TIR2000
Data Manual

High-Speed Serial Infrared Controller With 64-Byte FIFO

SLLS248A
June 1998

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.

Contents

List of Illustrations

List of Tables

1 Introduction

The TIR2000 is a serial communication controller with full infrared support which is also compatible to the TL16C550C and the TL16C750 UART devices. This device also supports the Sharp-IR, HPSIR, MIR, FIR and TV modes. The controller has a 64-byte FIFO which reduces the CPU of excessive software overhead. Also a 64-byte FIFO meets the minimum frame size requirements simplifying the software driver design. DMA and interrupt support for all operations have been included in this architecture. The TIR2000 offers programmable registers for routing interrupt DMA handshake signals. While in UART mode, the 64-byte FIFO and selectable auto-flow control for RTS and CTS increases system efficiency and baud rate.

- IrDA 1.0 mode with a data rate up to 115.2 Kbps
- IrDA 1.1 mode with a data rate up to 1.15 Mbps
- IrDA 1.1 mode with a data rate up to 4 Mbps
- Sharp ASK infrared mode
- Consumer television remote control mode (RC5, RC5 extended, NEC, RC6, and RECS80)

1.1 Features

- Full infrared support
  - Infrared Data Association (IrDA™) 1.0 supports up to 115.2 kbps
  - IrDA 1.1 supports 1.15 Mbps and 4 Mbps
  - Sharp Amplitude Shift Keying (ASK)
  - TV remote control mode
- Industry Standard Architecture (ISA) compatible bus interface
- Selectable 16- or 64-byte FIFO
- Full duplex infrared transmission and reception
- Controlled transmit start
- Supports back-to-back transactions
- Supports multiple optical transceivers
- Controlled Serial Interaction Pulse (SIP) generation
- Supports 11 IRQ options and 3 DMA configurations
- Power management support
- 8 general purpose I/O terminals
- Fully compatible with TL16C450, TL16C550C, TL16C750 UARTs
- Automatic fallback to TL16C550C mode
- UART baud rate up to 1 Mbps
- Auto-flow control in UART mode
- 3.3-V or 5-V operation
- Available in 64-pin TQFP package

1.2 Functional Block Diagram

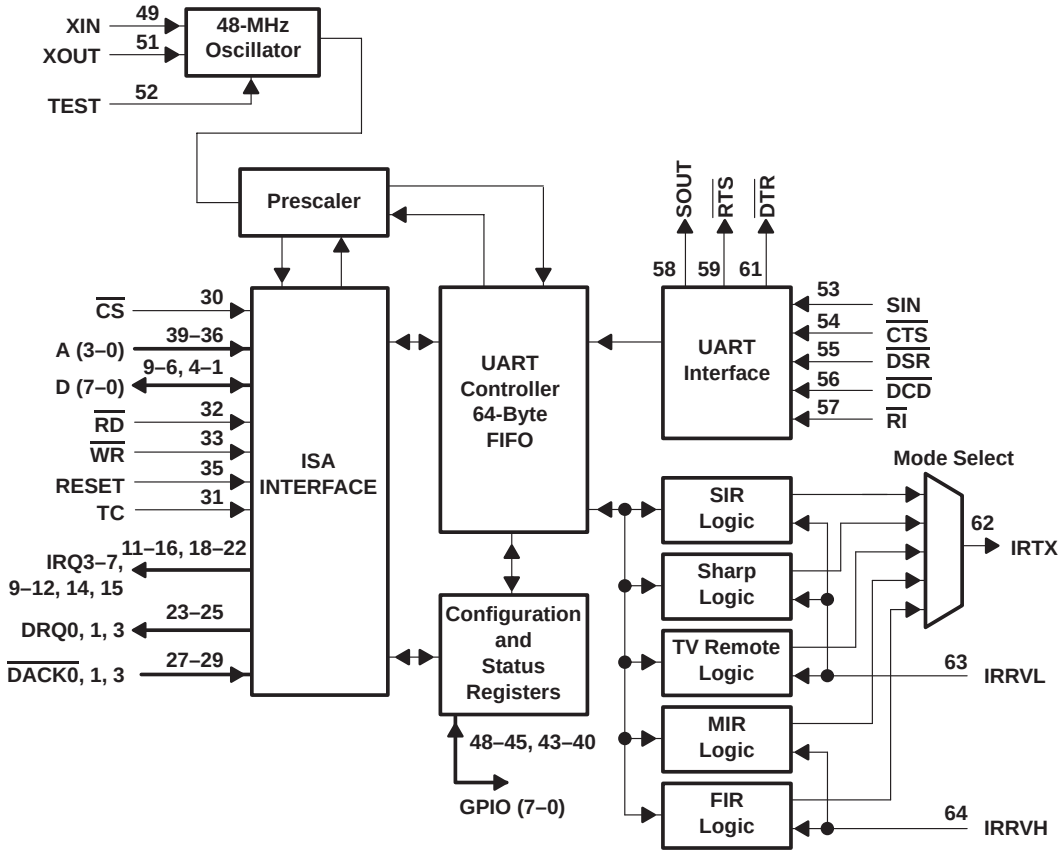


Figure 1–1. Functional Block Diagram

1.3 Terminal Assignments

PAG PACKAGE
(TOP VIEW)

