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

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

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

Contact us

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

GENERAL DESCRIPTION

The XR16V698[1] (698), is a 2.25V to 3.6V octal Universal Asynchronous Receiver and Transmitter (UART) with 5V tolerant inputs. The highly integrated device is designed for high bandwidth requirement in communication systems. The global interrupt source register provides a complete interrupt status indication for all 8 channels to speed up interrupt parsing. Each UART has its own 16C550 compatible set of configuration registers, TX and RX FIFOs of 32 bytes, fully programmable transmit and receive FIFO trigger levels, automatic RTS/CTS or DTR/DSR hardware flow control with programmable hysteresis, automatic software (Xon/Xoff) flow control, RS-485 half-duplex direction control with programmable turn-around delay, Intel or Motorola bus interface and sleep mode with a wake-up indicator.

NOTE: *Covered by US patents #5,649,122 and #5,949,787*

APPLICATIONS

- Remote Access Servers
- Ethernet Network to Serial Ports
- Network Management
- Factory Automation and Process Control
- Point-of-Sale Systems
- Multi-port RS-232/RS-422/RS-485 Cards

FEATURES

- 2.25V to 3.6V with 5V Tolerant Inputs
- Single Interrupt output for all 8 UARTs
- A Global Interrupt Source Register for all 8 UARTs
- 5G “Flat” UART Registers for easier programming
- Simultaneous Initialization of all UART channels
- General Purpose 16-bit Timer/counter
- Sleep Mode with Wake-up Indication
- Highly Integrated Device for Space Saving
- Each UART is independently controlled with:
  - 16C550 Compatible 5G Register Set
  - 32-byte Transmit and Receive FIFOs
  - Fractional Baud Rate Generator
  - Programmable TX and RX FIFO Trigger Level
  - Automatic RTS/CTS or DTR/DSR Flow Control
  - Automatic Xon/Xoff Software Flow Control
  - RS-485 Half-Duplex Direction Control Output with Selectable Turn-around Delay
  - Infrared (IrDA 1.0) Data Encoder/Decoder
  - Programmable Data Rate with Prescaler
- Up to 15 Mbps Serial Data Rate
- Pin compatible to XR16V798. Same 100-pin QFP Package (14x20x3 mm)

FIGURE 1. BLOCK DIAGRAM

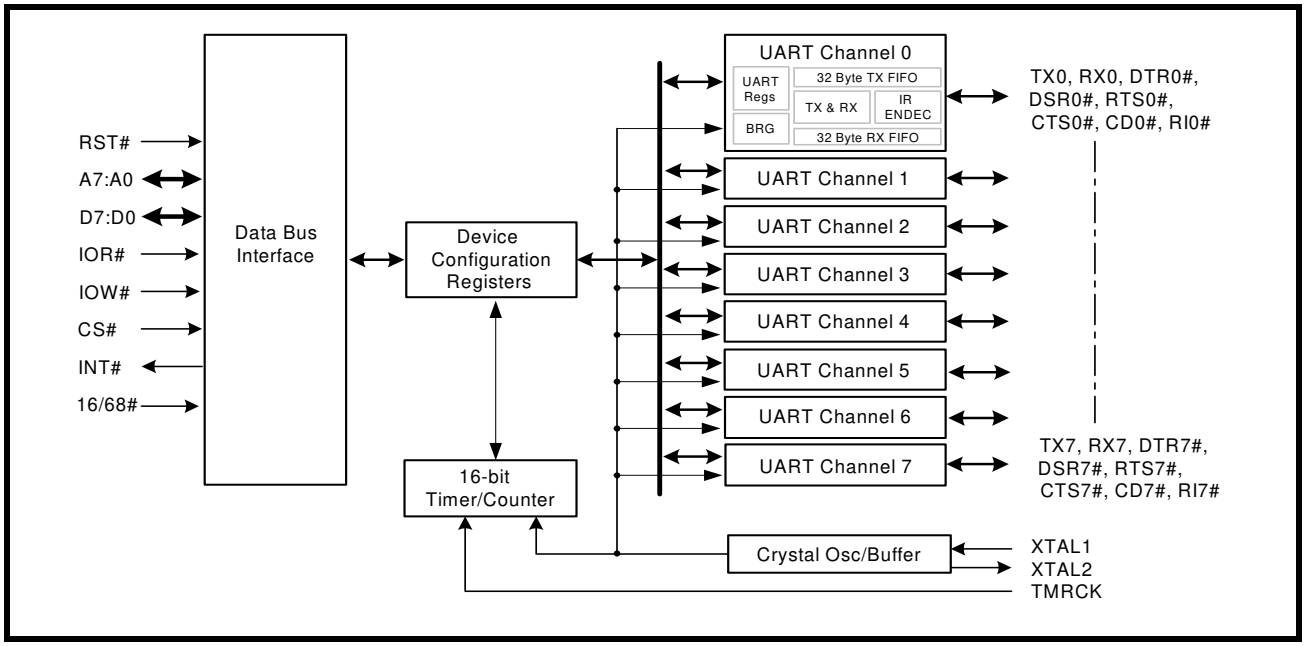
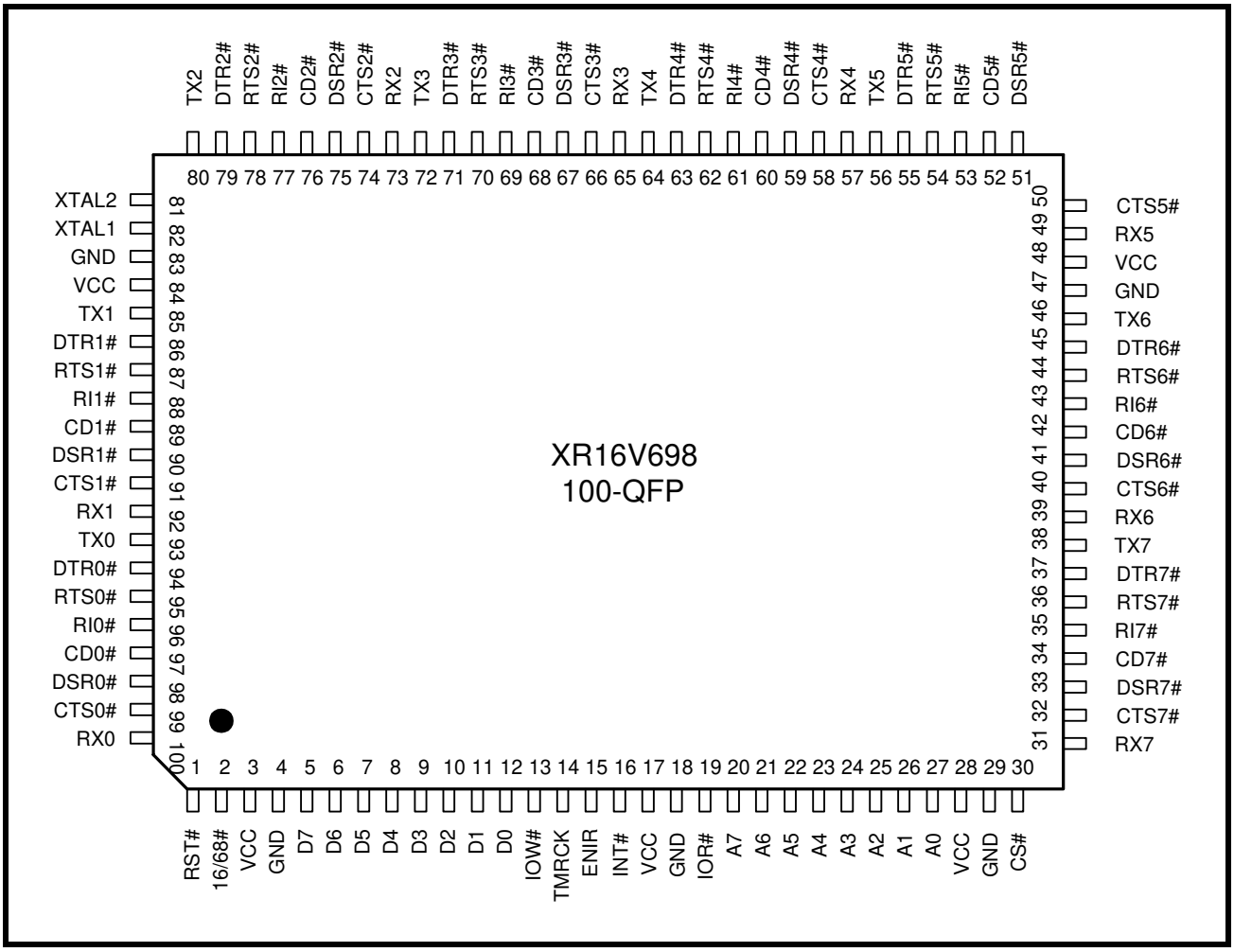


