TAN - 014

A Comparison between Exar's XR-88C681 with Signetics' SC26C92 DUART Devices

Comparison of EXAR DUART (XR-88C681 device) with that of Signetics (SC26C92)

Introduction

Although the XR-88C681 and the Signetics SC26C92 devices have exactly the same pin outs, they are not drop-in compatible devices. There are three minor hardware difference, and numerous software/firmware related differences between these two devices.

Hardware Difference:

1. The XR-88C681 device uses 3 byte FIFOs in the Transmitter and Receiver of both channels; whereas the SC26C92 device uses 8 byte FIFOs. Of course this Hardware difference also results in differences in the DUART Interrupt Structure for the Transmit and Receiver FIFO, as will be presented below.

2. The SC26C92 has an additional Mode Register, MR0n, for each channel.

3. The SC26C92 does not come in a 28 pin DIP

Firmware/Software Related Differences:

The Register Addressing is slightly different between these two devices. As well as the meaning of commands written to the Command Registers. The specifics of these differences are enumerated in this write up in Tables 1 through 3. The are numerous other differences in the feature being offered by each of the devices. These differences are also listed below..

1. I/Z Modes - XR-88C681

The XR-88C681 (in 40 pin DIP or 44 pin PLCC package) may be programmed to operate in two modes to accommodate different CPU interface requirements. This feature is not available in the 28 pin DIP packaged devices. In the I-mode (or Intel Mode), which is the default mode after a hardware reset, interrupt prioritization and interrupt vector generation, if required are implemented using external hardware. In this mode, the on-chip interrupt vector register (IVR) is not used, and is available for use as an auxiliary read/write register for any purpose. The Signetics SC26C92 device only operates in the I mode. In the Z (or Zilog) mode, which is invoked via a command to Command Register B, pins 37, 38, and 39 are designated interrupt acknowledge input (IACKN), interrupt enable output (IEO) and interrupt enable input (IEI), respectively. IEI and IEO are the input and output of an interrupt daisy chain, as illustrated in Figure 1 on the XR-88C681 data sheet. A logic high at the IEI input allow the DUART to generate an interrupt request. A device with its IEI input "high" which is requesting an interrupt sets its IEO output low to inhibit lower priority devices from generating their own interrupt requests.

A device with its IEI input held low is inhibited from generating an interrupt and must, in turn keep its IEO output low.

2. Mode Register "0" - SC26C92

The SC26C92 device offers an additional mode register for both of the channels. This is over and above the MR1n and MR2n registers which are available for both channels in the XR-88C681 device. The bit-format and description of the MR0n Register is presented below.

MR0n Register (SC26C92 Only)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Rx WATCH DOG	**Rx INT BIT 2**	**TxINT (1:0)**		**Unused**	**BAUD RATE EXTENDED II**	**TEST 2**	**BAUD RATE EXTENDED I**
0 = Disable 1 = Enable	See Below	See Below		Set to 0	0 = Normal 1 = Extend II	Set to 0	0 = Normal 1 = Extend I

MR0n DESCRIPTION

MR0n[7] - Rx WATCH DOG

This bit controls the receiver watch dog timer. (0 = Disable, 1 = Enable). When enabled, the watch dog timer will generate a receiver interrupt if the receiver FIFO has not been accessed within 64 bit time of the receiver 1X clock. This is used to alert the control processor that data is in the RxFIFO that has not been read. This situation may occur when the byte count of the last part of a message is not large enough to generate an interrupt.

MR0n[6] - RxINT BIT 2

Bit 2 of Receiver FIFO Interrupt Level. This bit along with MR1n[6] sets the fill level of the 8 byte FIFO that generates the receiver interrupt, as shown in the table below:

MR0n[6]	MR1n[6]	Interrupt Condition
0	0	1 or more bytes in FIFO (RxRDY)
0	1	3 or more bytes in FIFO
1	0	6 or more bytes in FIFO
1	1	8 bytes in FIFO (RxFULL)

MR0n[5:4] - TxINT
These two bits set the fill level of the 8 byte Transmit FIFO that generates the Transmitter interrupts, as shown in the table below:

MR0n[5]	MR0n[4]	Interrupt Condition
0	0	8 bytes empty (Tx EMPTY)
0	1	4 or more bytes empty
1	0	6 or more bytes empty
1	1	1 or more bytes empty

MR0n[3] DON'T CARE:
Not used. Should be set to 0.

