



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



**High Performance 1.62V To 3.6V Dual Uart with 64-Byte FIFO****Features**

- 1.62V to 3.6V with 5V Tolerant Serial Inputs
- Programmable Sleep Mode with automatic wake-up
  - Intel or Motorola Data Bus Interface select
- Each UART is independently controlled with:
  - 16C550 Compatible Register Set
  - 64-byte Transmit and Receive FIFOs
  - Transmit and Receive FIFO Level Counters
  - Programmable TX and RX FIFO Trigger Levels for DMA and Interrupt Generation
    - Programmable Receive FIFO Trigger Levels for Software/Hardware Flow Control
    - Programmable hysteresis(Table A-D) for Software/Hardware Flow Control
  - Automatic RTS/CTS Flow Control
  - Automatic Xon/Xoff Software Flow Control with Optional Data Flow Resume by Xon Any Character
    - DMA Signaling Capability for both Received and Transmitted Data
  - RS485 HDX Control Output
  - RS485 auto address detection
  - Infrared (IrDA 1.0/1.1) Data Encoder/Decoder
  - Programmable Data Rate with Prescaler
- Up to 16 Mbps Serial Data Rate with 64MHz external clock input
- Crystal oscillator(up to 24MHz) or external clock(up to 80MHz) input
- Built in Power-On-Reset circuit

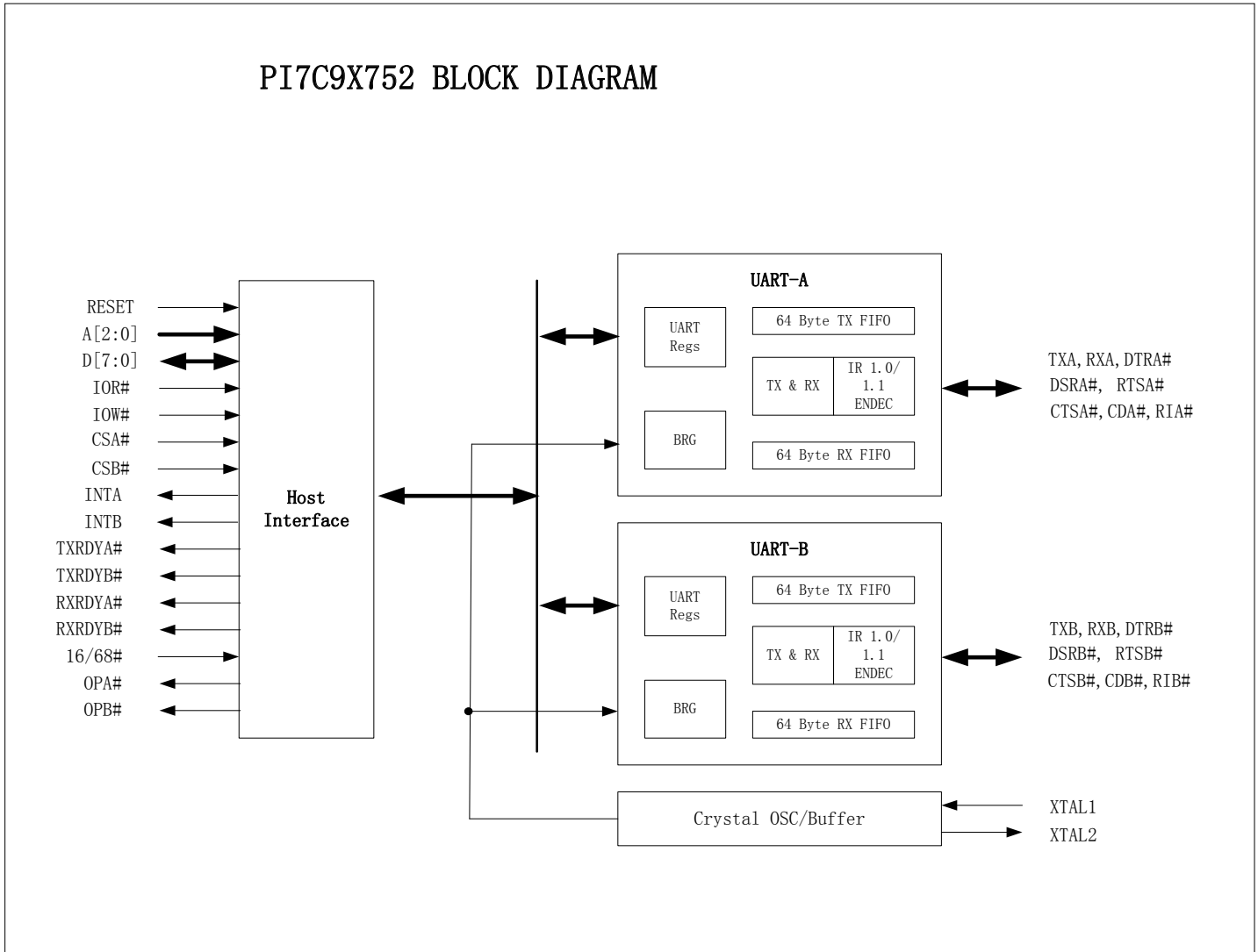
**Application**

- 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

**Description**

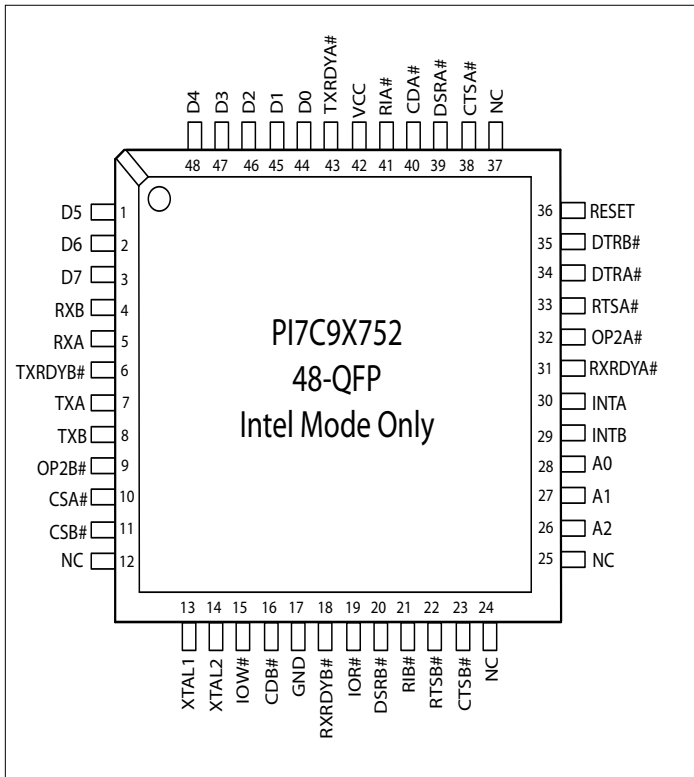
The PI7C9X752 (752), is a 1.62V to 3.6V dual Universal Asynchronous Receiver and Transmitter (UART) with 5V tolerant serial (modem) inputs. The highly integrated device is designed for high bandwidth requirement in communication systems. Each UART has its own 16C550 compatible set of configuration registers, TX and RX FIFOs of 64 bytes, fully programmable transmit and receive FIFO trigger levels, TX and RX FIFO level counters, automatic RTS/CTS hardware flow control with programmable hysteresis, automatic software (Xon/Xoff) flow control, RS-485 half-duplex direction control, Intel or Motorola bus interface and sleep mode for power saving.

