)

This register provides the current state of the control interface signals from the modem, or other peripheral device to which the UART is connected. Four bits of this register are used to indicate the changed information. These bits are set to logic 1 whenever a control input from the modem changes state. These bits are set to logic 0 whenever the CPU reads this register.

Table 21. Modem Status Register (MSR) Bits Description

Bit	Symbol	Description
7	MSR[7]	CD (active HIGH, logical 1). Normally this bit is the complement of the CD input. In the Loop-back mode this bit is equivalent to the OP2 bit in the MCR register.
6	MSR[6]	RI (active HIGH, logical 1). Normally this bit is the complement of the RI input. In the Loop-back mode this bit is equivalent to the OP1 bit in the MCR register.
5	MSR[5]	DSR (active HIGH, logical 1). Normally this bit is the complement of the DSR input. In Loop-back mode this bit is equivalent to the DTR bit in the MCR register.
4	MSR[4]	CTS (active HIGH, logical 1). CTS functions as hardware flow control

		signal input if it is enabled via MCR[5]. Flow control (when enabled) allows starting and stopping the transmissions based on the external modem CTS signal. A logic 1 at the CTS pin will stop UART transmissions as soon as current character has finished transmission. Normally MSR[4] is the complement of the CTS input. However, in the loop-back mode, this bit is equivalent to the RTS bit in the MCR register.
3	MSR[3]	CD Logic 0 = No CD change (Default). Logic 1 = The CD input to the UART has changed state since the last time it was read. A modem Status Interrupt will be generated.
2	MSR[2]	RI Logic 0 = No RI change (Default). Logic 1 = The RI input to the UART has changed from a logic 0 to logic 1. A modem Status Interrupt will be generated.
1	MSR[1]	DSR Logic 0 = No DSR change (Default). Logic 1 = The DSR input to the UART has changed state since the last time it was read. A modem Status Interrupt will be generated.
0	MSR[0]	CTS Logic 0 = No CTS change (Default). Logic 1 = The CTS input to the UART has changed state since the last time it was read. A modem Status Interrupt will be generated.

3.2.9. Scratchpad Register (SPR)

The UART provides a temporary data register to store 8 bits of user information.

3.3. Baud Rate Selection

Initializing the chip requires knowledge of the UART's register set. The first step is to set the baud rate divisor. You do this by first setting the Divisor Latch Enable bit LCR[7] high. This bit is Bit 7 at Line Control Register (LCR).

You then load the divisor into MSB of Divisor Latch (DLM) and LSB of Divisor Latch (DLL) registers. The following equation defines the relationship between baud rate and divisor:

Desired Baud Rate = 14.7456Mhz / (16 * Divisor)

Table 22. Baud Rate Divisor Table

Baud Rate (bps)	Divisor (Hexadecimal)	Max Cable Length (ft)
921,600	1 (0x0001)	250
460,800	2 (0x0002)	550
230,400	4 (0x0004)	1400
115,200	8 (0x0008)	3000
57,600	16 (0x0010)	4000
38,400	24 (0x0018)	4000
28,800	32 (0x0020)	4000
19,200	48 (0x0030)	4000
14,400	64 (0x0040)	4000
9,600	96 (0x0060)	4000 (Most Common)
4,800	192 (0x00C0)	4000

2,400	384 (0x0180)	4000
1,200	768 (0x0300)	4000

3.4. Interrupt

The ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module*** provides 11 interrupt sources of IRQ3 to IRQ7, IRQ9 to IRQ12, IRQ14 and IRQ15. The interrupt of each serial channel can be independently directed to any one of the 11 interrupt sources by software.

Table 23. Interrupt Register

Interrupt Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Base + 8x(n-1) + 5 *	N/A				Interrupt Channel Number for the UART			
	Interrupt Channel Number: 0x0 : Disable Interrupt for the Input Port 0x1 : Disable Interrupt for the Input Port 0x2 : Disable Interrupt for the Input Port 0x3 : Set IRQ3 to the Input Port 0x4 : Set IRQ4 to the Input Port 0x5 : Set IRQ5 to the Input Port 0x6 : Set IRQ6 to the Input Port 0x7 : Set IRQ7 to the Input Port 0x8 : Disable Interrupt for the Input Port 0x9 : Set IRQ9 to the Input Port 0xA : Set IRQ10 to the Input Port 0xB : Set IRQ11 to the Input Port 0xC : Set IRQ12 to the Input Port 0xD : Disable Interrupt for the Input Port 0xE : Set IRQ14 to the Input Port 0xF : Set IRQ15 to the Input Port							

* n = Serial Port Channel Number of 1 to 8

On occasion, a system application will require more interrupt levels than are available interrupt sources. You can simply share the same interrupt source by setting to same interrupt channel to relative serial channel. You can also disable the interrupt for the input connectors by simply writing a "0" to the relative interrupt register.