FIGURE 2. PIN OUT OF THE DEVICE



ORDERING INFORMATION

PART NUMBER	PACKAGE	OPERATING TEMPERATURE RANGE	DEVICE STATUS
XR16V698IQ100	100-Lead QFP	-40°C to +85°C	Active

PIN DESCRIPTIONS

NAME	PIN #	TYPE	DESCRIPTION
DATA BUS INTERFACE			
A7:A0	20-27	I	Address lines [7:0]. A0:A3 selects individual UART's 16 configuration registers, A4:A6 selects UART channel 0 to7, and A7 selects the global device configuration registers.
D7:D0	5-12	IO	Data bus lines [7:0] (bidirectional).
IOR#	19	I	When 16/68# pin is HIGH, it selects Intel bus interface and this input is read strobe (active LOW). The falling edge instigates an internal read cycle and retrieves the data byte from an internal register pointed by the address lines [A7:A0], places it on the data bus to allow the host processor to read it on the leading edge. When 16/68# pin is LOW, it selects Motorola bus interface and this input should be connected to VCC.
IOW# (R/W#)	13	I	When 16/68# pin is HIGH, it selects Intel bus interface and this input becomes write strobe (active LOW). The falling edge instigates the internal write cycle and the leading edge transfers the data byte on the data bus to an internal register pointed by the address lines. When 16/68# pin is LOW, it selects Motorola bus interface and this input becomes read (HIGH) and write (LOW) signal (R/W#).
CS#	30	I	When 16/68# pin is HIGH, this input is chip select (active LOW) to enable the XR16V698 device. When 16/68# pin is LOW, this input becomes the read and write strobe (active LOW) for the Motorola bus interface.
INT#	16	OD	Global interrupt output from XR16V698 (open drain, active LOW). This output requires an external pull-up resistor (47K-100K ohms) to operate properly. It may be shared with other devices in the system to form a single interrupt line to the host processor and have the software driver polls each device for the interrupt status.
MODEM OR SERIAL I/O INTERFACE			
TX0	93	O	UART channel 0 Transmit Data or infrared transmit data.
RX0	100	I	UART channel 0 Receive Data or infrared receive data. Normal RXD input idles HIGH. The infrared pulse can be inverted internally prior to decoding by setting FCTR bit-4.
RTS0#	95	O	UART channel 0 Request to Send or general purpose output (active LOW). This port may be used for one of two functions: 1) Auto hardware flow control, see EFR bit-6, MCR bits-1 & 2, FCTR bits 0-3 and IER bit-6 2) RS-485 half-duplex direction control, see FCTR bit-5, MCR bit-2 and MSR bits 0-7.
CTS0#	99	I	UART channel 0 Clear to Send or general purpose input (active LOW). It can be used for auto hardware flow control, see EFR bit-7, MCR bit-2 and IER bit-7.
DTR0#	94	O	UART channel 0 Data Terminal Ready or general purpose output (active LOW). This port may be used forone of two functions. 1) auto hardware flow control, see EFR bit-6, FCTR bits-0 to 3, MCR bits-0 & 2, and IER bit-6 2) RS-485 half-duplex direction control, see FCTR bit-5, MCR bit-2 and MSR bits 0-7.
DSR0#	98	I	UART channel 0 Data Set Ready or general purpose input (active LOW). It can be used for auto hardware flow control, see EFR bit-7, MCR bit-2 and IER bit-7.
CD0#	97	I	UART channel 0 Carrier Detect or general purpose input (active LOW).

NAME	PIN #	TYPE	DESCRIPTION
RI0#	96	I	UART channel 0 Ring Indicator or general purpose input (active LOW).
TX1	85	O	UART channel 1 Transmit Data or infrared transmit data.
RX1	92	I	UART channel 1 Receive Data or infrared receive data. Normal RXD input idles HIGH. The infrared pulse can be inverted internally prior to decoding by setting FCTR bit-4.
RTS1#	87	O	UART channel 1 Request to Send or general purpose output (active LOW). See description of RTS0# pin.
CTS1#	91	I	UART channel 1 Clear to Send or general purpose input (active LOW). See description of CTS0# pin.
DTR1#	86	O	UART channel 1 Data Terminal Ready or general purpose output (active LOW). See description of DTR0# pin.
DSR1#	90	I	UART channel 1 Data Set Ready or general purpose input (active LOW). See description of DSR0# pin.
CD1#	89	I	UART channel 1 Carrier Detect or general purpose input (active LOW).
RI1#	88	I	UART channel 1 Ring Indicator or general purpose input (active LOW).
TX2	80	O	UART channel 2 Transmit Data or infrared transmit data.
RX2	73	I	UART channel 2 Receive Data or infrared receive data. Normal RXD input idles HIGH. The infrared pulse can be inverted internally prior to decoding by setting FCTR bit-4.
RTS2#	78	O	UART channel 2 Request to Send or general purpose output (active LOW). See description of RTS0# pin.
CTS2#	74	I	UART channel 2 Clear to Send or general purpose input (active LOW). See description of CTS0# pin.
DTR2#	79	O	UART channel 2 Data Terminal Ready or general purpose output (active LOW). See description of DTR0# pin.
DSR2#	75	I	UART channel 2 Data Set Ready or general purpose input (active LOWactive LOW). See description of DSR0# pin.
CD2#	76	I	UART channel 2 Carrier Detect or general purpose input (active LOW).
RI2#	77	I	UART channel 2 Ring Indicator or general purpose input (active LOW).
TX3	72	O	UART channel 3 Transmit Data or infrared transmit data.
RX3	65	I	UART channel 3 Receive Data or infrared receive data. Normal RXD input idles HIGH. The infrared pulse can be inverted internally prior to decoding by setting FCTR bit-4.
RTS3#	70	O	UART channel 3 Request to Send or general purpose output (active LOW). See description of RTS0# pin.
CTS3#	66	I	UART channel 3 Clear to Send or general purpose input (active LOW). See description of CTS0# pin.
DTR3#	71	O	UART channel 3 Data Terminal Ready or general purpose output (active LOW). See description of DTR0# pin.
DSR3#	67	I	UART channel 3 Data Set Ready or general purpose input (active LOW). See description of DSR0# pin.
CD3#	68	I	UART channel 3 Carrier Detect or general purpose input (active LOW).
RI3#	69	I	UART channel 3 Ring Indicator or general purpose input (active LOW).

NAME	PIN #	TYPE	DESCRIPTION
TX4	64	O	UART channel 4 Transmit Data or infrared transmit data.
RX4	57	I	UART channel 4 Receive Data or infrared receive data. Normal RXD input idles HIGH. The infrared pulse can be inverted internally prior to decoding by setting FCTR bit-4.
RTS4#	62	O	UART channel 4 Request to Send or general purpose output (active LOW). See description of RTS0# pin.
CTS4#	58	I	UART channel 4 Clear to Send or general purpose input (active LOW). See description of CTS0# pin.
DTR4#	63	O	UART channel 4 Data Terminal Ready or general purpose output (active LOW). See description of DTR0# pin.
DSR4#	59	I	UART channel 4 Data Set Ready or general purpose input (active LOW). See description of DSR0# pin.
CD4#	60	I	UART channel 4 Carrier Detect or general purpose input (active LOW).
RI4#	61	I	UART channel 4 Ring Indicator or general purpose input (active LOW).
TX5	56	O	UART channel 5 Transmit Data or infrared transmit data.
RX5	49	I	UART channel 5 Receive Data or infrared receive data. Normal RXD input idles HIGH. The infrared pulse can be inverted internally prior to decoding by setting FCTR bit-4.
RTS5#	54	O	UART channel 5 Request to Send or general purpose output (active LOW). See description of RTS0# pin.
CTS5#	50	I	UART channel 5 Clear to Send or general purpose input (active LOW). See description of CTS0# pin.
DTR5#	55	O	UART channel 5 Data Terminal Ready or general purpose output (active LOW). See description of DTR0# pin.
DSR5#	51	I	UART channel 5 Data Set Ready or general purpose input (active LOW). See description of DSR0# pin.
CD5#	52	I	UART channel 5 Carrier Detect or general purpose input (active LOW).
RI5#	53	I	UART channel 5 Ring Indicator or general purpose input (active LOW).
TX6	46	O	UART channel 6 Transmit Data or infrared transmit data.
RX6	39	I	UART channel 6 Receive Data or infrared receive data. Normal RXD input idles HIGH. The infrared pulse can be inverted internally prior to decoding by setting FCTR bit-4.
RTS6#	44	O	UART channel 6 Request to Send or general purpose output (active LOW). See description of RTS0# pin.
CTS6#	40	I	UART channel 6 Clear to Send or general purpose input (active LOW). See description of CTS0# pin.
DTR6#	45	O	UART channel 6 Data Terminal Ready or general purpose output (active LOW). See description of DTR0# pin.
DSR6#	41	I	UART channel 6 Data Set Ready or general purpose input (active LOW). See description of DSR0# pin.
CD6#	42	I	UART channel 6 Carrier Detect or general purpose input (active LOW).
RI6#	43	I	UART channel 6 Ring Indicator or general purpose input (active LOW).
TX7	38	O	UART channel 7 Transmit Data or infrared transmit data.

NAME	PIN #	TYPE	DESCRIPTION
RX7	31	I	UART channel 7 Receive Data or infrared receive data. Normal RXD input idles HIGH. The infrared pulse can be inverted internally prior to decoding by setting FCTR bit-4.
RTS7#	36	O	UART channel 7 Request to Send or general purpose output (active LOW). See description of RTS0# pin.
CTS7#	32	I	UART channel 7 Clear to Send or general purpose input (active LOW). See description of CTS0# pin.
DTR7#	37	O	UART channel 7 Data Terminal Ready or general purpose output (active LOW). See description of DTR0# pin.
DSR7#	33	I	UART channel 7 Data Set Ready or general purpose input (active LOW). See description of DSR0# pin.
CD7#	34	I	UART channel 7 Carrier Detect or general purpose input (active LOW).
RI7#	35	I	UART channel 7 Ring Indicator or general purpose input (active LOW).
ANCILLARY SIGNALS			
XTAL1	82	I	Crystal or external clock input. Caution: this input is not 5V tolerant.
XTAL2	81	O	Crystal or buffered clock output.
TMRCK	14	I	16-bit timer/counter external clock input.
ENIR	15	I	Infrared mode enable (active HIGH). This pin is sampled during power up, following a hardware reset (RST#) or soft-reset (register RESET). It can be used to start up all 8 UARTs in the infrared mode. The sampled logic state is transferred to MCR bit-6 in the UART.
RST#	1	I	Reset (active LOW). The XR16V698 does not have a Power-on reset. Therefore, a hardware reset must be issued using this pin during power-up. The configuration and UART registers are reset to default values, see **Table 18**.
16/68#	2	I	Intel or Motorola data bus interface select. The Intel bus interface is selected when this input is HIGH and the Motorola bus interface is selected when this input is LOW. This input affects the functionality of IOR#, IOW# and CS# pins.
VCC	3,17,28,48,84		+2.25V to 3.6V supply with 5V tolerant serial (modem) inputs.
GND	4,18,29,47,83		Power supply common, ground.