**Block Diagram**

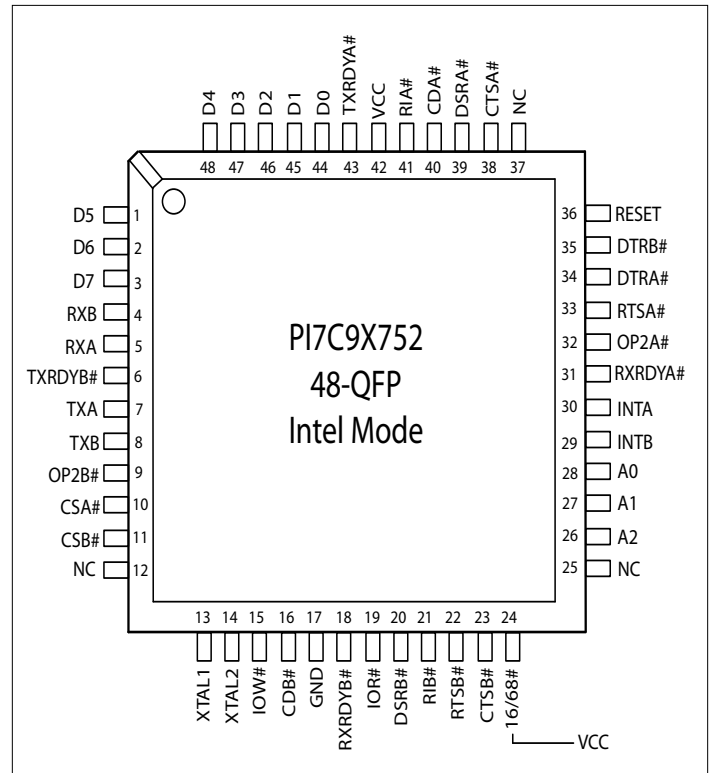


**PI7C9X752**

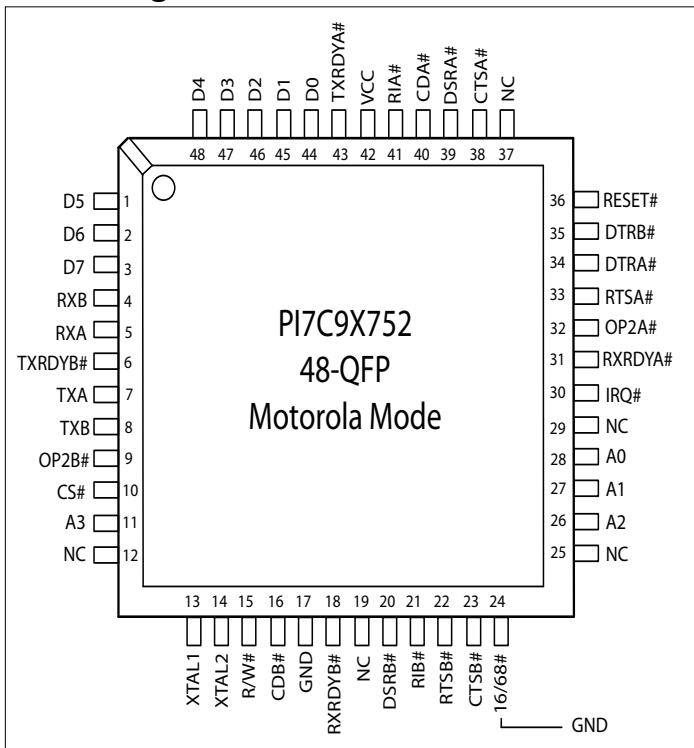
**Pin Configuration 48-QFP**



**Pin Configuration 48-QFP**



**Pin Configuration 48-QFP**



## Pin Description

Pin Name	48-TQFP Pin #	Type	Description
<b>Data Bus Interface</b>			
A2	26	I	Address data lines [2:0]. These 3 address lines select one of the internal registers in UART channel A/B during a data bus transaction.
A1	27		
A0	28		
D7	3	I/O	Data bus lines [7:0] (bidirectional).
D6	2		
D5	1		
D4	48		
D3	47		
D2	46		
D1	45		
D0	44		
IOR# (NC)	19	I	When 16/68# pin is HIGH, the Intel bus interface is selected and this input becomes 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 [A2:A0], puts the data byte on the data bus to allow the host processor to read it on the rising edge. When 16/68# pin is LOW, the Motorola bus interface is selected and this input is not used.
IOW# (R/W#)	15	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 rising edge transfers the data byte on the data bus to an internal register pointed by the address lines. When 16/68# pin is LOW, the Motorola bus interface is selected and this input becomes read (HIGH) and write (LOW) signal.
CSA#(CS#)	10	I	When 16/68# pin is HIGH, this input is chip select A (active low) to enable channel A in the device. When 16/68# pin is LOW, this input becomes the chip select (active low) for the Motorola bus interface.
CSB#(A3)	11	I	When 16/68# pin is HIGH, this input is chip select B (active low) to enable channel B in the device. When 16/68# pin is LOW, this input becomes address line A3 which is used for channel selection in the Motorola bus interface. Input logic 0 selects channel A and logic 1 selects channel B.
INTA (IRQ#)	30	O	When 16/68# pin is HIGH for Intel bus interface, this output becomes channel A interrupt output. The output state is defined by the user through the software setting of MCR[3]. INTA is set to the active mode and OP2A# output LOW when MCR[3] is set to a logic 1. INTA is set to the three state mode and OP2A# to HIGH when MCR[3] is set to a logic 0. See MCR[3]. When 16/68# pin is LOW for Motorola bus interface, this output becomes device interrupt output (active low, open drain). An external pull-up resistor is required for proper operation.

Pin Name	48-TQFP Pin #	Type	Description
INTB (NC)	29	O	When 16/68# pin is HIGH for Intel bus interface, this output becomes channel B interrupt output. The output state is defined by the user through the software setting of MCR[3]. INTB is set to the active mode and OP2A# output to LOW when MCR[3] is set to a logic 1. INTA is set to the three state mode and OP2A# to HIGH when MCR[3] is set to a logic 0. See MCR[3]. When 16/68# pin is LOW for Motorola bus interface, this output is not used.
TXRDYA#	43	O	UART channel A Transmitter Ready (active low). The output provides the TX FIFO/THR status for transmit channel A. See Table 3. If it is not used, leave it unconnected.
RXRDYA#	31	O	UART channel A Receiver Ready (active low). This output provides the RX FIFO/RHR status for receive channel A. See Table 3. If it is not used, leave it unconnected.
TXRDYB#	6	O	UART channel B Transmitter Ready (active low). The output provides the TX FIFO/THR status for transmit channel B. See Table 4. If it is not used, leave it unconnected.
RXRDYB#	18	O	UART channel B Receiver Ready (active low). This output provides the RX FIFO/RHR status for receive channel B. See Table 3. If it is not used, leave it unconnected.
<b>Modem Or Serial I/O Interface</b>			
TXA	7	O	UART channel A Transmit Data or infrared encoder data. Standard transmit and receive interface is enabled when MCR[6] = 0. In this mode, the TX signal will be HIGH during reset or idle (no data). Infrared IrDA transmit and receive interface is enabled when MCR[6] = 1. In the Infrared mode, the inactive state (no data) for the Infrared encoder/decoder interface is LOW. If it is not used, leave it unconnected.
RXA	5	I	UART channel A Receive Data or infrared receive data. Normal receive data input must idle HIGH. The infrared receiver pulses typically idles at LOW but can be inverted by software control prior going in to the decoder, see MCR[6]. If this pin is not used, tie it to VCC or pull it high via a 100k ohm resistor.
RTSA#	33	O	UART channel A Request-to-Send (active low) or general purpose output. This output must be asserted prior to using auto RTS flow control, see EFR[6] and IER[6]. For auto RS485 half-duplex direction control, see DLD[6].
CTSA#	38	I	UART channel A Clear-to-Send (active low) or general purpose input. It can be used for auto CTS flow control, see EFR[7] and IER[7]. This input should be connected to VCC or GND when not used.
DTRA#	34	O	UART channel A Data-Terminal-Ready (active low) or general purpose output. If it is not used, leave it unconnected.
DSRA#	39	I	UART channel A Data-Set-Ready (active low) or general purpose input. This input should be connected to VCC or GND when not used.
CDA#	40	I	UART channel A Carrier-Detect (active low) or general purpose input. This input should be connected to VCC or GND when not used.
RIA#	41	I	UART channel A Ring-Indicator (active low) or general purpose input. This input should be connected to VCC or GND when not used.

Pin Name	48-TQFP Pin #	Type	Description
OP2A#	32	O	Output Port 2 Channel A - The output state is defined by the user and through the software setting of MCR[3]. INTA is set to the active mode and OP2A# output LOW when MCR[3] is set to a logic 1. INTA is set to the three state mode and OP2A# output HIGH when MCR[3] is set to a logic 0. See MCR[3]. If INTA is used, this output should not be used as a general output else it will disturb the INTA output functionality.
TXB	8	O	UART channel B Transmit Data or infrared encoder data. Standard transmit and receive interface is enabled when MCR[6] = 0. In this mode, the TX signal will be HIGH during reset or idle (no data). Infrared IrDA transmit and receive interface is enabled when MCR[6] = 1. In the Infrared mode, the inactive state (no data) for the Infrared encoder/decoder interface is LOW. If it is not used, leave it unconnected.
RXB	4	I	UART channel B Receive Data or infrared receive data. Normal receive data input must idle HIGH. The infrared receiver pulses typically idles at logic 0 but can be inverted by software control prior going in to the decoder, see MCR[6]. If this pin is not used, tie it to VCC or pull it high via a 100k ohm resistor.
RTSB#	22	O	UART channel B Request-to-Send (active low) or general purpose output. This port must be asserted prior to using auto RTS flow control, see EFR[6] and IER[6]. For auto RS485 half-duplex direction control, see DLD[6].
CTSB#	23	I	UART channel B Clear-to-Send (active low) or general purpose input. It can be used for auto CTS flow control, see EFR[7] and IER[7]. This input should be connected to VCC or GND when not used.
DTRB#	35	O	UART channel B Data-Terminal-Ready (active low) or general purpose output. If it is not used, leave it unconnected.
DSRB#	20	I	UART channel B Data-Set-Ready (active low) or general purpose input. This input should be connected to VCC or GND when not used.
CDB#	16	I	UART channel B Carrier-Detect (active low) or general purpose input. This input should be connected to VCC or GND when not used.
RIB#	21	I	UART channel B Ring-Indicator (active low) or general purpose input. This input should be connected to VCC or GND when not used.
OP2B#	9	O	Output Port 2 Channel B - The output state is defined by the user and through the software setting of MCR[3]. INTB is set to the active mode and OP2B# output LOW when MCR[3] is set to a logic 1. INTB is set to the three state mode and OP2B# output HIGH when MCR[3] is set to a logic 0. See MCR[3]. If INTB is used, this output should not be used as a general output else it will disturb the INTB output functionality.
<b>Ancillary Signals</b>			
XTAL1	13	I	Crystal or external clock input.
XTAL2	14	O	Crystal or buffered clock output.
16/68#	24	I	Intel or Motorola Bus Select (internal pull-up resistor). When 16/68# pin is HIGH, 16 or Intel Mode, the device will operate in the Intel bus type of interface. When 16/68# pin is LOW, 68 or Motorola mode, the device will operate in the Motorola bus type of interface.