Interrupt should be set prior to use, if an interrupt is required by your application software. Consult the user manual for the application software being used to determine the proper setting.

The following example in C Language is provided as a guide to assist you in developing your working software. In this example, the module base address is 0x300 (Hexadecimal). Set interrupt source of IRQ6 to serial channel 5 (P5).

Interrupt Register Address = (I/O Base Address + 8 x (5-1) + 5 = 0x300 + 0x20 + 0x05 = 0x325

```
main()
{
    ...
    outportb(0x325, 0xA6)   ; Set IRQ6 to serial channel 5 (P5)
    ...
```

```
}
```

3.5. Asynchronous Communications

Serial data communications implies that individual bits of a character are transmitted consecutively to a receiver that assembles the bits back into a character. Data rate, error checking, handshaking, and character framing (start/stop bits) are pre-defined and must correspond at both the transmitting and receiving ends.

Asynchronous communications is the standard means of serial data communication for PC compatibles and PS/2 computers. The original PC was equipped with a communication or COM: port that was designed around an 8250 Universal Asynchronous Receiver Transmitter (UART). This device allows asynchronous serial data to be transferred through a simple and straightforward programming interface. A starting bit followed by a pre-defined number of data bits (5, 6, 7, or 8) defines character boundaries for asynchronous communications. The end of the character is defined by the transmission of a pre-defined number of stop bits (usually 1, 1.5 or 2). An extra bit used for error detection is often appended before the stop bits.

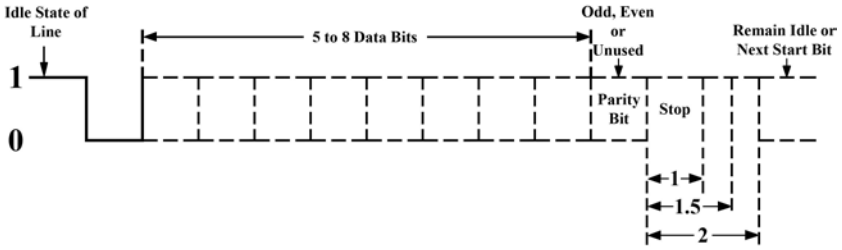

Figure 7. Asynchronous Communications Bit Diagram

This special bit is called the parity bit. Parity is a simple method of determining if a data bit has been lost or corrupted during transmission. There are several methods for implementing a parity check to guard against data corruption. Common methods are called (E)ven Parity or (O)dd Parity. Sometimes parity is not used to detect errors on the data stream. This is referred to as (N)o parity. Because each bit in asynchronous communications is sent consecutively, it is easy to generalize asynchronous communications by stating that each character is wrapped (framed) by pre-defined bits to mark the beginning and end of the serial transmission of the character. The data rate and communication parameters for asynchronous communications have to be the same at both the transmitting and receiving ends. The communication parameters are baud rate, parity, number of data bits per character, and stop bits (i.e. 9600, N, 8, 1).

3.6. RS-232 Protocol

Quite possibly the most widely used communication standard is RS-232. This implementation has been defined and revised several times and is often referred to as RS-232-C/D/E or EIA/TIA-232-C/D/E. It is defined as "*Interface between Data Terminal Equipment and Data Circuit- Terminating Equipment Employing Serial Binary Data Interchange*".

The mechanical implementation of RS-232 is on a 25-pin D sub connector. The IBM PC computer defined the RS-232 port on a 9 pin D sub connector and subsequently the EIA/TIA approved this implementation as the EIA/TIA-574 standard. This standard has defined as the "*9-Position Non-*

Synchronous Interface between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange". Both implementations are in wide spread use and will be referred to as RS-232 in this document.

RS-232 is capable of operating at data rates up to 20K bps / 50 ft. The absolute maximum data rate may vary due to line conditions and cable lengths. RS-232 often operates at 38.4K bps over very short distances. The voltage levels defined by RS-232 range from -12 to +12 volts. RS-232 is a single ended or unbalanced interface, meaning that a single electrical signal is compared to a common signal (ground) to determine binary logic states. A voltage of +12 volts (usually +3 to +10 volts) represents a binary 0 (space) and -12 volts (-3 to -10 volts) denote a binary 1 (mark).