NOTE: *Pin type: I=Input, O=Output, IO= Input/output, OD=Output Open Drain.*

1.0 DESCRIPTION

The XR16V698 (698) integrates the functions of 8 enhanced 16550 UARTs, a general purpose 16-bit timer/counter and an on-chip oscillator. The device configuration registers include a set of four consecutive interrupt source registers that provides interrupt-status for all 8 UARTs, timer/counter and a sleep wake up indicator. Each UART channel has its own 16550 UART compatible configuration register set for individual channel control, status, and data transfer. Additionally, each UART channel has 32-byte of transmit and receive FIFOs, automatic RTS/CTS or DTR/DSR hardware flow control with hysteresis control, automatic Xon/Xoff and special character software flow control, programmable transmit and receive FIFO trigger levels, infrared encoder and decoder (IrDA ver. 1.0), programmable baud rate generator with a prescaler of divide by 1 or 4, and data rate up to 15Mbps with 4X sampling clock or 7.5Mbps with 8X sampling clock at 3.3V and 10Mbps with 4X sampling clock or 5Mbps with 8X sampling clock at 2.5V. The XR16V598 is a 2.25-3.6V device with 5 volt tolerant inputs (except XTAL1).

2.0 FUNCTIONAL DESCRIPTIONS

2.1 Device Reset

2.1.1 Hardware Reset

The RST# input resets the internal registers and the serial interface outputs in all 8 channels to their default state (see Table 18). A LOW pulse of longer than 40 ns duration will be required to activate the reset function in the device.

2.1.2 Software Reset

The internal registers of each UART can be reset by writing to the RESET register in the Device Configuration Registers. For more details, see the RESET register description on page 29.

2.2 UART Channel Selection

A LOW on the chip select pin, CS#, allows the user to select one of the UART channels to configure, send transmit data and/or unload receive data to/from the UART. When address line A7 = 0, address lines A6:A4 are used to select one of the eight channels. See **Table 1** below for UART channel selection.

TABLE 1: UART CHANNEL SELECTION

A7	A6	A5	A4	FUNCTION
0	0	0	0	Channel 0 Selected
0	0	0	1	Channel 1 Selected
0	0	1	0	Channel 2 Selected
0	0	1	1	Channel 3 Selected
0	1	0	0	Channel 4 Selected
0	1	0	1	Channel 5 Selected
0	1	1	0	Channel 6 Selected
0	1	1	1	Channel 7 Selected

2.3 Simultaneous Write to All Channels

During a write cycle, the setting of the Device Configuration register REGB (**See Table 8**) bit-0 to a logic 1 will override the channel selection of address A6:A4 and allow a simultaneous write to all 8 UART channels when any channel is written to. This functional capability allow the registers in all 8 UART channels to be modified concurrently, saving individual channel initialization time. Caution should be considered, however, when using this capability. Any in-process serial data transfer may be disrupted by changing an active channel's mode.

2.4 INT# Ouput

The INT# interrupt output changes according to the operating mode and enhanced features setup. **Table 2 and 3** summarize the operating behavior for the transmitter and receiver.

TABLE 2: INT# PIN OPERATION FOR TRANSMITTER

Auto RS-485 Mode	FCR BIT-0 = 0 (FIFO DISABLED)	FCR BIT-0 = 1 (FIFO ENABLED)
NO	HIGH = a byte in THR LOW = THR empty	HIGH = FIFO above trigger level LOW = FIFO below trigger level or FIFO empty
YES	HIGH = a byte in THR LOW = transmitter empty	HIGH = FIFO above trigger level LOW = FIFO below trigger level or transmitter empty

TABLE 3: INT# PIN OPERATION FOR RECEIVER

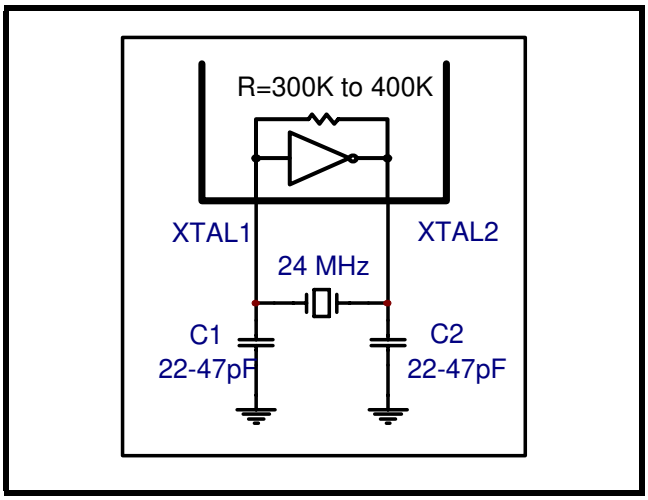
FCR BIT-0 = 0 (FIFO DISABLED)	FCR BIT-0 = 1 (FIFO ENABLED)
HIGH = no data LOW = 1 byte	HIGH = FIFO below trigger level LOW = FIFO above trigger level

2.5 Crystal Oscillator

The 698 includes an on-chip oscillator. The crystal oscillator provides the system clock to the Baud Rate Generators (BRG) in each of the 8 UARTs, the 16-bit general purpose timer/counter and internal logics. XTAL1 is the input to the oscillator or external clock buffer input with XTAL2 pin being the output. For programming details, see **"Section 2.6, Programmable Baud Rate Generator with Fractional Divisor" on page 9**.

The on-chip oscillator is designed to use an industry standard microprocessor crystal (parallel resonant, fundamental frequency with 10-22 pF capacitance load, ESR of 20-120 ohms and 100ppm frequency tolerance) connected externally between the XTAL1 and XTAL2 pins (see **Figure 3**). Alternatively, an external clock can be connected to the XTAL1 pin to clock the internal 8 baud rate generators for standard or custom rates. The typical oscillator connections are shown in **Figure 3**. For further reading on oscillator circuit please see application note DAN108 on EXAR's web site.

FIGURE 3. TYPICAL OSCILLATOR CONNECTIONS



2.6 Programmable Baud Rate Generator with Fractional Divisor

Each UART has its own Baud Rate Generator (BRG) with a prescaler for the transmitter and receiver. The prescaler is controlled by a software bit in the MCR register. The MCR register bit-7 sets the prescaler to divide the input crystal or external clock by 1 or 4. The output of the prescaler clocks to the BRG. The BRG further divides this clock by a programmable divisor between 1 and (2^{16} - 0.0625) in increments of 0.0625 (1/16) to obtain a 16X or 8X or 4X sampling clock of the serial data rate. The sampling clock is used by the transmitter for data bit shifting and receiver for data sampling. The BRG divisor (DLL, DLM and DLD registers) defaults to the value of '1' (DLL = 0x01, DLM = 0x00 and DLD = 0x00) upon reset. Therefore, the BRG must be programmed during initialization to the operating data rate. The DLL and DLM registers provide the integer part of the divisor and the DLD register provides the fractional part of the dvisior. Only the four lower bits of the DLD are implemented and they are used to select a value from 0 (for setting 0000) to 0.9375 or 15/16 (for setting 1111). Programming the Baud Rate Generator Registers DLL, DLM and DLD provides the capability for selecting the operating data rate. **Table 4** shows the standard data rates available with a 24MHz crystal or external clock at 16X clock rate. If the pre-scaler is used (MCR bit-7 = 1), the output data rate will be 4 times less than that shown in **Table 4**. At 8X sampling rate, these data rates would double. Also, when using 8X sampling mode, please note that the bit-time will have a jitter (+/- 1/16) whenever the DLD is non-zero and is an odd number. At 4X sampling rate, these data rates would quadruple. When using a non-standard data rate crystal or external clock, the divisor value can be calculated with the following equation(s):

8XMODE [7:0] = 0x00 4XMODE [7:0] = 0x00	Required Divisor (decimal) = (XTAL1 clock frequency / prescaler) / (serial data rate x 16)
8XMODE [7:0] = 0xFF 4XMODE [7:0] = 0x00	Required Divisor (decimal) = (XTAL1 clock frequency / prescaler / (serial data rate x 8)
8XMODE [7:0] = 0x00 4XMODE [7:0] = 0xFF	Required Divisor (decimal) = (XTAL1 clock frequency / prescaler / (serial data rate x 4)
8XMODE [7:0] = 0xFF 4XMODE [7:0] = 0xFF	Reserved.

The closest divisor that is obtainable in the 698 can be calculated using the following formula:

ROUND((Required Divisor - TRUNC(Required Divisor))*16)/16 + TRUNC(Required Divisor), where

DLM = TRUNC(Required Divisor) >> 8
DLL = TRUNC(Required Divisor) & 0xFF
DLD = ROUND((Required Divisor-TRUNC(Required Divisor))*16)

In the formulas above, please note that:

TRUNC (N) = Integer Part of N. For example, TRUNC (5.6) = 5.

ROUND (N) = N rounded towards the closest integer. For example, ROUND (7.3) = 7 and ROUND (9.9) = 10.

A >> B indicates right shifting the value 'A' by 'B' number of bits. For example, 0x78A3 >> 8 = 0x0078.