MR0n[2:0] BAUD EXTENDED MODES
These bits are used to select one of the six baud rates (see Table E, below)

Table E, Baud Rate (for SC26C92)

CSRA[7:4]	MR0n[0] = 0 (Normal Mode)		MR0n[0] = 1 (Extended Mode I)		MR0n[2] = 1 (Extended Mode II)	
	ACR[7] = 0	ACR[7] = 1	ACR[7] = 0	ACR[7] = 1	ACR[7] = 0	ACR[7] = 1
0000	50	75	50	450	4800	7200
0001	110	110	110	110	880	880
0010	134.5	134.5	134.5	134.5	1076	1076
0011	200	150	200	900	19.2K	14.4K
0100	300	300	1800	1800	28.8K	38.4K
0101	600	600	3600	3600	57.6K	57.6K
0110	1200	1200	7200	7200	115.2K	115.2K
0111	1050	2000	1050	2000	1050	2000
1000	2400	2400	14.4K	14.4K	57.6K	57.6K
1001	4800	4800	28.8K	28.8K	4800	4800
1010	7200	1800	7200	1800	57.6K	57.6K
1011	9600	9600	57.6K	57.6K	9600	9600
1100	38.4K	19.2K	230.4K	115.2K	38.4K	19.2K
1101	Timer	Timer	Timer	Timer	Timer	Timer
1110	IP4 - 16X	IP4 - 16X	IP4 - 16X	IP4 - 16X	IP4 - 16X	IP4 - 16X
1111	IP4 - 1X	IP4 - 1X	IP4 - 1X	IP4 - 1X	IP4 - 1X	IP4 - 1X

A Comparison between Exar's XR-88C681 with Signetics' SC26C92 DUART Devices

In Summary, the Resulting Differences (or departures from the XR-88C681), due to MR0n Register are as follows:

a. Program in the Interrupt Level for both the Transmit and Receive FIFOs.
b. Gives the user a wider range of Baud Rates (see Table E).
c. Program the DUART to generate an Interrupt if the Receiver FIFO has not been read by the CPU in the last 64 bit times of the Receiver 1X clock (Watch Dog Timer).

3. Receiver Time-out Mode - SC26C92

In addition to the "Watch Dog Timer" described in the receiver section, the Counter/Timer may be used for a similar function. When the Receiver Time-out Mode is enabled, the received data stream will control the counter. Each time a received character is transferred from the shift register to the RxFIFO, the counter is restarted. If a new character is not received before the counter reaches the "zero" count, the counter ready bit is set and an interrupt can be generated. This mode can be used to indicate when data has been left in the RxFIFO for more than the programmed time limit. Otherwise, if the receiver has been programmed to interrupt the CPU when the receive FIFO is full, and the message end before the FIFO is full, the CPU may not know there is data left in the FIFO. The CTU and CTL value would be programmed for just one character time, so that the CPU would be interrupted as soon as it has stopped receiving continuous data. This mode can also be used to indicate when the serial line has been marking for longer than the programmed time limit. In this case, the CPU has read all the characters from the FIFO, but the last character received has started the count. If there is no new data during the programmed time interval, the counter ready bit will get set, and an interrupt can be generated.

Note: The Receiver Time-out mode can be Enabled and Disabled by writing the appropriate commands to the Channel Command Register (see Table 3).

4. Register Addressing differences between the two devices

Each of these devices consists of Read Only and Write Only registers. There are numerous cases where, for a given DUART address, the register that is accessed during a read is not the same as that accessed during a write. Hence, I have divided the DUART registers into two groups: Read Mode and Write Mode registers.

Read Mode registers are those registers that are accessed during a "read" to a particular DUART Address. Likewise, Write Mode registers are those registers that are accessed during a "write" to a particular DUART Address. Tables 1 and 2 presents the Read and

A Comparison between Exar's XR-88C681 with Signetics' SC26C92 DUART Devices

Write Mode Register Addressing for both the XR-88C681 and the Signetics SC26C92 device, respectively.

A Comparison between Exar's XR-88C681 with Signetics' SC26C92 DUART Devices

Table 1, Read Register Addressing

A3	A2	A1	A0	SC26C92	XR-88C681
0	0	0	0	Mode Register A (***MR0A***, MR1A, MR2A)	Mode Register A MR1A, MR2A)
0	0	0	1	Status Register A (SRA)	Status Register A (SRA)
0	***0***	***1***	***0***	***Reserved***	***Interrupt Status Register, Masked, ISR_M***
0	0	1	1	Rx Holding Register A (RHRA)	Rx Holding Register A (RHRA)
0	1	0	0	Input Port Change Register (IPCR)	Input Port Change Control Register (IPCR)
0	1	0	1	Interrupt Status Register (ISR)	Interrupt Status Register (ISR)
0	1	1	0	Counter/Timer Upper Byte (CTUR)	Counter/Timer Upper Byte (CTUR)
0	1	1	1	Counter/Timer Lower Byte (CTLR)	Counter/Timer Lower Byte (CTLR)
1	0	0	0	Mode Register B (***MR0B***, MR1B, MR2B)	Mode Register B (MR1B, MR2B)
1	0	0	1	Status Register B (SRB)	Status Register B (SRB)
1	0	1	0	Reserved	Reserved
1	0	1	1	Rx Holding Register B (RHRB)	Rx Holding Register B (RHRB)
1	***1***	***0***	***0***	***Reserved***	***Interrupt Vector Register (IVR)***
1	1	0	1	Input Port (PR)	Input Port (PR)
1	1	1	0	Start Counter Command	Start Counter Command
1	1	1	1	Stop Counter Command	Stop Counter Command

Note: Differences between the two devices are written in Bold-Italics
Shaded boxes denote "Address Triggered" commands.