3.7. RS-422/485 Protocol

RS-422 is a differential interface that defines voltage levels and driver/receiver electrical specifications. On a differential interface, logic levels are defined by the difference in voltage between a pair of outputs or inputs. In contrast, a single ended interface, for example RS-232, defines the logic levels as the difference in voltage between a single signal and a common ground connection. Differential interfaces are typically more immune to noise or voltage spikes that may occur on the communication lines. Differential interfaces also have greater drive capabilities that allow for longer cable lengths. RS-422 is rated up to 10 Megabits per second and can have cabling 4000 feet long.

RS-422 also defines driver and receiver electrical characteristics that will allow 1 driver and up to 32 receivers on the line at once. RS-422 signal levels range from 0 to +5 volts. RS-422 does not define a physical connector.

RS-485 is backwardly compatible with RS-422; however, it is optimized for party line or multi-drop applications. The output of the RS-422/485 driver is capable of being **Active** (enabled) or **Tri-State** (disabled). This capability allows multiple ports to be connected in a multi-drop bus and selectively polled. RS-485 allows cable lengths up to 4000 feet and data rates up to 10 Megabits per second.

The signal levels for RS-485 are the same as those defined by RS-422. RS-485 has electrical characteristics that allow for 32 drivers and 32 receivers to be connected to one line. This interface is ideal for multi-drop or network environments. RS-485 tri-state driver (not dual-state) will allow the electrical presence of the driver to be removed from the line. Only one driver may be active at a time and the other driver(s) must be tri-stated. RS-485 can be cabled in two ways, two wire and four wire mode.

Two-wire mode does not allow for full duplex communication, and requires that data be transferred in only one direction at a time. For half-duplex operation, the two transmit pins should be connected to the two receive pins (TX+ to RX+ and TX- to RX-). Four wire mode allows full duplex data transfers. RS-485 does not define a connector pin-out or a set of modem control signals. RS-485 does not define a physical connector.

Working with RS-422/485 devices is not much different from working with standard RS-232 serial devices and this standard overcomes deficiencies in the RS-232 standard. First, the cable length between two RS-232 devices must be short; less than 50 feet. Second, many RS-232 errors are the result of noise induced on the cables. The RS-485 standard permits cable lengths up to 4000 feet and, because it operates in differential mode, it is more immune to induced noise.

A third deficiency of RS-232 is that more than two devices cannot share the same cable. This is also true for RS422 but RS-485 offers all the benefits of RS422 plus allows up to 32 devices to share

the same twisted pairs. An exception to the foregoing is that multiple RS422 devices can share a single cable if only one will talk and the others will always receive.

The direction of communication is controlled through the RTS signal. The RTS signal is controlled by programming the UART. The UART is controlled in the same manner as all series 16C550 compatible UARTs. Details on its operation are included in the data sheet for the UART. If it is desired to have the channel in the receive mode, it must be put into this mode by changing the mode, through the RTS signal, upon completion of each transmission. Similarly, the UART must be put into the transmit mode prior to transmitting.

3.7.1. Balanced Differential Signals

The reason that RS422 and RS-485 devices can drive longer lines with more noise immunity than RS-232 devices is that a balanced differential drive method is used. In a balanced differential system, the voltage produced by the driver appears across a pair of wires. A balanced line driver can also have an input "enable" signal that connects the driver to its output terminals. If the "enable" signal is OFF, the driver is disconnected from the transmission line. This disconnected or disabled condition is usually referred to as the "tristate" condition and represents high impedance. RS-485 drivers must have this control capability. RS422 drivers may have this control but it is not always required.

A balanced differential line receiver senses the voltage state of the transmission line across the two signal input lines. If the differential input voltage is greater than +200 mV, the receiver will provide a specific logic state on its output. If the differential voltage input is less than -200 mV, the receiver will provide the opposite logic state on its output.

A maximum common mode voltage rating of ±7V provides good noise immunity from voltages induced on the twisted pair lines. The signal ground line connection is necessary in order to keep the common mode voltage within that range. The circuit may operate without the ground connection but may not be reliable.

3.7.2. Two-Wire Multi-drop Network

The following illustration shows a typical multidrop or party line network. Note that the transmission line is terminated on both ends of the line but not at drop points in the middle of the line.