**PI7C9X752**

Pin Name	48-TQFP	Type	Description
	Pin #		
RESET (RESET#)	36	I	When 16/68# pin is HIGH for Intel bus interface, this input becomes RESET (active high). When 16/68# pin is LOW for Motorola bus interface, this input becomes RESET# (active low). A 40 ns minimum active pulse on this pin will reset the internal registers and all outputs of channel A and B. The UART transmitter output will be held HIGH, the receiver input will be ignored and outputs are reset during reset period (see Table 16).
VCC	42	Pwr	1.62V to 3.63V power supply.
GND	17	Pwr	Power supply common, ground.
N.C.	12, 25, 37		No Connection.

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



## Functional Description

The PI7C9X752 integrates the functions of 4 enhanced 16550 UARTs. 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 64-byte of transmit and receive FIFOs, automatic RTS/CTS hardware flow control with hysteresis control, automatic Xon/Xoff and special character software flow control, programmable transmit and receive FIFO trigger levels, FIFO level counters, infrared encoder and decoder (IrDA ver. 1.0 and 1.1), programmable baud rate generator with a prescaler of divide by 1 or 4, and data rate up to 16Mbps with 4X sampling clock. The PI7C9X752 is a 1.62-3.6V device with 5 volt tolerant inputs (except XTAL1).

The PI7C9X752 comes with packages: 48-pin QFP which both Intel and Motorola interfere.

## 1. Trigger levels

The PI7C9X752 provides independent selectable and programmable trigger levels for both Receiver and transmitter DMA and interrupt generation. After reset, both transmitter and receiver FIFOs are disabled and so, in effect, the trigger level is the default value of one character. The selectable trigger levels are controlled via Register TLR(if TLR is non-zero) or (if TLR is zero) via FIFO Control Register(FCR) and Feature Control Register(FCTR). Refer to Table 1.

**Table 1. Transmit and Receive FIFO Trigger Table and Level Selection**

Trigger	FCTR BIT-5	FCTR BIT-4	FCR BIT-7	FCR BIT-6	FCR BIT-5	FCR BIT-4	Receive Trigger Level	Transmit Trigger Level
Table-A	0	0	0	0	0	0	1(Default)	1(Default)
			0	1			4	
			1	0			8	
			1	1			14	
Table-B	0	1			0	0		16
					0	1		8
					1	0		24
					1	1		30
			0	0	1	1	8	
			0	1	1	1	16	
Table-C	1	0			0	0		8
					0	1		16
					1	0		32
					1	1		56
			0	0	1	1	8	
			0	1	1	1	16	
Table-D	1	1	X	X	X	X	Programmable via RX-TRG register	Programmable via TX-TRG register
			X	X	X	X		
			X	X	X	X		
			X	X	X	X		

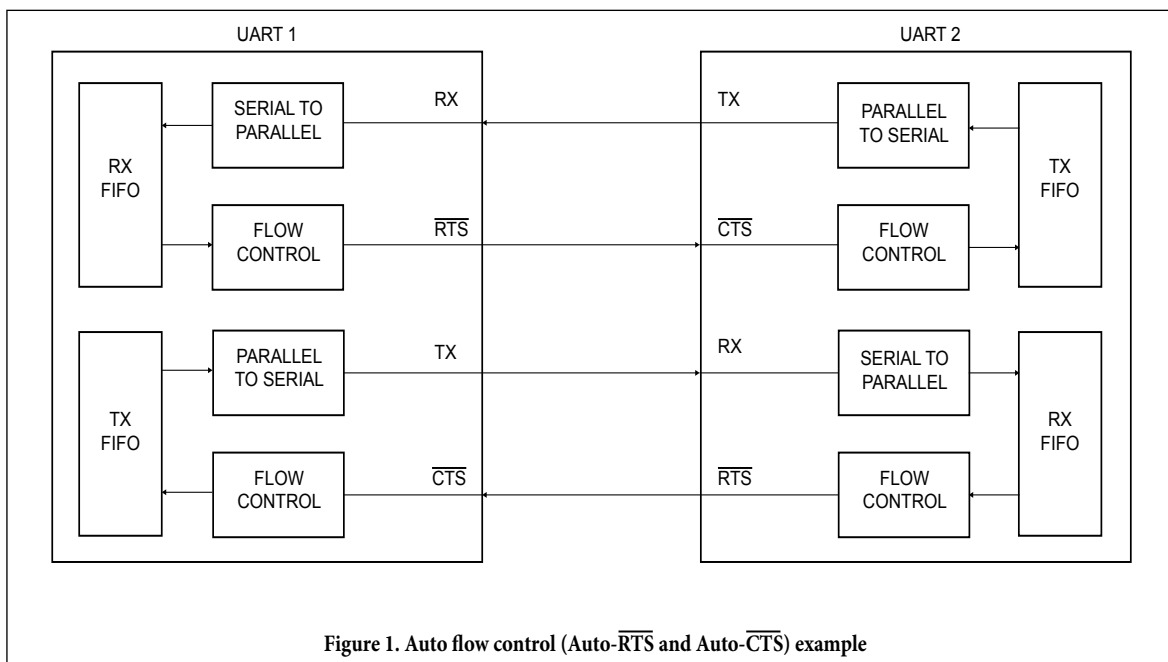
## 2. Hardware flow control

Hardware flow control is comprised of Auto- $\overline{\text{CTS}}$  and Auto-RTS (see Figure 1). Auto- $\overline{\text{CTS}}$  and Auto- $\overline{\text{RTS}}$  can be enabled/disabled independently by programming EFR[7:6].

With Auto- $\overline{\text{CTS}}$ ,  $\overline{\text{CTS}}$  must be active before the UART can transmit data.

Auto- $\overline{\text{RTS}}$  only activates the  $\overline{\text{RTS}}$  output when there is enough room in the FIFO to receive data and de-activates the  $\overline{\text{RTS}}$  output when the RX FIFO is sufficiently full. The halt and resume trigger levels is controlled by FCR and FCTR bits.

If both Auto- $\overline{\text{CTS}}$  and Auto- $\overline{\text{RTS}}$  are enabled, when  $\overline{\text{RTS}}$  is connected to  $\overline{\text{CTS}}$ , data transmission does not occur unless the receiver FIFO has empty space. Thus, overrun errors are eliminated during hardware flow control. If not enabled, overrun errors occur if the transmit data rate exceeds the receive FIFO servicing latency.



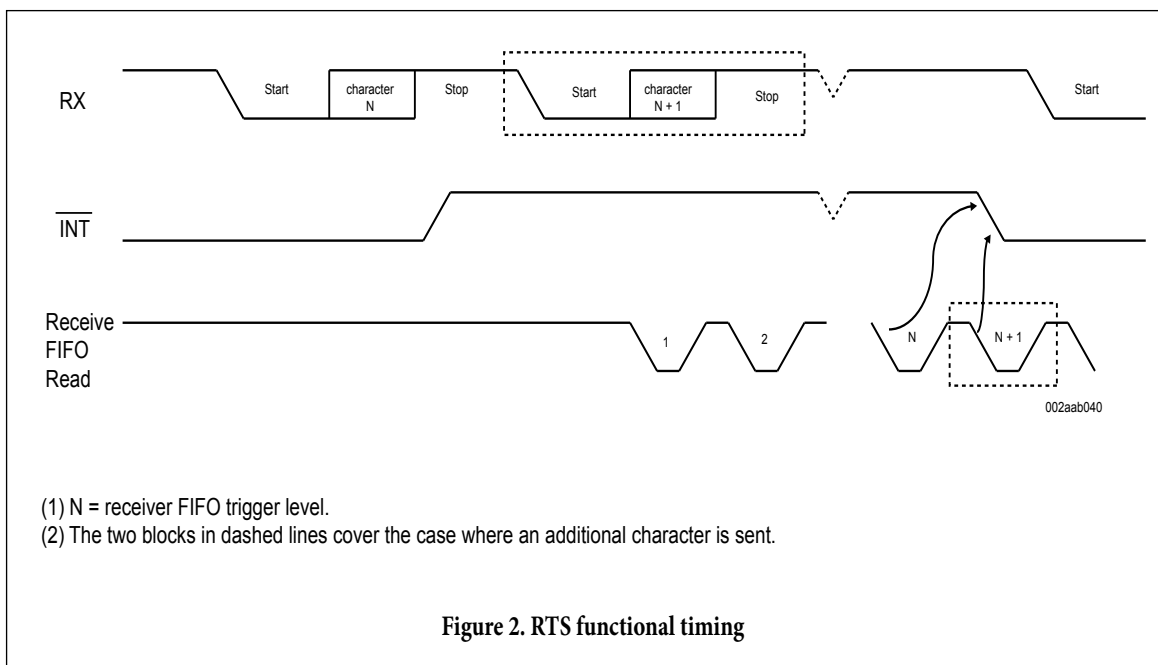
## 2.1 Auto RTS Hardware Flow Control Operation

Figure 2 shows RTS# functional timing. The RTS# 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:

- Enable auto RTS flow control using EFR bit-6.
- The auto RTS function must be started by asserting the RTS# output pin (MCR bit-1 to a logic 1) after it is enabled. With the Auto RTS function enabled, the RTS# output pin will be de-asserted (HIGH) when the FIFO reaches the halt level. The halting level is setting by TCR[3:0](if non-zero) or is the next trigger level for Trigger Tables A-C (See Table 1), or is the RX trigger level plus the hysteresis level for Trigger Table D.