FIGURE 4. BAUD RATE GENERATOR

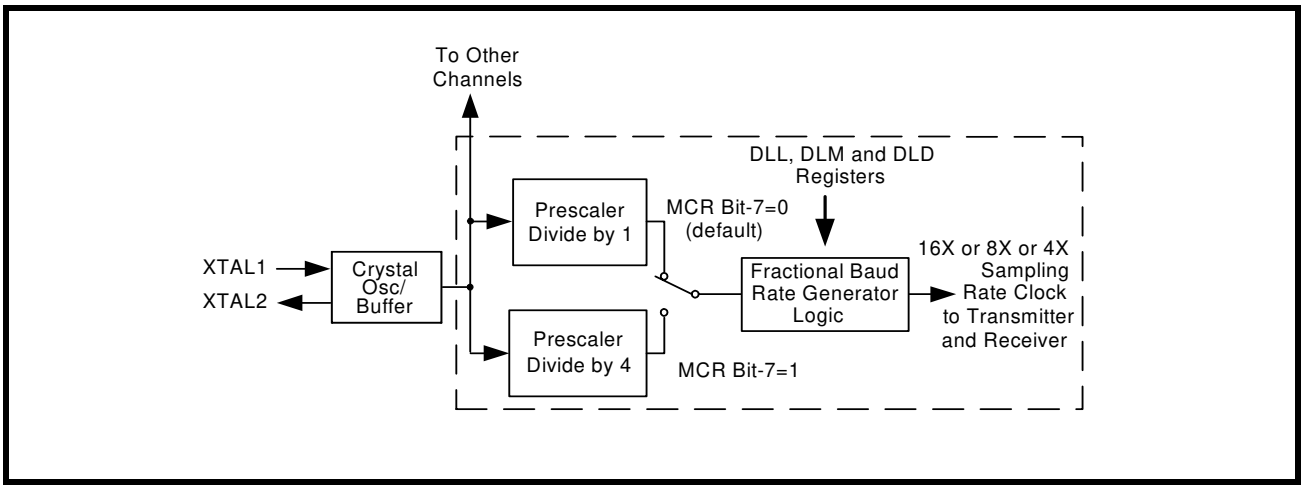


TABLE 4: TYPICAL DATA RATES WITH A 24 MHZ CRYSTAL OR EXTERNAL CLOCK AT 16X SAMPLING

Required Output Data Rate	DIVISOR FOR 16x Clock (Decimal)	DIVISOR OBTAINABLE IN 698	DLM PROGRAM VALUE (HEX)	DLL PROGRAM VALUE (HEX)	DLD PROGRAM VALUE (HEX))	DATA ERROR RATE (%)
400	3750	3750	E	A6	0	0
2400	625	625	2	71	0	0
4800	312.5	312 8/16	1	38	8	0
9600	156.25	156 4/16	0	9C	4	0
10000	150	150	0	96	0	0
19200	78.125	78 2/16	0	4E	2	0
25000	60	60	0	3C	0	0
28800	52.0833	52 1/16	0	34	1	0.04
38400	39.0625	39 1/16	0	27	1	0
50000	30	30	0	1E	0	0
57600	26.0417	26 1/16	0	1A	1	0.08
75000	20	20	0	14	0	0
100000	15	15	0	F	0	0
115200	13.0208	13	0	D	0	0.16
153600	9.7656	9 12/16	0	9	C	0.16
200000	7.5	7 8/16	0	7	8	0
225000	6.6667	6 11/16	0	6	B	0.31
230400	6.5104	6 8/16	0	6	8	0.16
250000	6	6	0	6	0	0
300000	5	5	0	5	0	0

TABLE 4: TYPICAL DATA RATES WITH A 24 MHZ CRYSTAL OR EXTERNAL CLOCK AT 16X SAMPLING

Required Output Data Rate	DIVISOR FOR 16x Clock (Decimal)	DIVISOR OBTAINABLE IN 698	DLM PROGRAM VALUE (HEX)	DLL PROGRAM VALUE (HEX)	DLD PROGRAM VALUE (HEX))	DATA ERROR RATE (%)
400000	3.75	3 12/16	0	3	C	0
460800	3.2552	3 4/16	0	3	4	0.16
500000	3	3	0	3	0	0
750000	2	2	0	2	0	0
921600	1.6276	1 10/16	0	1	A	0.16
1000000	1.5	1 8/16	0	1	8	0

2.7 Transmitter

The transmitter section comprises of an 8-bit Transmit Shift Register (TSR) and 32 bytes of FIFO which includes a byte-wide Transmit Holding Register (THR). TSR shifts out every data bit with the 16X or 8X or 4X (if 8X or 4X sampling is selected via the **8XMODE Register** or 4XMODE Register) internal clock. A bit time is 16 (or 8 or 4) clock periods. The transmitter sends the start-bit followed by the number of data bits, inserts the proper parity-bit if enabled, and adds the stop-bit(s). The status of the FIFO and TSR are reported in the Line Status Register (LSR bit-5 and bit-6).

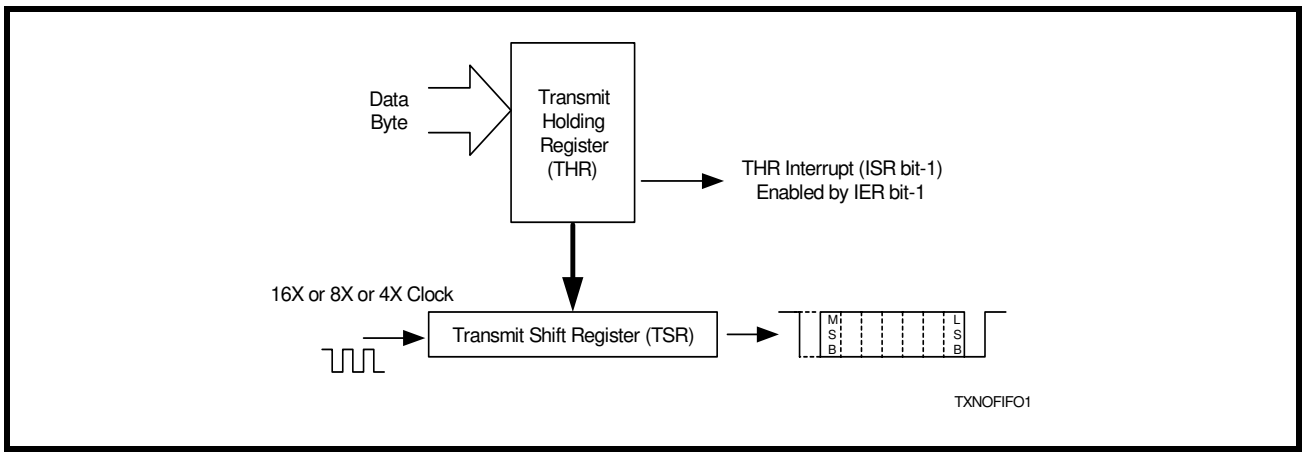
2.7.1 Transmit Holding Register (THR) - Write Only

The transmit holding register is an 8-bit register providing a data interface to the host processor. The host writes transmit data byte to the THR to be converted into a serial data stream including start-bit, data bits, parity-bit and stop-bit(s). The least-significant-bit (Bit-0) becomes first data bit to go out. The THR is the input register to the transmit FIFO of 32 bytes when FIFO operation is enabled by FCR bit-0. Every time a write operation is made to the THR, the FIFO data pointer is automatically bumped to the next sequential data location.

2.7.2 Transmitter Operation in non-FIFO Mode

The host loads transmit data to THR one character at a time. The THR empty flag (LSR bit-5) is set when the data byte is transferred to TSR. THR flag can generate a transmit empty interrupt (ISR bit-1) when it is enabled by IER bit-1. The TSR flag (LSR bit-6) is set when TSR becomes completely empty.

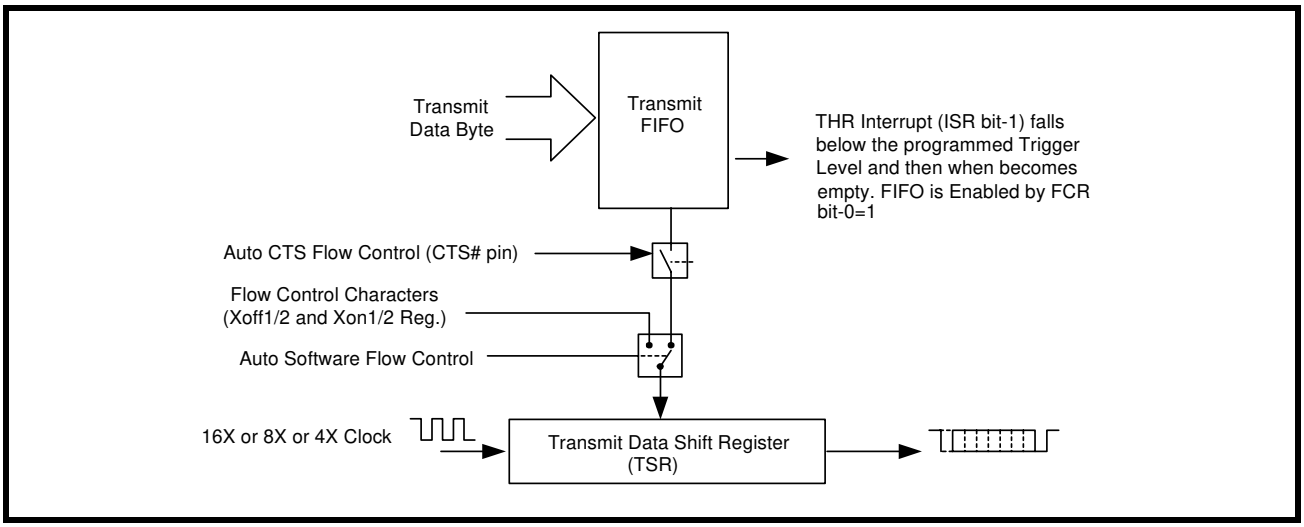
FIGURE 5. TRANSMITTER OPERATION IN NON-FIFO MODE



2.7.3 Transmitter Operation in FIFO Mode

The host may fill the transmit FIFO with up to 32 bytes of transmit data. The THR empty flag (LSR bit-5) is set whenever the FIFO is empty. The THR empty flag can generate a transmit empty interrupt (ISR bit-1) when the amount of data in the FIFO falls below its programmed trigger level. The transmit empty interrupt is enabled by IER bit-1. The TSR flag (LSR bit-6) is set when TSR/FIFO becomes empty.

FIGURE 6. TRANSMITTER OPERATION IN FIFO AND FLOW CONTROL MODE



2.8 Receiver

The receiver section contains an 8-bit Receive Shift Register (RSR) and 32 bytes of FIFO which includes a byte-wide Receive Holding Register (RHR). The RSR uses the 16X (or the 8X or the 4X) clock for timing. It verifies and validates every bit on the incoming character in the middle of each data bit. On the falling edge of a start or false start bit, an internal receiver counter starts counting the number of 16X (or 8X or 4X) clocks. After 8 (or 4 or 2) clocks the start bit period should be at the center of the start bit. At this time the start bit is sampled and if it is still a logic 0 it is validated. Evaluating the start bit in this manner prevents the receiver from assembling a false character. The rest of the data bits and stop bits are sampled and validated in this same manner to prevent false framing. If there were any error(s), they are reported in the LSR register bits 2-4. Upon unloading the receive data byte from RHR, the receive FIFO pointer is bumped and the error tags are immediately updated to reflect the status of the data byte in RHR register. RHR can generate a receive data ready interrupt upon receiving a character or delay until it reaches the FIFO trigger level. Furthermore, data delivery to the host is guaranteed by a receive data ready time-out interrupt when data is not received for 4 word lengths as defined by LCR[1:0] plus 12 bits time. This is equivalent to 3.7-4.6 character times. The RHR interrupt is enabled by IER bit-0.