Table 2, Write Register Addressing

A3	A2	A1	A0	SC26C92	XR-88C681
0	0	0	0	Mode Register A (***MR0A***, MR1A, MR2A)	Mode Register A (MR1A, MR2A)
0	0	0	1	Clock Select Register A (CSRA)	Clock Select Register A (CSRA)
0	0	1	0	Command Register A (CRA)	Command Register A (CRA)
0	0	1	1	Tx Holding Register A (THRA)	Tx Holding Register A (THRA)
0	1	0	0	Aux. Control Register (ACR)	Aux. Control Register (ACR)
0	1	0	1	Interrupt Mask Register (IMR)	Interrupt Mask Register (IMR)
0	1	1	0	Counter/Timer Upper Byte Register (CTUR)	Counter/Timer Upper Byte Register (CTUR)
0	1	1	1	Counter/Timer Lower Byte Register (CTLR)	Counter/Timer Lower Byte Register (CTLR)
1	0	0	0	Mode Register B (***MR0B***, MR1B, MR2B)	Mode Register B (MR1B, MR2B)
1	0	0	1	Clock Select Register B (CSRB)	Clock Select Register B (CSRB)
1	0	1	0	Command Register B	Command Register B (CRB)
1	0	1	1	Tx Holding Register B (THRB)	Tx Holding Register B (THRB)
1	***1***	***0***	***0***	***Reserved***	***Interrupt Vector Register (IVR)***
1	1	0	1	Output Port Configuration Register (OPCR)	Output Port Configuration Register (OPCR)
1	1	1	0	Set Output Port Bits Command	Set Output Port Bits Commands
1	1	1	1	Reset Output Port Bits Command	Reset Output Port Bits Command

Note: Differences between the two devices are written in Bold Italics
Shaded boxes denote "Address Triggered" commands.

5. Differences in Commands being issued to the Command Register, between the XR-88C681 and the SC26C92 devices.

The Command Register Format for both the XR-88C681 and the SC26C92 devices is as follows:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Miscellaneous Commands				**Tx Enable/Disable**		**Rx Enable/Disable**	
See Table 3				00 = No Change 01 = Enable Tx 10 = Disable Tx 11 = Undefined, Do Not Use		00 = No Change 01 = Enable Rx 10 = Disable Rx 11 = Undefined, Do Not Use	

The Lower Nibble of the Channel Command Register is for Enabling/Disabling the Transmitter and Receiver. The Upper Nibble of the Channel Command Register is for Miscellaneous Commands. There are some differences in the sets of these Miscellaneous Commands, between these two devices. These differences are listed in Table 3.

Table 3, Comparison of Commands to the Command Register between the Two Devices

First 4 bits of Command Word to CR	SC26C92	XR-88C681
0 0 0 0	NULL Command	NULL Command
0 0 0 1	Reset MR Pointer[1]	Reset MR Pointer
0 0 1 0	Reset Rx	Reset Rx
0 0 1 1	Reset Tx	Reset Tx
0 1 0 0	Reset Error Status	Reset Error Status
0 1 0 1	Reset Break Change Interrupt	Reset Break Change Interrupt
0 1 1 0	Start Break	Start Break
0 1 1 1	Stop Break	Stop Break
1 0 0 0	***Assert RTSN[3]***	***Set Rx BRG Select Extend Bit***
1 0 0 1	***Negate RTSN[4]***	***Clear Rx BRG Select Extend Bit***
1 0 1 0	***Set Time-out Mode ON***	***Set Tx BRG Select Extend Bit***
1 0 1 1	***Set MR Pointer to "0"[2]***	***Clear Tx BRG Select Extend Bit***
1 1 0 0	***Disable Time-out Mode***	***Set Standby Mode(A)/ Reset IUS Latch (B)***
1 1 0 1	***Not Used***	***Set Active Mode (A)/Set Z Mode (B)***
1 1 1 0	***Power Down Mode On(A)/Reserved (B)***	***Reserved***
1 1 1 1	***Disable Power Down Mode (A)/Reserved (B)***	***Reserved***

Notes:

[1] Writing 0001 to the upper nibble of the Channel n Command Register will reset the MRn Pointer to MR1n.

[2] Writing 1011 to the upper nibble of the Channel n Command Register will set the MRn Pointer to MRn0.

[3] Writing 1000 to the upper nibble of the Channel n Command Register will assert the active-low RTSn output.

[4] Writing 1001 to the upper nibble of the Channel n Command Register will negate the active-low RTSn output.