The RTS# output pin will be asserted (LOW) again after the FIFO is unloaded to programmed resume level. The resume level is setting by TCR[7:4](if non-zero), or is the next trigger level below the programmed trigger level for Trigger Table A-C, or is the RX trigger level minus the hysteresis level if Table D is selected. However, even under these conditions, the 754 will continue to accept data until the receive FIFO is full if the remote UART transmitter continues to send data.

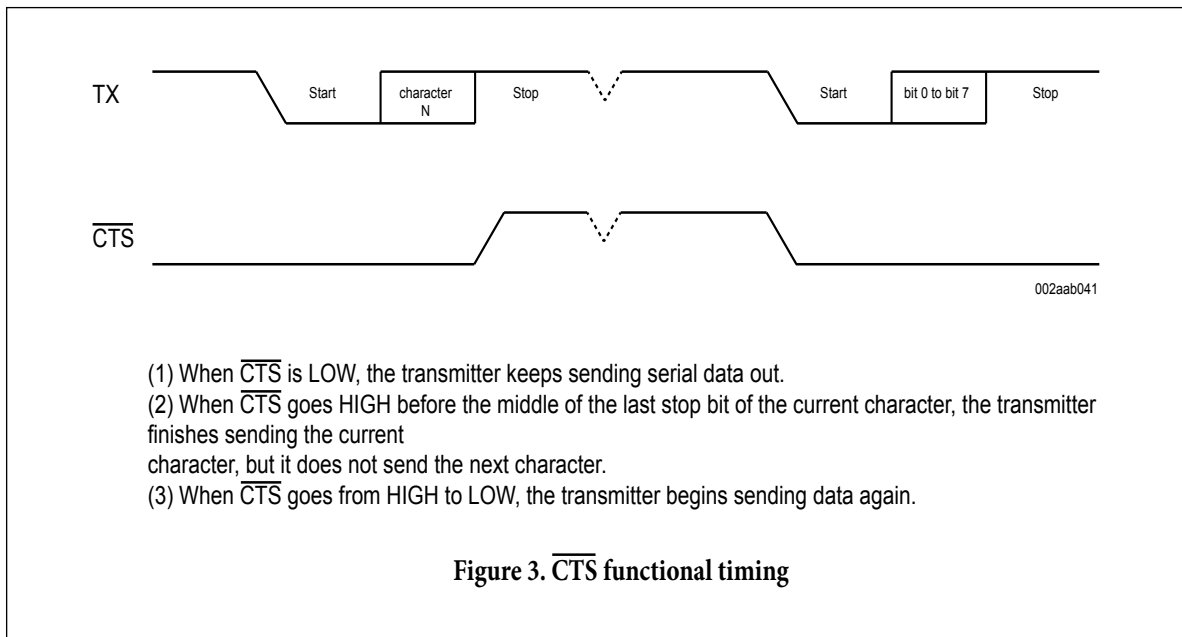
- If used, enable RTS interrupt through IER bit-6 (after setting EFR bit-4). The UART issues an interrupt when the RTS# pin makes a transition: ISR bit-5 will be set to 1.



## 2.2 Auto CTS Flow Control

The CTS pin is monitored to suspend/restart local transmitter. The flow control features are individually selected to fit specific application requirement:

- Enable auto CTS flow control using EFR bit-7. With the Auto CTS 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# input is re-asserted (LOW), indicating more data may be sent.
- If used, enable CTS interrupt through IER bit-7 (after setting EFR bit-4). The UART issues an interrupt when the CTS# 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# input returns LOW, indicating more data may be sent.



### 3 Software flow control

Software flow control is enabled through the Enhanced Features Register and the Modem Control Register. Different combinations of software flow control can be enabled by setting different combinations of EFR[3:0]. Table 1 shows software flow control options.

**Table 2. Software flow control options (EFR[3:0])**

EFR[3]	EFR[2]	EFR[1]	EFR[0]	TX, RX software flow control
0	0	x	x	no transmit flow control
1	0	x	x	transmit Xon1, Xoff1
0	1	x	x	transmit Xon2, Xoff2
1	1	x	x	transmit Xon1 and Xon2, Xoff1 and Xoff2
x	x	0	0	no receive flow control
x	x	1	0	receiver compares Xon1, Xoff1
x	x	0	1	receiver compares Xon2, Xoff2
1	0	1	1	transmit Xon1, Xoff1 receiver compares Xon1 or Xon2, Xoff1 or Xoff2
0	1	1	1	transmit Xon2, Xoff2 receiver compares Xon1 or Xon2, Xoff1 or Xoff2
1	1	1	1	transmit Xon1 and Xon2, Xoff1 and Xoff2 receiver compares Xon1 and Xon2, Xoff1 and Xoff2

There are two other enhanced features relating to software flow control:

- Xon Any function (MCR[5]): Receiving any character will resume operation after recognizing the Xoff character. It is possible that an Xon1 character is recognized as an Xon Any character, which could cause an Xon2 character to be written to the RX FIFO.

- Special character (EFR[5]): Incoming data is compared to Xoff2. Detection of the special character sets the Xoff interrupt (ISR[4]) but does not halt transmission. The Xoff interrupt is cleared by a read of the Interrupt Status Register (ISR). The special character is transferred to the RX FIFO.

### 3.1 Receive flow control

When software flow control operation is enabled, UART will compare incoming data with Xoff1/Xoff2 programmed characters (in certain cases, Xoff1 and Xoff2 must be received sequentially). When the correct Xoff characters are received, transmission is halted after completing transmission of the current character. Xoff detection also sets ISR[4] (if enabled via IER[5]) and causes IRCE# to go LOW.

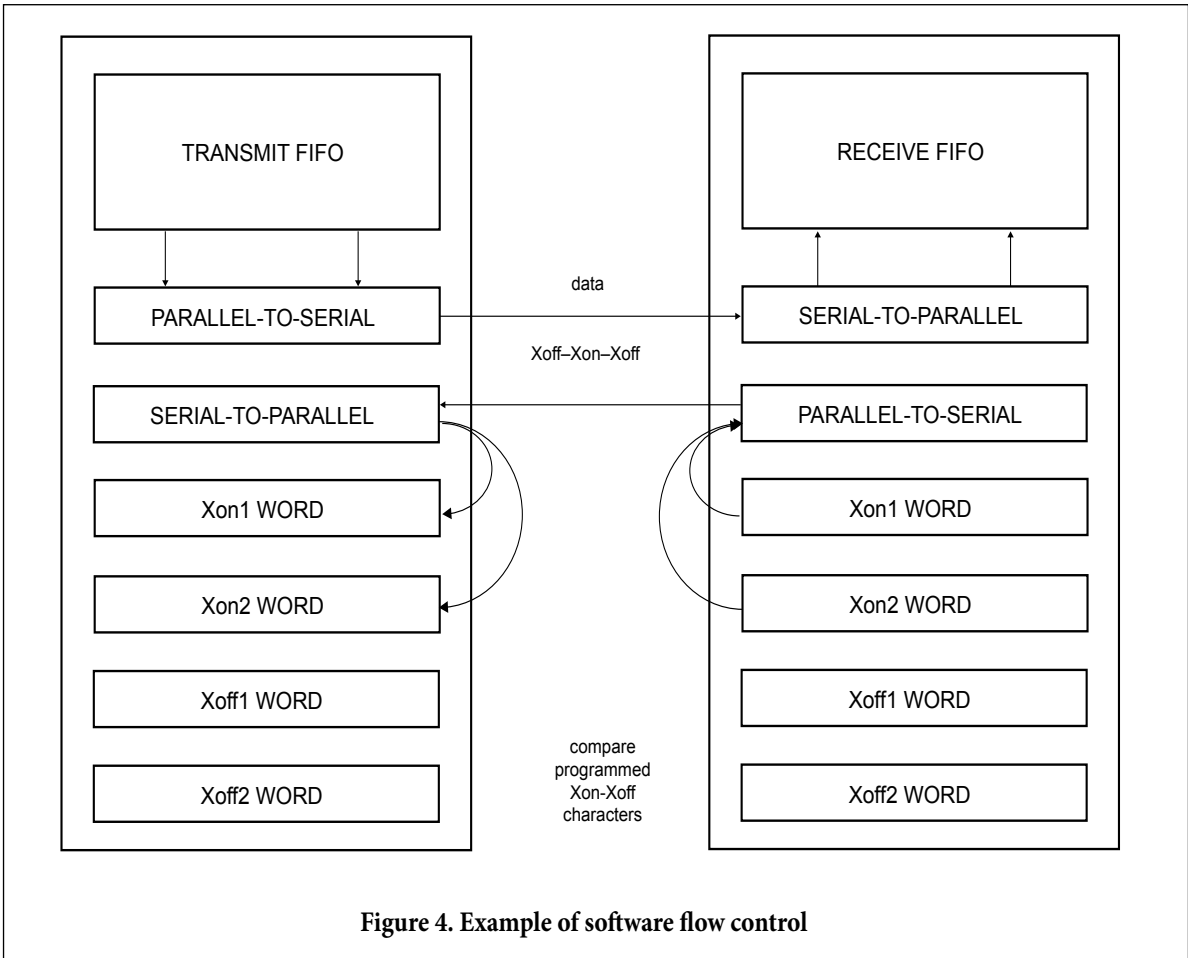
To resume transmission, an Xon1/Xon2 character must be received (in certain cases Xon1 and Xon2 must be received sequentially). When the correct Xon characters are received, ISR[4] is cleared, and the Xoff interrupt disappears.