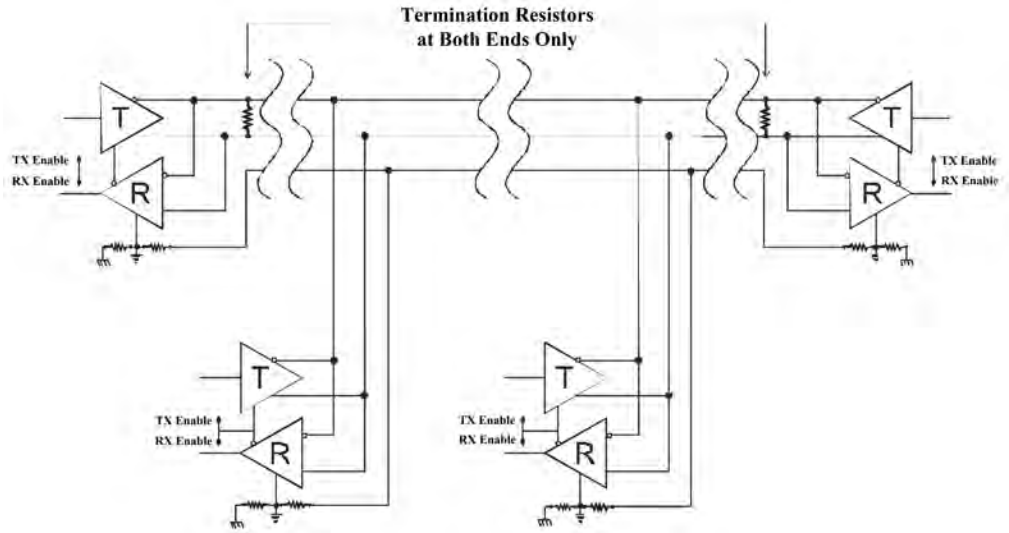

Figure 8. Typical RS-485 Two-Wire Multi-drop Network

3.8. PC/104 Stack Expansion

PC/104 compatible expansion modules can be installed on the LW-M104-SP8A (DX) expansion bus. The buffered output signals to the expansion bus are standard TTL level signals. All inputs to the LW-M104-SP8A (DX) operate at TTL levels and present a typical CMOS load to the expansion bus.

You can install one or more PC/104 compatible modules on the LW-M104-SP8A (DX) expansion connectors. You can stack several modules on the ***LW-M104-SP8A (DX) 8-Port RS-232/422/485 Serial Module***. Each additional module increases the thickness of the package by 0.66 inches (15 mm). See Following Figure for details.

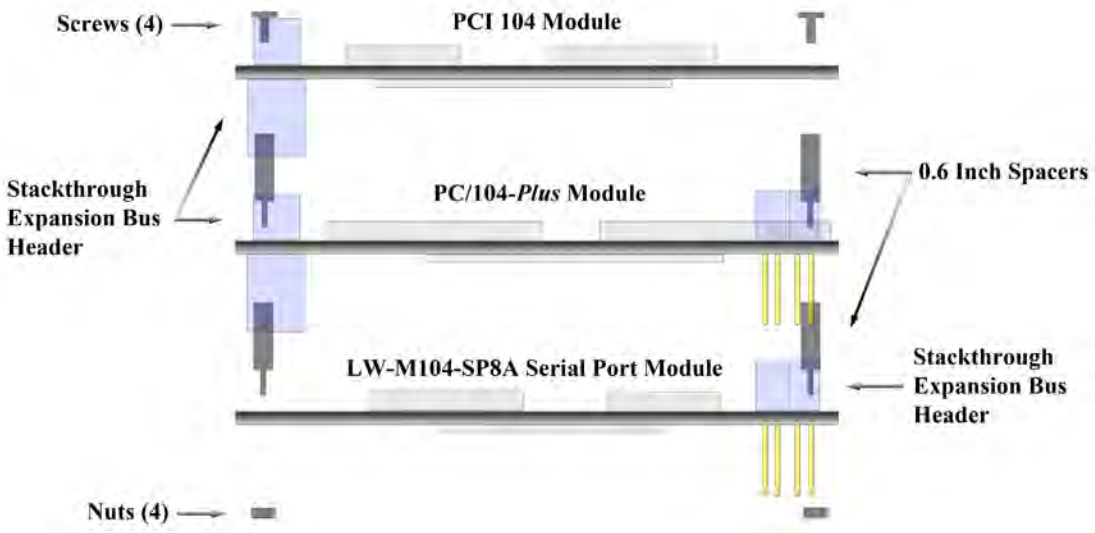

Figure 9. PC/104 Module Stack

Index

S

T

U

V

W

X