2.8.1 Receive Holding Register (RHR) - Read-Only

The Receive Holding Register is an 8-bit register that holds a receive data byte from the Receive Shift Register. It provides the receive data interface to the host processor. The RHR register is part of the receive FIFO of 32 bytes by 11-bits wide, the 3 extra bits are for the 3 error tags to be reported in LSR register. When the FIFO is enabled by FCR bit-0, the RHR contains the first data character received by the FIFO. After the RHR is read, the next character byte is loaded into the RHR and the errors associated with the current data byte are immediately updated in the LSR bits 2-4.

FIGURE 7. RECEIVER OPERATION IN NON-FIFO MODE

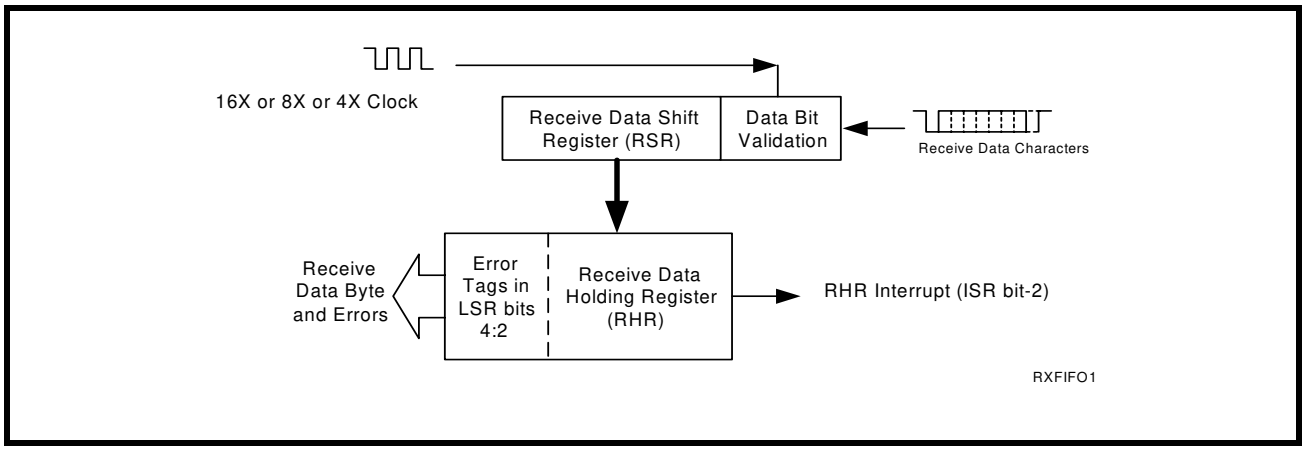
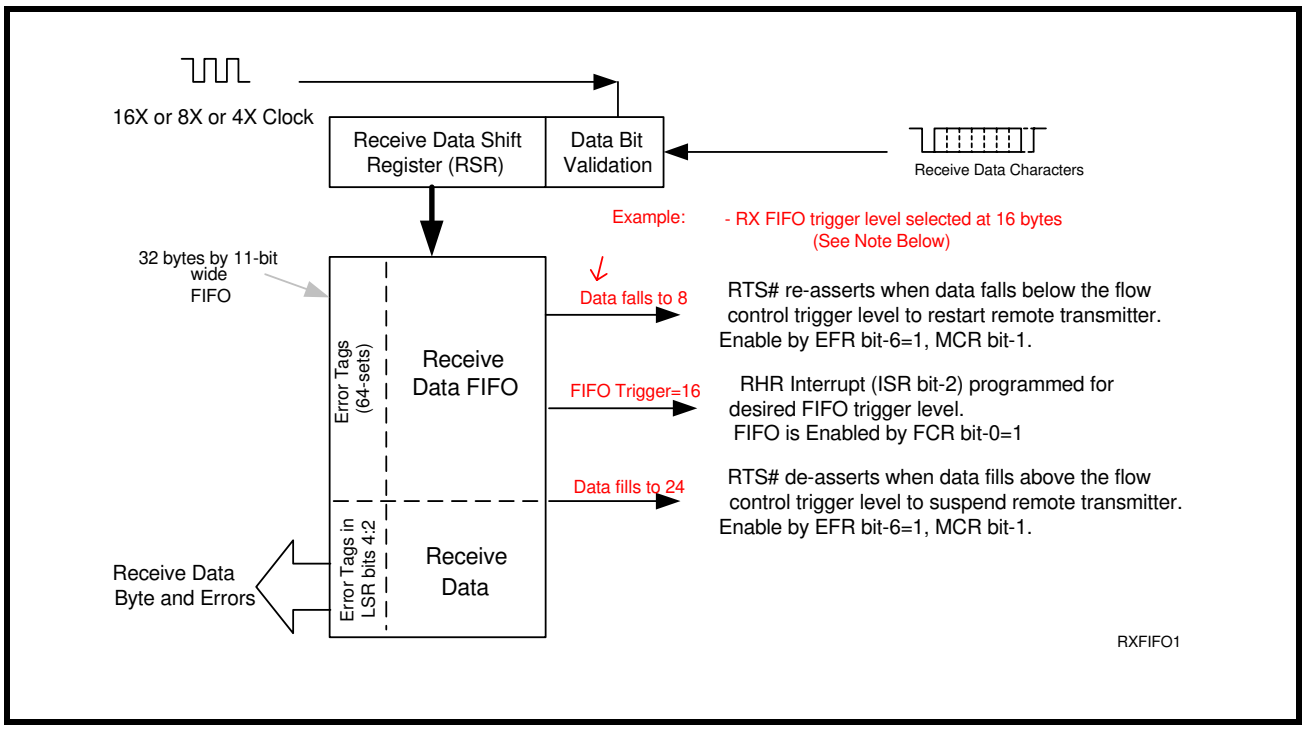


FIGURE 8. RECEIVER OPERATION IN FIFO AND AUTO RTS FLOW CONTROL MODE



2.9 THR and RHR Register Locations

The THR and RHR register addresses for channel 0 to channel 7 are shown in **Table 5** below. The THR and RHR for channels 0 to 7 are located at address 0x00, 0x10, 0x20, 0x30, 0x40, 0x50, 0x60 and 0x70 respectively. Transmit data byte is loaded to the THR when writing to that address and receive data is unloaded from the RHR register when reading that address. Both THR and RHR registers are 16C550 compatible in 8-bit format, so each bus operation can only write or read in bytes.

.

TABLE 5: TRANSMIT AND RECEIVE HOLDING REGISTER LOCATIONS, 16C550 COMPATIBLE

THR and RHR Address Locations For CH0 to CH7 (16C550 Compatible)

CH0	0x00	Write THR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH0	0x00	Read RHR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH1	0x10	Write THR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH1	0x10	Read RHR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH2	0x20	Write THR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH2	0x20	Read RHR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH3	0x30	Write THR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH3	0x30	Read RHR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH4	0x40	Write THR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH4	0x40	Read RHR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH5	0x50	Write THR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH5	0x50	Read RHR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH6	0x60	Write THR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH6	0x60	Read RHR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH7	0x70	Write THR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
CH7	0x70	Read RHR	Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0

THRRHR1

2.10 Auto RTS/DTR Hardware Flow Control Operation

Automatic RTS/DTR flow control is used to prevent data overrun to the **local** receiver FIFO. The RTS#/DTR# output pin is used to request remote unit to suspend/resume data transmission. The flow control features are individually selected to fit specific application requirement (see **Figure 9**):

- Select RTS (and CTS) or DTR (and DSR) through MCR bit-2.
- Enable auto RTS/DTR flow control using EFR bit-6.
- The auto RTS or auto DTR function must be started by asserting the RTS# or DTR# output pin (MCR bit-1 or bit-0 to a logic 1, respectively) after it is enabled.

With the Auto RTS function enabled, the RTS# output pin will not be de-asserted (HIGH) when the receive FIFO reaches the programmed trigger level, but will be de-asserted when the FIFO reaches the next trigger level (**See Table 14**). The RTS# output pin will be asserted (LOW) again after the FIFO is unloaded to the next trigger level below the programmed trigger level.

However, even under these conditions, the 698 will continue to accept data until the receive FIFO is full if the remote UART transmitter continues to send data.

- If used, enable RTS/DTR interrupt through IER bit-6 (after setting EFR bit-4). The UART issues an interrupt when the RTS#/DTR# pin makes a transition: ISR bit-5 will be set to 1.