### 3.2 Transmit flow control

Xoff1/Xoff2 character is transmitted after the receive FIFO crosses the programmed halting level in TCR[3:0] (if non-zero) or programmed receiver trigger level (for all trigger tables A-D). Xon1/Xon2 character is transmitted as soon as receive FIFO is below the programmed resuming level in TCR[7:4] (if non-zero) or less than one trigger level below the programmed receiver trigger level (for Trigger Tables A, B, and C) or when receive FIFO is less than the trigger level minus the hysteresis value (for Trigger Table D). This hysteresis value is the same as the Auto RTS Hysteresis value in Table 3.

**Table 3. SELECTABLE HYSTERESIS LEVELS WHEN TRIGGER TABLE-D IS SELECTED**

FCTR BIT-3	FCTR BIT-2	FCTR BIT-1	FCTR BIT-0	RTS Hysteresis (Characters)
0	0	0	0	0
0	0	0	1	+/- 4
0	0	1	0	+/- 6
0	0	1	1	+/- 8
0	1	0	0	+/- 8
0	1	0	1	+/- 16
0	1	1	0	+/- 24
0	1	1	1	+/- 32
1	1	0	0	+/- 12
1	1	0	1	+/- 20
1	1	1	0	+/- 28
1	1	1	1	+/- 36
1	0	0	0	+/- 40
1	0	0	1	+/- 44
1	0	1	0	+/- 48
1	0	1	1	+/- 52



#### 4. Hardware Reset, Power-On Reset (POR) and Software Reset

These three reset methods are identical and will reset the internal registers as indicated in Table 4.

Table 4 summarizes the state of register after reset.

**Table 4. UART Reset Conditions**

Register	Reset state
DLL	Bits 7-0 = 0x01
DLM	Bits 7-0 = 0x00
DLD	Bits 7-0 = 0x00
RHR	Bits 7-0 = 0xXX
THR	Bits 7-0 = 0xXX
IER	Bits 7-0 = 0x00
FCR	Bits 7-0 = 0x00
ISR	Bits 7-0 = 0x01
LCR	Bits 7-0 = 0x1D
MCR	Bits 7-0 = 0x00
LSR	Bits 7-0 = 0x60
MSR	Bits 3-0 = logic 0 Bits 7-4 = logic level of the inputs
SPR	Bits 7-0 = 0xFF
FCTR	Bits 7-0 = 0x20
EFR	Bits 7-0 = 0x00
TRG	Bits 7-0 = 0x00
FC	Bits 7-0 = 0x00
TCR	Bits 7-0 = 0x00
TLR	Bits 7-0 = 0x00
FSRDY	Bits 7-0 = 0x0F
XON1	Bits 7-0 = 0x00
XON2	Bits 7-0 = 0x00
XOFF1	Bits 7-0 = 0x00
XOFF2	Bits 7-0 = 0x00

Remark: Registers DLL, DLH, APR, XON1, XON2, XOFF1, XOFF2 are not reset by the top-level reset signal RESET, Software Reset, that is, they hold their initialization values during reset.

**Table 5. Output signals after reset**

Signal	Reset state
TX	HIGH
$\overline{\text{RTS/DTR}}$	HIGH
IRTX	LOW
RXRDY#	HIGH
TXRDY#	LOW
INT	Hi-Z (INTSEL=LOW) or LOW (INTSEL=HIGH)
IRQ#	Hi-Z (INTSEL=LOW)
OP2#	HIGH

## 5. Interrupts

The UART has interrupt generation and prioritization (seven prioritized levels of interrupts) capability. The interrupt enable registers (IER and IOIntEna) enable each of the seven types of interrupts and the INT signal in response to an interrupt generation. When an interrupt is generated, the ISR indicates that an interrupt is pending and provides the type of interrupt through ISR[5:0]. Table 4 summarizes the interrupt control functions.

**Table 6. Interrupt Source and Priority Level**

ISR[5:0]	Priority level	Interrupt type	Interrupt source
00 0001	none	none	None
00 0110	1	receiver line status	Overrun Error (OE), Framing Error (FE), Parity Error (PE), or Break Interrupt (BI) errors occur in characters in the RX FIFO
00 1100	3	RX time-out	Stale data in RX FIFO
00 0100	2	RHR interrupt	Receive data ready (FIFO disable) or RX FIFO above trigger level (FIFO enable)
00 0010	4	THR interrupt	Transmit FIFO empty (FIFO disable) or TX FIFO passes above trigger level (FIFO enable)
00 0000	5	modem status	Change of state of modem input pins
01 0000	6	Xoff interrupt	Receive Xoff character(s)/special character
10 0000	7	$\overline{\text{CTS}}$ , $\overline{\text{RTS}}$	$\overline{\text{RTS}}$ pin or $\overline{\text{CTS}}$ pin change state from active (LOW) to inactive (HIGH)

It is important to note that for the framing error, parity error, and break conditions, Line Status Register bit 7 (LSR[7]) generates the interrupt. LSR[7] is set when there is an error anywhere in the RX FIFO, and is cleared only when there are no more errors remaining in the FIFO. LSR[4:2] always represent the error status for the received character at the top of the RX FIFO. Reading the RX FIFO updates LSR[4:2] to the appropriate status for the new character at the top of the FIFO. If the RX FIFO is empty, then LSR[4:2] are all zeros.

For the Xoff interrupt, if an Xoff flow character detection caused the interrupt, the interrupt is cleared by an Xon flow character detection. If a special character detection caused the interrupt, the interrupt is cleared by a read of the ISR.



## 5.1 Interrupt Generation

- LSR is by any of the LSR bits 1, 2, 3, 4 and 7.
- RXRDY is by RX trigger level.
- RXRDY Time-out is by a 4-char delay timer.
- TXRDY is by TX trigger level or TX FIFO empty (or transmitter empty in auto RS-485 control).
- MSR is by any of the MSR bits 0, 1, 2 and 3.
- Receive Xoff/Special character is by detection of a Xoff or Special character.
- CTS# is when its transmitter toggles the input pin (from LOW to HIGH) during auto CTS flow control.
- RTS# is when its receiver toggles the output pin (from LOW to HIGH) during auto RTS flow control.

## 5.2 Interrupt Clearing

- LSR interrupt is cleared by reading all characters with errors out of the RX FIFO if it is Frame/Parity/Break Error, and is cleared by reading LSR if it is Overrun Error.
- RXRDY interrupt is cleared by reading data until FIFO falls below the trigger level.
- RXRDY Time-out interrupt is cleared by reading RHR.
- TXRDY interrupt is cleared by a read to the ISR register or writing to THR.
- MSR interrupt is cleared by a read to the MSR register.
- Xoff interrupt is cleared when Xon character(s) is received or reading ISR.
- Special character interrupt is cleared by a read to ISR
- RTS# and CTS# flow control interrupts are cleared by a read to the MSR register

## 5.3 Interrupt mode operation

In Interrupt mode (if any bit of IER[3:0] is 1) the host is informed of the status of the receiver and transmitter by an interrupt signal, IRQ#. Therefore, it is not necessary to continuously poll the Line Status Register (LSR) to see if any interrupt needs to be serviced. Figure 5 shows Interrupt mode operation.