2.10.1 Auto CTS/DSR Flow Control

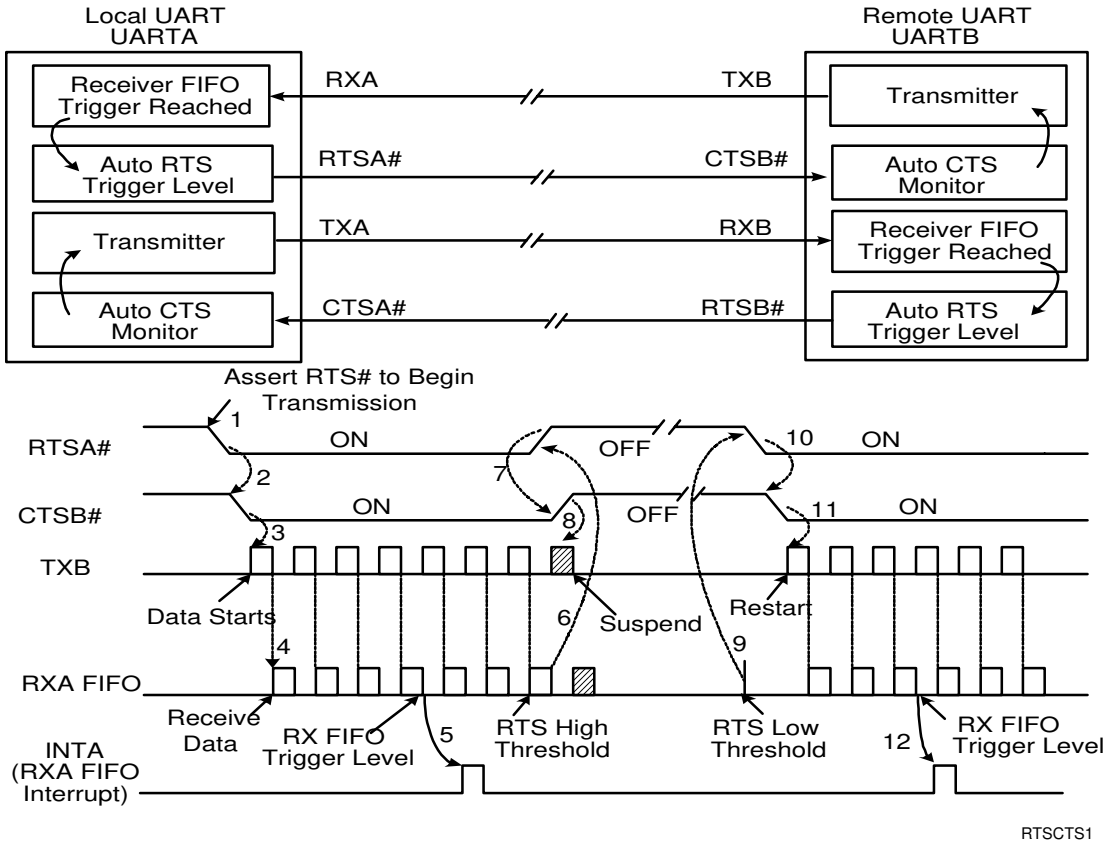
Automatic CTS/DSR flow control is used to prevent data overrun to the remote receiver FIFO. The CTS/DSR pin is monitored to suspend/restart local transmitter. The flow control features are individually selected to fit specific application requirement (see **Figure 9**):

- Select CTS (and RTS) or DSR (and DTR) through MCR bit-2.
- Enable auto CTS/DSR flow control using EFR bit-7.

With the Auto CTS or Auto DTR function enabled, the UART will suspend transmission as soon as the stop bit of the character in the Transmit Shift Register has been shifted out. Transmission is resumed after the CTS#/ DTR# input is re-asserted (LOW), indicating more data may be sent.

- If used, enable CTS/DSR interrupt through IER bit-7 (after setting EFR bit-4). The UART issues an interrupt when the CTS#/DSR# pin makes a transition: ISR bit-5 will be set to a logic 1, and UART will suspend TX transmissions as soon as the stop bit of the character in process is shifted out. Transmission is resumed after the CTS#/DSR# input returns LOW, indicating more data may be sent.

FIGURE 9. AUTO RTS/DTR AND CTS/DSR FLOW CONTROL OPERATION



The local UART (UARTA) starts data transfer by asserting RTSA# (1). RTSA# is normally connected to CTSB# (2) of remote UART (UARTB). CTSB# allows its transmitter to send data (3). TXB data arrives and fills UARTA receive FIFO (4). When RXA data fills up to its receive FIFO trigger level, UARTA activates its RXA data ready interrupt (5) and continues to receive and put data into its FIFO. If interrupt service latency is long and data is not being unloaded, UARTA monitors its receive data fill level to match the upper threshold of RTS delay and de-assert RTSA# (6). CTSB# follows (7) and request UARTB transmitter to suspend data transfer. UARTB stops or finishes sending the data bits in its transmit shift register (8). When receive FIFO data in UARTA is unloaded to match the lower threshold of RTS delay (9), UARTA re-asserts RTSA# (10), CTSB# recognizes the change (11) and restarts its transmitter and data flow again until next receive FIFO trigger (12). This same event applies to the reverse direction when UARTA sends data to UARTB with RTSB# and CTSA# controlling the data flow.

2.11 Auto Xon/Xoff (Software) Flow Control

When software flow control is enabled (See Table 17), the 698 compares one or two sequential receive data characters with the programmed Xon-1,2 or Xoff-1,2 character value(s). If receive character(s) (RX) match the programmed Xoff-1,2 value(s), the 698 will halt transmission (TX) as soon as the current character has completed transmission. When a match occurs, the Xoff (if enabled via IER bit-5) flag will be set and the interrupt output pin will be activated. Following a suspension due to a match of the Xoff character(s), the 698 will monitor the receive data stream for a match to the Xon-1,2 character(s). If a match is found, the 698 will resume operation and clear the flags (ISR bit-4).

Reset initially sets the contents of the Xon1, Xon2, Xoff1 and Xoff2 flow control registers to '0'. Following reset, any desired Xon/Xoff value can be used for software flow control. Different conditions can be set to detect Xon/Xoff characters (See Table 17) and suspend/resume transmissions. When double 8-bit Xon/Xoff characters are selected, the 698 compares two consecutive receive characters with two software flow control 8-bit values (Xon1, Xon2, Xoff1, Xoff2) and controls TX transmissions accordingly. Under the above described flow control mechanisms, flow control characters are not placed (stacked) in the user accessible RX data buffer or FIFO.

In the event that the receive buffer is overfilling and flow control needs to be executed, the 698 automatically sends an Xoff message (when enabled) via the serial TX output to the remote modem. The 698 sends the Xoff-1,2 characters two character times (= time taken to send two characters at the programmed baud rate) after the receive FIFO crosses the programmed trigger level. To clear this condition, the 698 will transmit the programmed Xon-1,2 characters as soon as receive FIFO is less than one trigger level below the programmed trigger level. Table 6 below explains this.

TABLE 6: AUTO XON/XOFF (SOFTWARE) FLOW CONTROL

RX TRIGGER LEVEL	INT PIN ACTIVATION	XOFF CHARACTER(S) SENT (CHARACTERS IN RX FIFO)	XON CHARACTER(S) SENT (CHARACTERS IN RX FIFO)
8	8	8*	0
16	16	16*	8
24	24	24*	16
28	28	28*	24

** After the trigger level is reached, an xoff character is sent after a short span of time (= time required to send 2 characters); for example, after 2.083ms has elapsed for 9600 baud and 10-bit word length setting.*

2.12 Special Character Detect

A special character detect feature is provided to detect an 8-bit character when bit-5 is set in the Enhanced Feature Register (EFR). When this character (Xoff2) is detected, it will be placed in the FIFO along with normal incoming RX data. The 698 compares each incoming receive character with Xoff-2 data. If a match exists, the received data will be transferred to FIFO and ISR bit-4 will be set to indicate detection of special character. Although the Internal Register Table shows Xon, Xoff Registers with 8 bits of character information, the actual number of bits is dependent on the programmed word length. Line Control Register (LCR) bits 0-1 defines the number of character bits, i.e., either 5 bits, 6 bits, 7 bits, or 8 bits. The word length selected by LCR bits 0-1 also determines the number of bits that will be used for the special character comparison. Bit-0 in the Xon, Xoff Registers corresponds to the LSB bit for the receive character.

2.13 Auto RS-485 Half-duplex Control

The auto RS-485 half-duplex direction control changes the behavior of the transmitter when enabled by FCTR bit-5. It also changes the behavior of the transmit empty interrupt (see **Table 2**). It asserts RTS# or DTR# (LOW) after a specified delay indicated in MSR[7:4] following the last stop bit of the last character that has been transmitted. This helps in turning around the transceiver to receive the remote station's response. The delay optimizes the time needed for the last transmission to reach the farthest station on a long cable network before switching off the line driver. This delay prevents undesirable line signal disturbance that causes signal degradation. When the host is ready to transmit next polling data packet again, it only has to load data bytes to the transmit FIFO. The transmitter automatically de-asserts RTS# or DTR# output (HIGH) prior to sending the data. The auto RS-485 half-duplex direction control also changes the transmitter empty interrupt to TSR empty instead of THR empty.

2.13.1 Normal Multidrop Mode

Normal multidrop mode is enabled when MSR bit-0 = 1 and EFR bit-5 = 0 (Special Character Detect disabled). The receiver is set to Force Parity 0 (LCR[5:3] = '111') in order to detect address bytes.

With the receiver initially disabled, it ignores all the data bytes (parity bit = 0) until an address byte is received (parity bit = 1). This address byte will cause the UART to set the parity error. The UART will generate an LSR interrupt and place the address byte in the RX FIFO. The software then examines the byte and enables the receiver if the address matches its slave address, otherwise, it does not enable the receiver.

If the receiver has been enabled, the receiver will receive the subsequent data. If an address byte is received, it will generate an LSR interrupt. The software again examines the byte and if the address matches its slave address, it does not have to anything. If the address does not match its slave address, then the receiver should be disabled.

2.13.2 Auto Address Detection

Auto address detection mode is enabled when MSR bit-0 = 1 and EFR bit-5 = 1. The desired slave address will need to be written into the XOFF2 register. The receiver will try to detect an address byte that matches the porgrammed character in the XOFF2 register. If the received byte is a data byte or an address byte that does not match the programmed character in the XOFF2 register, the receiver will discard these data. Upon receiving an address byte that matches the XOFF2 character, the receiver will be automatically enabled if not already enabled, and the address character is pushed into the RX FIFO along with the parity bit (in place of the parity error bit). The receiver also generates an LSR interrupt. The receiver will then receive the subsequent data. If another address byte is received and this address does not match the programmed XOFF2 character, then the receiver will automatically be disabled and the address byte is ignored. If the address byte matches XOFF2, the receiver will put this byte in the RX FIFO along with the parity bit in the parity error bit.