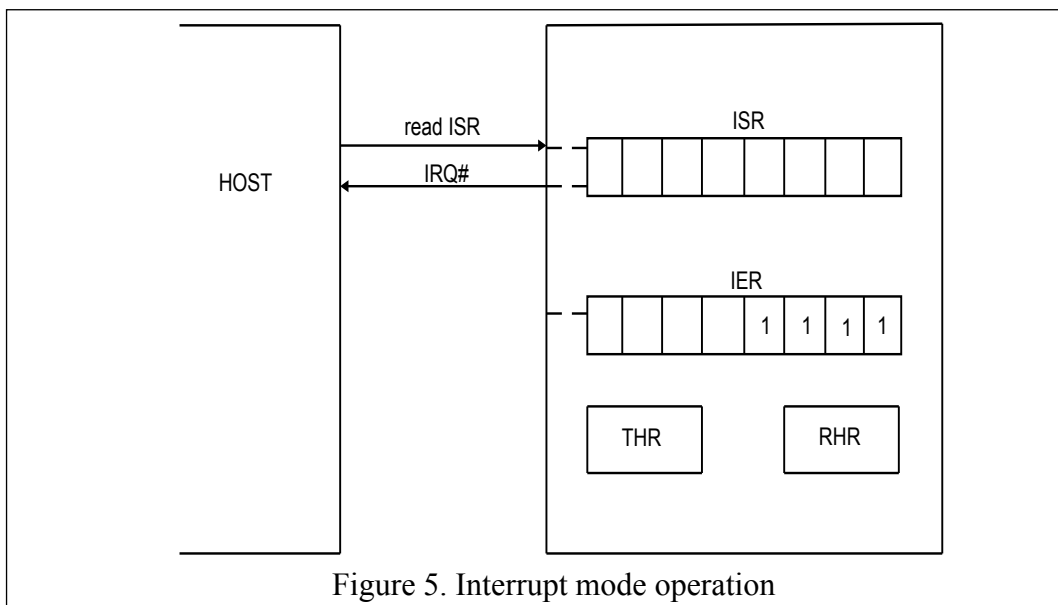
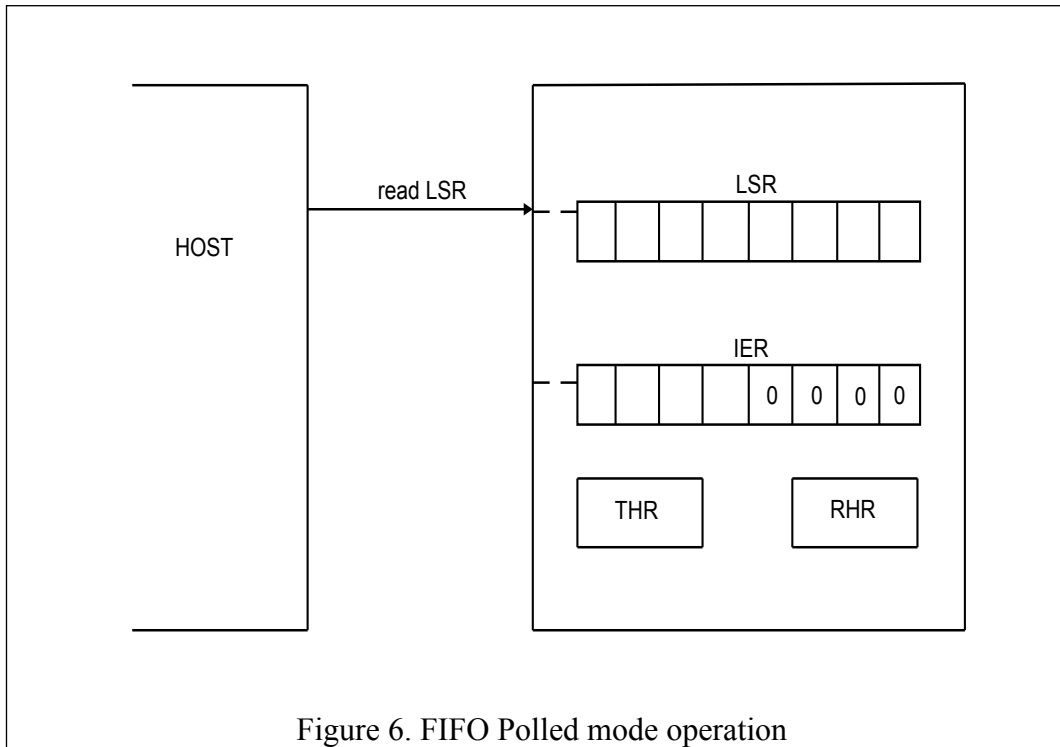


Figure 5. Interrupt mode operation

### 5.4 Polled mode operation

In Polled mode (IER[3:0] = 0000) the status of the receiver and transmitter can be checked by polling the Line Status Register (LSR). This mode is an alternative to the FIFO Interrupt mode of operation where the status of the receiver and transmitter is automatically known by means of interrupts sent to the CPU. Figure 6 shows FIFO Polled mode operation.



## 6 Sleep mode

Sleep mode is an enhanced feature of the UART. It is enabled when EFR[4], the enhanced functions bit, is set and when IER[4] is set. Sleep mode is entered when:

- The serial data input line, RX, is idle (see Section 8 “Break and time-out conditions”).
- The TX FIFO and TX shift register are empty.
- There are no interrupts pending except THR.
- Modem inputs are not toggling

Remark: Sleep mode will not be entered if there is data in the RX FIFO.

In Sleep mode, the clock to the UART is stopped. Since most registers are clocked using these clocks, the power consumption is greatly reduced. The UART will wake up when any change is detected on the RX line, when there is any change in the state of the modem input pins, or if data is written to the TX FIFO.

Remark: Writing to the divisor latches DLL and DLH to set the baud clock must not be done during Sleep mode. Therefore, it is advisable to disable Sleep mode using IER[4] before writing to DLL or DLH.

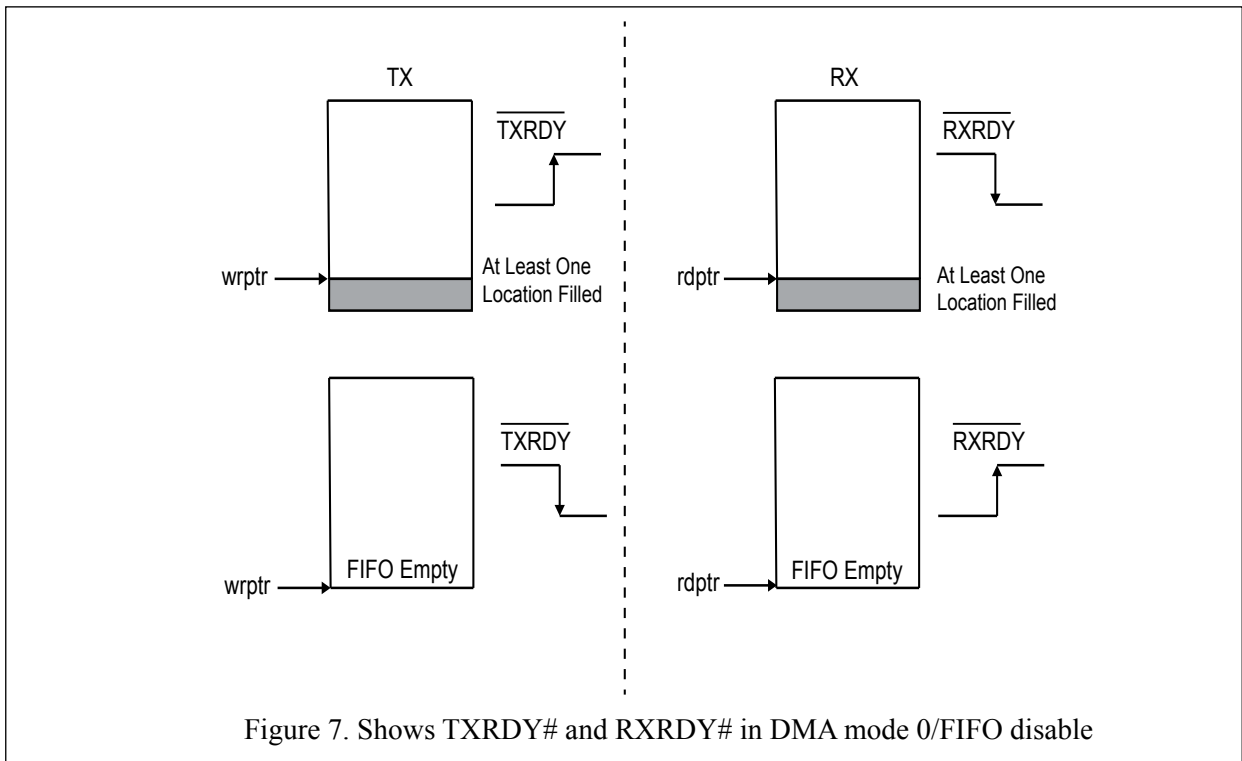
## 7. DMA Signaling

There are two modes of DMA operation, DMA mode 0 or 1, selected by FCR[3]. In DMA mode 0 or FIFO disable(FCR[0]=0), DMA occurs in single character transfers. In DMA mode 1, multicharacter DMA transfers are managed to relieve the processor for longer periods of time.

Single DMA Transfers(DMA mode 0/FIFO disable):

Transmitter: When empty, the TXRDY# signal becomes active. TXRDY# will go inactive after one character has been loaded into it.