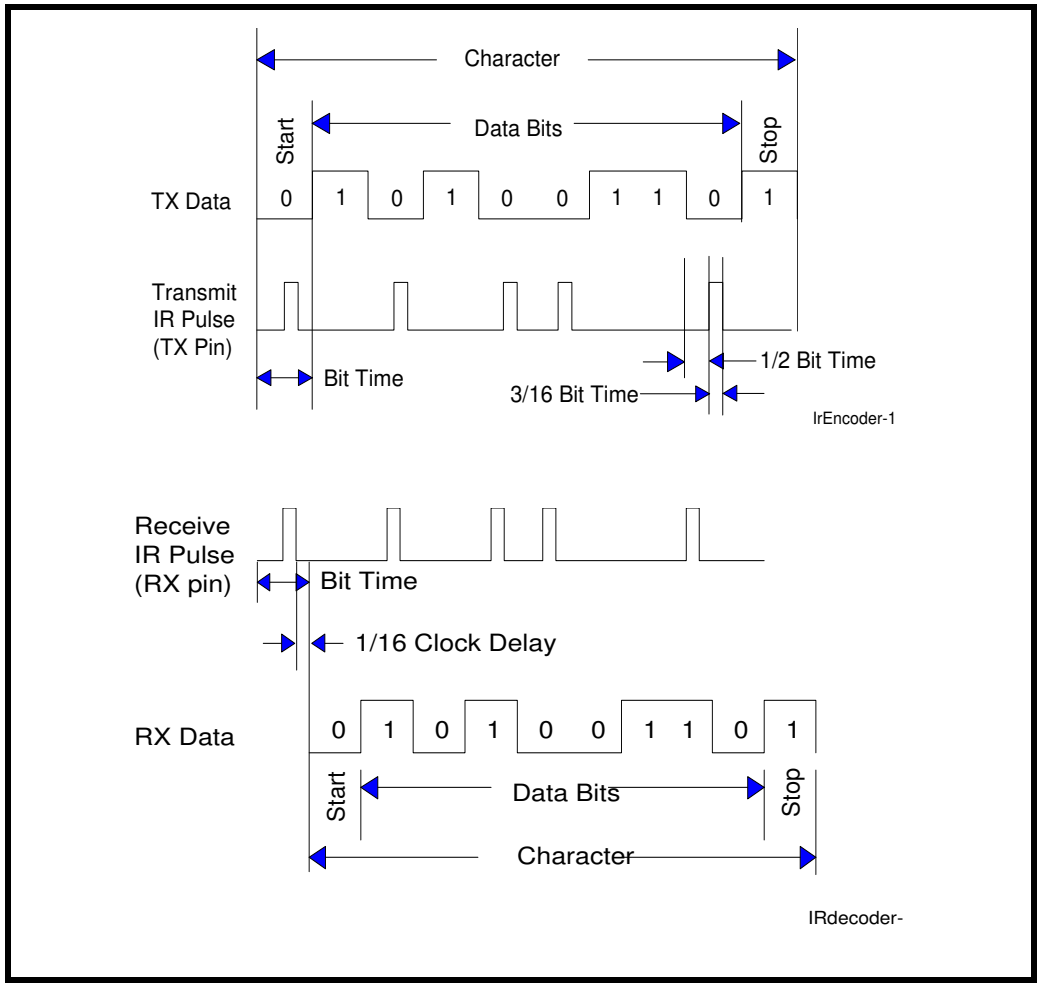
2.14 Infrared Mode

Each UART in the 698 includes the infrared encoder and decoder compatible to the IrDA (Infrared Data Association) version 1.0. The input pin ENIR conveniently activates all 8 UART channels to start up in the infrared mode. Note that the ENIR pin is sampled when the RST# input is de-asserted. This global control pin enables the MCR bit-6 function in every UART channel register. After power up or a reset, the software can overwrite MCR bit-6 if so desired. ENIR and MCR bit-6 also disable the receiver while the transmitter is sending data. This prevents echoed data from reaching the receiver. The global activation ENIR pin prevents the infrared emitter from turning on and drawing large amount of current while the system is starting up. When the infrared feature is enabled, the transmit data outputs, TX[7:0], would idle at logic zero level. Likewise, the RX [7:0] inputs assume an idle level of logic zero.

The infrared encoder sends out a 3/16 of a bit wide HIGH-pulse for each “0” bit in the transmit data stream. This signal encoding reduces the on-time of the infrared LED, hence reduces the power consumption. See **Figure 10** below.

The infrared decoder receives the input pulse from the infrared sensing diode on RX pin. Each time it senses a light pulse, it returns a logic zero to the data bit stream. The decoder also accepts (when FCTR bit-4 = 1) an inverted IR-encoded input signal. This option supports active LOW instead of normal active HIGH pulse from some infrared modules on the market.

FIGURE 10. INFRARED TRANSMIT DATA ENCODING AND RECEIVE DATA DECODING



2.15 Sleep Mode with Auto Wake-Up

The 698 supports low voltage system designs, hence, a sleep mode is included to reduce its power consumption when the chip is not actively used.

All of these conditions must be satisfied for the 698 to enter sleep mode:

- no interrupts pending for all 8 channels of the 698 (ISR bit-0 = 1)
- SLEEP register = 0xFF
- modem inputs are not toggling (MSR bits 0-3 = 0)
- RX input pin of all 8 channels are idling HIGH

The 698 stops its crystal oscillator to conserve power in the sleep mode. User can check the XTAL2 pin for no clock output as an indication that the device has entered the sleep mode.

The 698 resumes normal operation by any of the following:

- a receive data start bit transition (HIGH to LOW)
- a data byte is loaded to the transmitter, THR or FIFO
- a change of logic state on any of the modem or general purpose serial inputs: CTS#, DSR#, CD#, RI#

If the 698 is awakened by any one of the above conditions, it will generate an interrupt. If the interrupt for the event that woke up the 698 is not enabled, then a special wake-up interrupt occurs where reading the interrupt status register will return a "no interrupt" indication. For example, there is a change of state on the CTS# input that wakes up the 698, but the MSR interrupt is not enabled. Reading the interrupt status register will return a value indicating that there are no pending interrupts and will clear the wake-up interrupt.

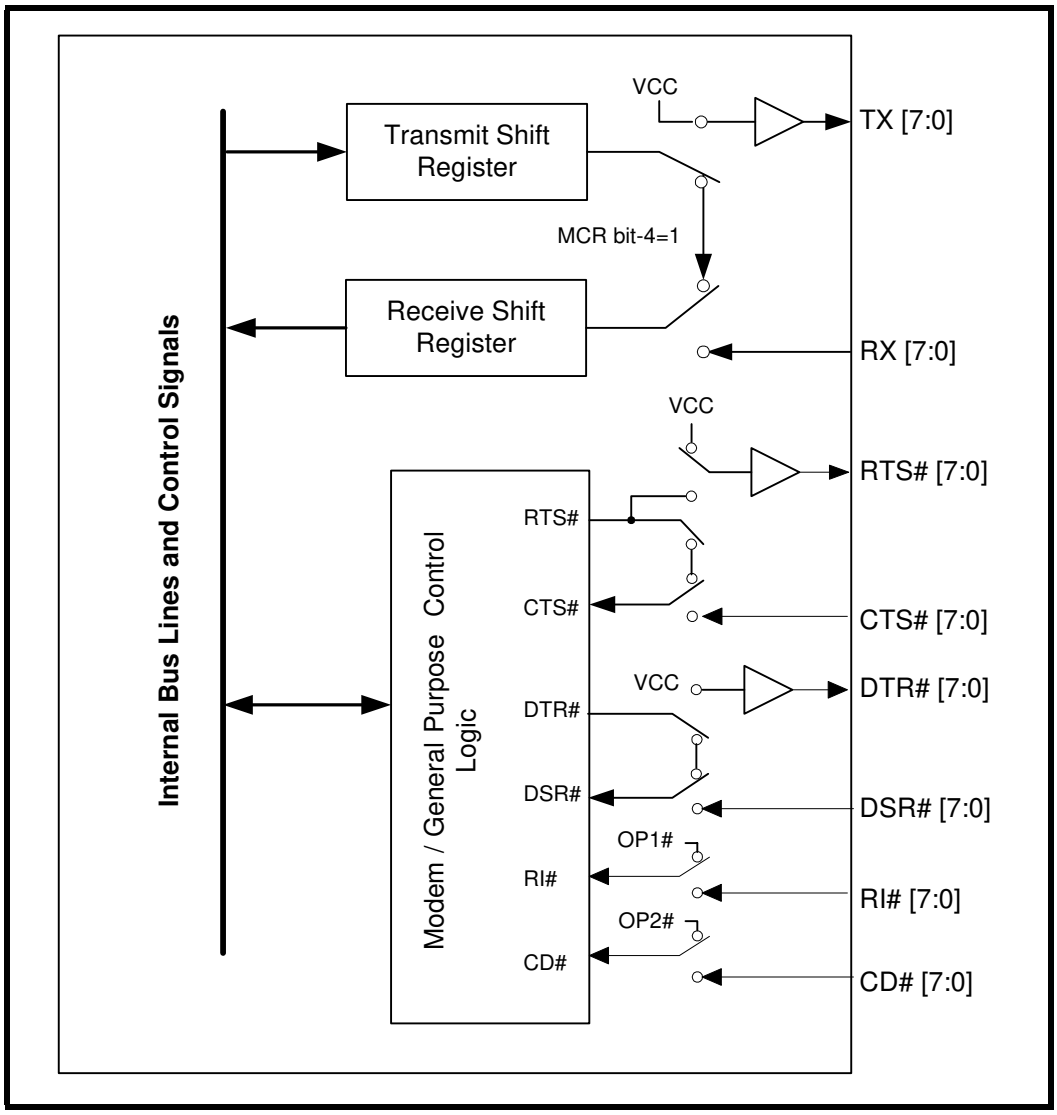
The 698 will return to the sleep mode automatically after all interrupting conditions have been serviced and cleared. If the 698 is awakened by the modem inputs, a read to the MSR is required to reset the modem inputs. In any case, the sleep mode will not be entered while an interrupt is pending in any channel. The 698 will stay in the sleep mode of operation until it is disabled by setting SLEEP = 0x00.

A word of caution: owing to the starting up delay of the crystal oscillator after waking up from sleep mode, the first few receive characters may be lost. The number of characters lost during the restart also depends on your operating data rate. More characters are lost when operating at higher data rate.