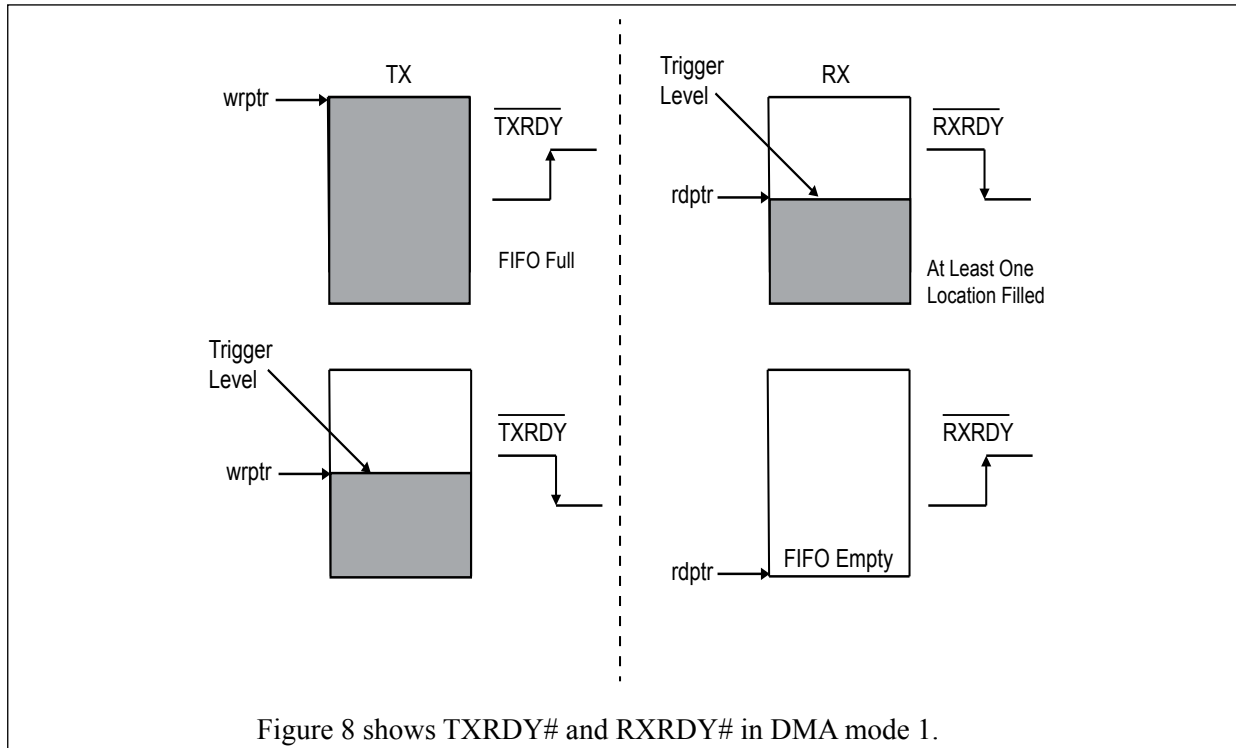
Receiver: RXRDY# is active when there is at least one character in the FIFO. It becomes inactive when the receiver is empty.



Block DMA Transfers(DMA mode 1):

Transmitter: TXRDY# is active when a trigger level number of spaces are available. It becomes inactive when the FIFO is full.

Receiver: RXRDY# becomes active when the trigger level has been reached or when a timeout interrupt occurs. It will go inactive when the FIFO is empty or an error in the RX FIFO is flagged by LSR(7).



### 8. Break and time-out condition

When the UART receives a number of characters and these data are not enough to set off the receive interrupt (because they do not reach the receive trigger level), the UART will generate a time-out interrupt instead, 4 character times after the last character is received. The time-out counter will be reset at the center of each stop bit received or each time the receive FIFO is read.

A break condition is detected when the RX pin is pulled LOW for a duration longer than the time it takes to send a complete character plus start, stop and parity bits. A break condition can be sent by setting LCR[6], when this happens the TX pin will be pulled LOW until LSR[6] is cleared by the software.

### 9. Programmable baud rate generator

The UART contains a programmable baud rate generator that takes any clock input and divides it by a divisor in the range between 1 and  $(2^{16} - 1)$ . An additional divide-by-4 prescaler is also available and can be selected by MCR[7], as shown in Figure 7. The formula for the baud rate is:

$$\text{Baud rate} = \frac{\left( \frac{\text{XTAL1 crystal input frequency}}{\text{prescaler}} \right)}{\text{divisor} \times \text{sample rate}}$$

where:

prescaler = 1, when MCR[7] is set to logic 0 after reset (divide-by-1 clock selected)

prescaler = 4, when MCR[7] is set to logic 1 after reset (divide-by-4 clock selected).

Divisor = {DLH, DLL}

Sample rate = 4 if DLD[5] = 1, or = 8 if DLD[4] = 1, or = 16-SCR+CPR if DLD[5:4] = 2'b0.

Remark: The default value of prescaler after reset is divide-by-1.

DLL and DLH must be written to in order to program the baud rate. DLL and DLH are the least significant and most significant byte of the baud rate divisor. If DLL and DLH are both zero, the UART is effectively disabled, as no baud clock will be generated.

Remark: The programmable baud rate generator is provided to select both the transmit and receive clock rates.

Table 7 to 10 show the baud rate and divisor correlation for crystal with frequency 1.8432 MHz, 3.072 MHz, 14.74926 MHz, and 24MHz respectively.

**Table 7. Baud rates using a 1.8432 MHz crystal**

Desired baud rate (bit/s)	Divisor used to generate 16x clock	Sample rate	Percent error difference between desired and actual
50	2304	16	0
75	1536	16	0
110	1047	16	0.026
134.5	857	16	0.058
150	768	16	0
300	384	16	0
600	192	16	0
1200	96	16	0
1800	64	16	0
2000	46	20	0.617
2400	48	16	0
3600	32	16	0
4800	24	16	0
7200	16	16	0
9600	12	16	0
19200	6	16	0
38400	3	16	0
56000	2	16	2.86

**Table 8. Baud rates using a 3.072 MHz crystal**

Desired baud rate (bit/s)	Divisor used to generate 16x clock	Sample rate	Percent error difference between desired and actual
50	2304	16	0
75	2560	16	0
110	1745	16	0.026
134.5	1428	16	0.034
150	1280	16	0
300	640	16	0
600	320	16	0
1200	160	16	0
1800	90	19	0.195
2000	96	16	0
2400	80	16	0
3600	45	19	0.195
4800	40	16	0
7200	25	17	0.392
9600	20	16	0
19200	10	16	0
38400	5	16	0

**Table 9. Baud rates using a 14.74926 MHz crystal**

Desired baud rate (bit/s)	Divisor used to generate 16x clock	Sample rate	Percent error difference between desired and actual
38400	24	16	0.025
56000	11	24	0.235
57600	16	16	0.025
115200	8	16	0.025
153600	6	16	0.025
921600	1	16	0.025

**Table 10. Baud rates using a 24 MHz crystal**

Desired baud rate (bit/s)	Divisor used to generate 16x clock	Sample rate	Percent error difference between desired and actual
4800	250	20	0
7200	159	21	0.17

25000	48	20	0
38400	25	25	0
57600	22	19	0.32
115200	8	26	0.16
225000	6	18	1.2
400000	3	20	0
921600	1	26	0.16
1000000	1	24	0

## 10. RS-485 features

### 10.1 Auto RS-485 RTS control

Normally the  $\overline{\text{RTS}}$  pin is controlled by MCR bit 1, or if hardware flow control is enabled, the logic state of the  $\overline{\text{RTS}}$  pin is controlled by the hardware flow control circuitry. DLD register bit 6 will take the precedence over the other two modes; once this bit is set, the transmitter will control the state of the  $\overline{\text{RTS}}$  pin. The transmitter automatically de-asserts the  $\overline{\text{RTS}}$  pin (logic 1) once the host writes data to the transmit FIFO, and asserts  $\overline{\text{RTS}}$  pin (logic 0) once the last bit of the data has been transmitted.

To use the auto RS-485  $\overline{\text{RTS}}$  mode the software would have to disable the hardware flow control function.

### 10.2 RS-485 RTS output inversion

RS485 register bit 5 reverses the polarity of the  $\overline{\text{RTS}}$  pin if the UART is in auto RS-485  $\overline{\text{RTS}}$  mode. When the transmitter has data to be sent it asserts the  $\overline{\text{RTS}}$  pin (logic 0), and when the last bit of the data has been sent out the transmitter de-asserts the  $\overline{\text{RTS}}$  pin (logic 1).

### 10.3 Auto RS-485

RS485 register bit 0 is used to enable the RS-485 mode (multidrop or 9-bit mode). In this mode of operation, a ‘master’ station transmits an address character followed by data characters for the addressed ‘slave’ stations. The slave stations examine the received data and interrupt the controller if the received character is an address character (parity bit = 1).

To use the auto RS-485 RTS mode the software would have to disable the hardware flow control function.

#### 10.3.1 Normal multidrop mode

The 9-bit mode in RS485 register bit 0 is enabled, but not Special Character Detect (EFR bit 5). 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 a line status interrupt (IER bit 2 must be set to ‘1’ at this time), and at the same time puts this address byte in the RX FIFO. After the controller examines the byte it must make a decision whether or not to enable the receiver; it should enable the receiver if the address byte addresses its ID address, and must not enable the receiver if the address byte does not address its ID address.

If the controller enables the receiver, the receiver will receive the subsequent data until being disabled by the controller after the controller has received a complete message from the ‘master’ station. If the controller does not disable the receiver after receiving a message from the ‘master’ station, the receiver will generate a parity error upon receiving another address byte. The controller then determines if the address byte addresses its ID address, if it is not, the controller then can disable the receiver. If the address byte addresses the ‘slave’ ID address, the controller take no further action; the receiver will receive the subsequent data.

### 10.3.2 Auto address detection

If Special Character Detect is enabled (EFR[5] is set and XOFF2 contains the address byte) the receiver will try to detect an address byte that matches the programmed character in XOFF2. If the received byte is a data byte or an address byte that does not match the programmed character in XOFF2, 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 a line status interrupt (IER bit 2 must be set to 1 at this time). The receiver will then receive the subsequent data from the ‘master’ station until being disabled by the controller after having received a message from the ‘master’ station.