2.16 Internal Loopback

Each UART channel provides an internal loopback capability for system diagnostic purposes. The internal loopback mode is enabled by setting MCR register bit-4 to logic 1. All regular UART functions operate normally. Figure 11 shows how the modem port signals are re-configured. Transmit data from the transmit shift register output is internally routed to the receive shift register input allowing the system to receive the same data that it was sending. The TX pin is held at HIGH or mark condition while RTS# and DTR# are de-asserted (HIGH), and CTS#, DSR# CD# and RI# inputs are ignored.

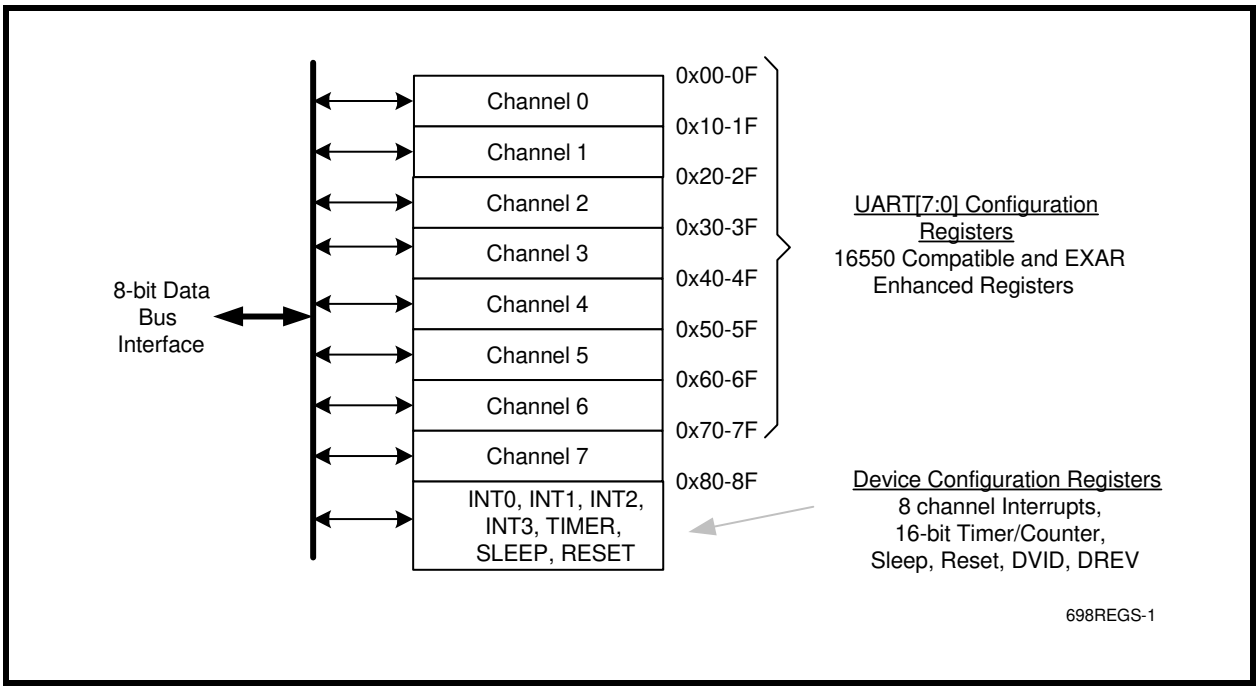
FIGURE 11. INTERNAL LOOP BACK



3.0 XR16V698 REGISTERS

The XR16V698 octal UART register set consists of the Device Configuration Registers that are accessible directly from the data bus for programming general operating conditions of the UARTs and monitoring the status of various functions. These functions include all 8 channel UART's interrupt control and status, 16-bit general purpose timer control and status, sleep mode, soft-reset, and device identification and revision. Also, each UART channel has its own set of internal UART Configuration Registers for its own operation control, status reporting and data transfer. These registers are mapped into a 256-byte of the data memory address space. The following paragraphs describe all the registers in detail.

FIGURE 12. THE XR16V698 REGISTERS



3.1 DEVICE CONFIGURATION REGISTER SET

The device configuration registers are directly accessible from the bus. This provides easy programming of general operating parameters to the 698 UART and for monitoring the status of various functions. The device configuration registers are mapped onto address 0x80-8F as shown on the register map in Table 8 and Figure 12. These registers provide global controls and status of all 8 channel UARTs that include interrupt status, 16-bit general purpose timer control and status, 4X or 8X or 16X sampling clock, sleep mode control, soft-reset control, simultaneous UART initialization, and device identification and revision.

TABLE 7: XR16V698 REGISTER SETS

ADDRESS [A7:A0]	UART CHANNEL SPACE	REFERENCE	COMMENT
0x00 - 0x0F	UART channel 0 Registers	(Table 11 & 12)	First 8 registers are 16550 compatible
0x10 - 0x1F	UART channel 1 Registers	(Table 11 & 12)	
0x20 - 0x2F	UART channel 2 Registers	(Table 11 & 12)	
0x30 - 0x3F	UART channel 3 Registers	(Table 11 & 12)	
0x40 - 0x4F	UART channel 4 Registers	(Table 11 & 12)	
0x50 - 0x5F	UART channel 5 Registers	(Table 11 & 12)	
0x60 - 0x6F	UART channel 6 Registers	(Table 11 & 12)	
0x70 - 0x7F	UART channel 7 Registers	(Table 11 & 12)	
0x80 - 0x8F	Device Configuration Registers	(Table 8)	Interrupt registers and global controls

TABLE 8: DEVICE CONFIGURATION REGISTERS

ADDRESS [A7:A0]	READ/ WRITE	REGISTER	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x80	R	INT Source	UART 7	UART 6	UART 5	UART 4	UART 3	UART 2	UART 1	UART 0
0x81	R	INT 1	UART 2 bit 1	source bit 0	UART 1 bit 2	interrupt bit 1	source bit 0	UART 0 bit 2	interrupt bit 1	source bit 0
0x82	R	INT 2	UART 5 bit 0	UART 4 bit 2	interrupt bit 1	source bit 0	UART 3 bit 2	interrupt bit 1	source bit 0	UART 2 bit 2
0x83	R	INT 3	UART 7 bit 2	interrupt bit 1	source bit 0	UART 6 bit 2	interrupt bit 1	source bit 0	UART 5 bit 2	source bit 1
0x84	R/W	TIMER CTRL	0	0	0	0	TimerCtrl bit-3	TimerCtrl bit-2	TimerCtrl bit-1	TimerCtrl bit-0
0x85	R	TIMER	0	0	0	0	0	0	0	0
0x86	R/W	TIMER LSB	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0x87	R/W	TIMER MSB	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0x88	R/W	8X MODE	UART 7	UART 6	UART 5	UART 4	UART 3	UART 2	UART 1	UART 0
0x89	R/W	4X MODE	UART 7	UART 6	UART 5	UART 4	UART 3	UART 2	UART 1	UART 0
0x8A	W	RESET	Reset UART 7	Reset UART 6	Reset UART 5	Reset UART 4	Reset UART 3	Reset UART 2	Reset UART 1	Reset UART 0
0x8B	R/W	SLEEP	Enable sleep UART 7	Enable sleep UART 6	Enable sleep UART 5	Enable sleep UART 4	Enable sleep UART 3	Enable sleep UART 2	Enable sleep UART 1	Enable sleep UART 0
0x8C	R	DREV	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0x8D	R	DVID	0	1	1	0	1	0	0	0
0x8E	R/W	REGB	0	0	0	0	0	0	0	write to all UARTs

3.1.1 The Global Interrupt Source Registers

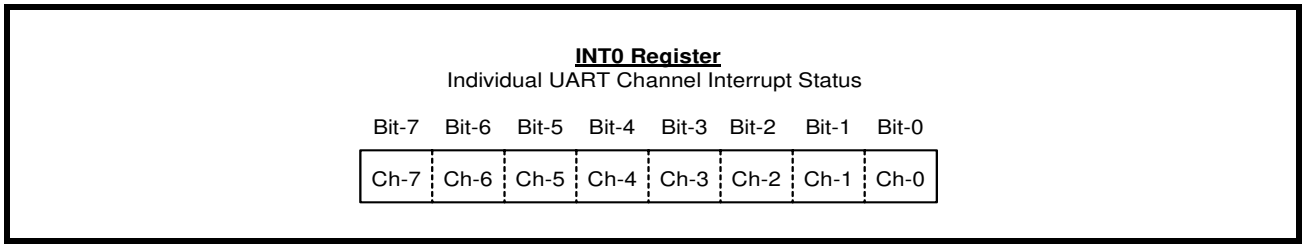
The XR16V698 has a global interrupt source register set that consists of 4 consecutive registers [INT0, INT1, INT2 and INT3]. The four registers are in the device configuration register address space.

INT3 [0x00]	INT2 [0x00]	INT1 [0x00]	INT0 [0x00]

All four registers default to logic zero (as indicated in square braces) for no interrupt pending. All 8 channel interrupts are enabled or disabled in each channel's IER register. INT0 shows individual status for each channel while INT1, INT2 and INT3 show the details of the source of each channel's interrupt with its unique 3-bit encoding. **Figure 13** shows the 4 interrupt registers in sequence for clarity. The 16-bit timer and sleep wake-up interrupts are masked in the device configuration registers, TIMERCNTL and SLEEP. An interrupt is generated (if enabled) by the 698 when awakened from sleep if all 8 channels were placed in the sleep mode previously.

Each bit gives an indication of the channel that has requested for service. For example, bit-0 represents channel 0 and bit-7 indicates channel 7. Logic one indicates the channel N [7:0] has called for service. The interrupt bit clears after reading the appropriate register of the interrupting UART channel register (ISR, LSR and MSR). SEE"INTERRUPT CLEARING:" ON PAGE 34. for interrupt clearing details.

3.1.1.1 INT0 Channel Interrupt Indicator



3.1.1.2 INT1, INT2 and INT3 Interrupt Source Locator

INT3, INT2 and INT1 provide a 24-bit (3 bits per channel) encoded interrupt indicator. **Table 9** shows the 3 bit encoding and their priority order. The 16-bit Timer time-out interrupt will show up only as a channel 0 interrupt. For other channels, interrupt 7 is reserved.

FIGURE 13. THE GLOBAL INTERRUPT REGISTERS, INT0, INT1, INT2 AND INT3