If another address byte is received and this address byte does not match XOFF2 character, the receiver will be automatically disabled and the address byte is ignored. If the address byte matches XOFF2 character, the receiver will put this byte in the RX FIFO along with the parity bit in the parity error bit (LSR[2]).

## 11. Host interface

The host interface is 8 data bits wide with 8 address lines and control signals to execute data bus read and write transactions. The PI7C9X752 data interface supports the Intel compatible types of CPUs and it is compatible to the industry standard 16C550 UART. No clock (oscillator nor external clock) is required for a data bus transaction. Each bus cycle is asynchronous using CS# IOR# and IOW# or CS#, R/W#. All two UART channels share the same data bus for host operations. Please refer to pin description and host interface read/write timing(Fig 10 and Fig 11).

### 11.1 UART Channel Selection

During Intel Bus Mode (16/68# pin is connected to VCC), a logic 0 on chip select pins, CSA# or CSB# allows the user to select UART channel A or B to configure, send transmit data and/or unload receive data to/from the UART. Selecting all two UARTs can be useful during power up initialization to write to the same internal registers, but do not attempt to read from all two uarts simultaneously. Individual channel select functions are shown in Table 11.

Table 11. Channel A-B Select In 16 Mode

CSA#	CSB#	Function
1	1	UART de-selected
0	1	Channel A selected
1	0	Channel B selected
0	0	Channel A-B selected

During Motorola Bus Mode (16/68# pin is connected to GND), the package interface pins are configured for connection with Motorola, and other popular microprocessor bus types. In this mode the V654 decodes two additional addresses, A3 and A4, to select one of the two UART ports. The A3 and A4 address decode function is used only when in the Motorola Bus Mode. See Table 12.

Table 12. Channel A-B Select In 68 Mode

CS#	A3	Function
1	X	UART de-selected
0	0	Channel A selected
0	1	Channel B selected



## 12. Infrared Mode

The UART includes the infrared encoder and decoder compatible to the IrDA (Infrared Data Association) version 1.0 and 1.1. The IrDA 1.0 standard stipulates the infrared encoder sends out a 3/16 of a bit wide HIGH-pulse for each "0" bit in the transmit data stream with a data rate up to 115.2 Kbps. For the IrDA 1.1 standard, the infrared encoder sends out a 1/4 of a bit time wide HIGH-pulse for each "0" bit in the transmit data stream with a data rate up to 1.152 Mbps. This signal encoding reduces the on-time of the infrared LED, hence reduces the power consumption. See Figure 9 below.

The infrared encoder and decoder are enabled by setting DLD register bit-7 to a '1'. With this bit enabled, the infrared encoder and decoder is compatible to the IrDA 1.0 standard. For the infrared encoder and decoder to be compatible to the IrDA 1.1 standard, ASR bit-4 will also need to be set to a '1'. When the infrared feature is enabled, the transmit data output, TX, idles LOW. Likewise, the RX input also idles LOW, see Figure 9.

The wireless infrared decoder receives the input pulse from the infrared sensing diode on the RX pin. Each time it senses a light pulse, it returns a logic 1 to the data bit stream.

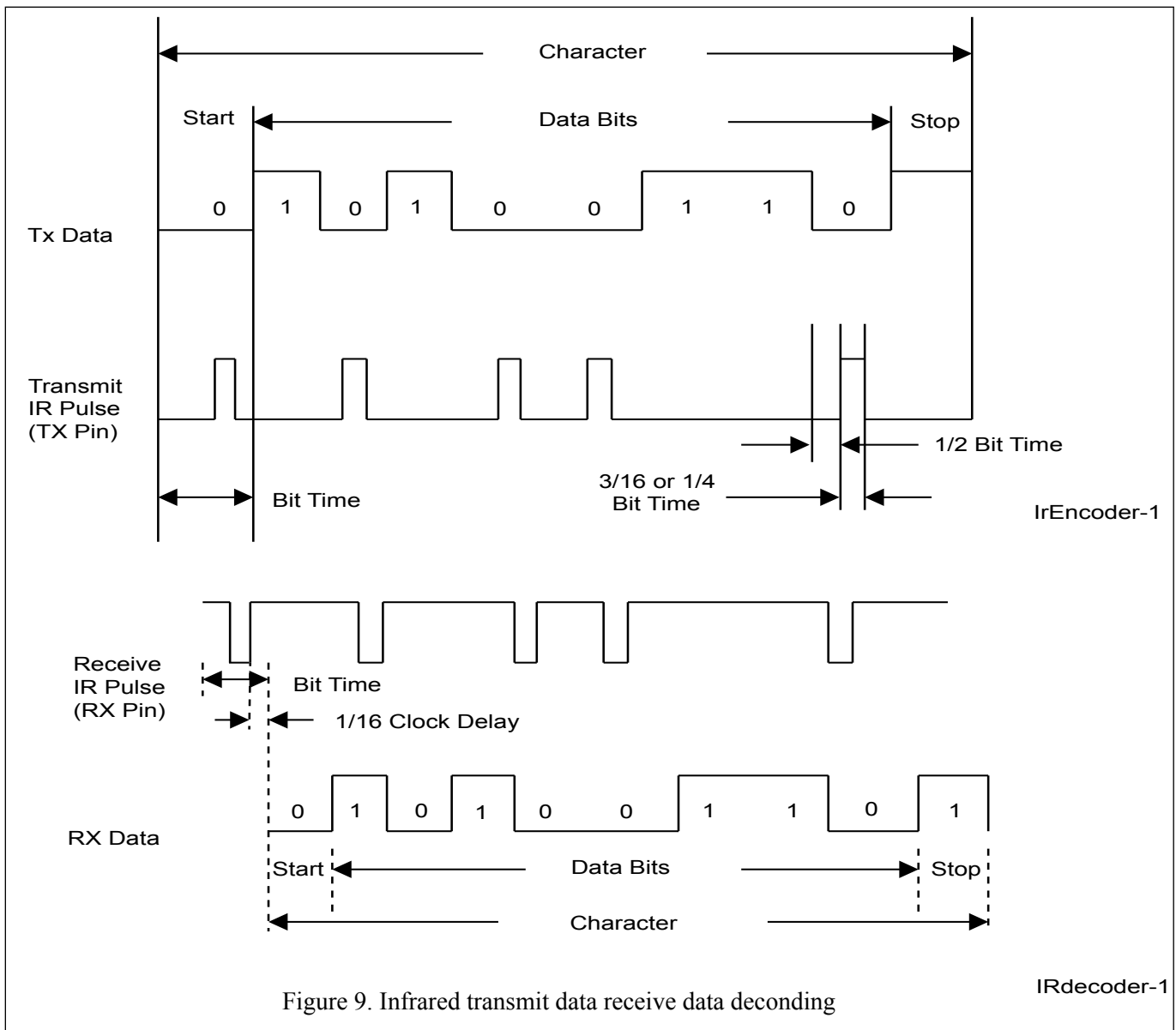


Figure 9. Infrared transmit data receive data decoding

### 13. Configuration Registers

Offset 00H(default=xxH)---Receiver Holding Register (RHR). Accessable when LCR[7]=0.		
Bit	Type	Description
[7:0]	RO	Rx Holding - When data are read from the RHR,they are removed from the top of the receiver's FIFO. Data read from the RHR when FIFO is empty are invalid. The Line Status Register(LSR) indicates the full or empty status of the FIFOs.
Offset 00H(default=xxH)---Transmitter Holding Register (THR). Accessable when LCR[7]=0		
Bit	Type	Description
[7:0]	WO	Tx Holding - When data are written to the THR,they are written to the bottom of the transmitter's FIFO. Data written to the THR when FIFO is full are lost. The Line Status Register(LSR) indicates the full or empty status of the FIFOs.
Offset 01H(default=00H)--- Interrupt Enable Register (IER). Accessable when LCR[7]=0.		
Bit	Type	Description
7	RW	CTS interrupt - "1": enable CTS/DSR interrupt
6	RW	RTS interrupt - "1": enable RTS/DTR interrupt
5	RW	Xoff/Special charactter interrupt - "1": enable the Software Flow Control interrupt
4	RW	Sleep mode - "1" : enable sleep mode. (it requires EFR[4]=1) the Uart may enter sleep mode when all conditions met: * no interrupts pending * modem inputs are not toggled * RX input pin is idling HIGH * TX/RX FIFO are empty it will exit from sleep mode when any below condition met: * modem inputs are toggling * RX input pin changed to LOW * a data byte is loaded to the TX FIFO At sleep mode, Crystal is stopped and no Uart clock
3	RW	Modem Status interrupt - "1": enable Modem Status interrupt
2	RW	Receiver Line Status interrupt - "1": enable Receiver Line Status interrupt
1	RW	Tx Ready interrupt - "1": enable THR Ready interrupt 1 = Interrupt is issued whenever the THR becomes empty in non-FIFO mode or when spaces in the FIFO is above the trigger level in the FIFO mode.
0	RW	Rx Data Ready interrupt - "1": enable Data Ready interrupt

Note: IER[7:4] can only be modified if EFR[4]=1.