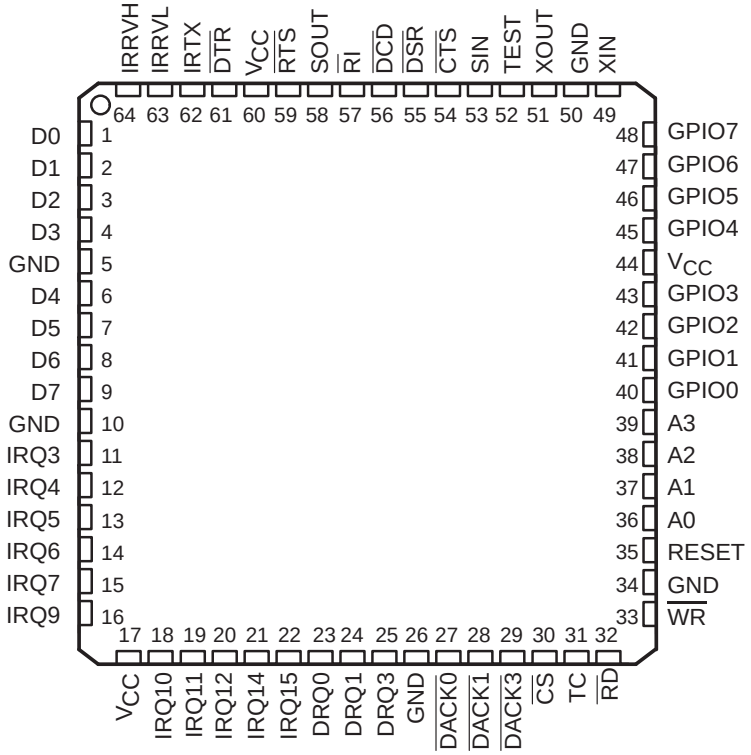


Figure 1–2. Terminal Assignments

1.4 Ordering Information

T_A	PACKAGE
	QUAD FLAT PACK (PAG)
0°C to 70°C	TIR2000PAG

1.5 Terminal Functions

TERMINALS NAME	TERMINALS NO.	I/O	DESCRIPTION
A0 – A3	36–39	I	Address bus. The CPU uses A0 -A3 and the $\overline{CS}$ signal to select the internal register of the TIR2000. A0 -A3 are decoded to select a particular register.
$\overline{CS}$	30	I	Chip select. The CPU uses $\overline{CS}$ to select the TIR2000 for read/write transactions.
$\overline{CTS}$	54	I	Clear to send. $\overline{CTS}$ is a modem-status signal. Its condition can be checked by reading bit 4 ($\overline{CTS}$) of the modem-status register (MSR). Bit 0 (ΔCTS) of the MSR indicates that $\overline{CTS}$ has changed states since the last read from the MSR. When modem-status interrupt is enabled, $\overline{CTS}$ changes states, and the auto-$\overline{CTS}$ mode is not enabled, an interrupt is generated. $\overline{CTS}$ is also used in the auto-$\overline{CTS}$ mode to control the transmitter.
D0–D7	1–4, 6–9	I/O	Data bus. These bidirectional data lines are connected to the CPU for data transfer between the TIR2000 and the CPU. D0 is the LSB and D7 is the MSB.
$\overline{DACK0}$, $\overline{DACK1}$, $\overline{DACK3}$	27–29	I	DMA acknowledge. $\overline{DACK0}$, $\overline{DACK1}$, and $\overline{DACK3}$ are DMA active low signals that are the corresponding acknowledge signals, for the DMA request signals which are DRQ0, DRQ1 and DRQ3.
$\overline{DCD}$	56	I	Data carrier detect. $\overline{DCD}$ is a modem-status signal. Its condition can be checked by reading bit 7 (DCD) of the MSR. Bit 3 (ΔDCD) of the MSR indicates that $\overline{DCD}$ has changed states since the last read from the MSR. When the modem-status interrupt is enabled and $\overline{DCD}$ changes state, an interrupt is generated.
DRQ0, DRQ1, DRQ3	23–25	O	DMA requests. DRQ0, DRQ1 and DRQ3 are used for DMA requests that are active high and are used to signal the DMA controller that data transfer between the TIR2000 and memory is required. When the DMA is enabled, one of the three channels configurable through the DMA channel select register (DCSR) can be selected.
$\overline{DSR}$	55	I	Data set ready. $\overline{DSR}$ is a modem-status signal. Its condition can be checked by reading bit 5 (DSR) of the MSR. Bit 1 (ΔDSR) of the MSR indicates the $\overline{DSR}$ has changed states since the last read from the MSR. When the modem-status interrupt is enabled and the $\overline{DSR}$ changes states, an interrupt is generated.
$\overline{DTR}$	61	O	Data terminal ready. When low, $\overline{DTR}$ informs a modem or data set that the UART is ready to establish communication. $\overline{DTR}$ is placed in the active state by setting the DTR bit of the modem-control register to one. $\overline{DTR}$ is placed in the inactive condition as a result of a master reset, during loop mode operation, or clearing of the DTR bit.
GND	5, 10, 26, 34, 50		Ground (0 V). GND terminals must be tied to ground for proper operation.
GPIO0–GPIO3, GPIO4–GPIO7	40–43, 45–48	I/O	General purpose I/O terminals. GPIO0–GPIO7 terminals are programmable general-purpose input/output terminals.
IRQ3–IRQ7, IRQ9–IRQ12, IRQ14, IRQ15	11–16, 18–22	O	Interrupt signals. These active-high interrupts are activated based on the IRQ configuration register.
IRRVH	64	I	Infrared receive. IRRVH is an input connected to the IR transceiver module that receives data in the high-speed modes (1.15 Mbps and 4 Mbps mode).

1.5 Terminal Functions (Continued)

IRRVL	63	I	Infrared receive. IRRVL is connected to the IR transceiver module and receives data in the slow-speed mode such as the SIR, Sharp Ir, and TV remote-control modes.
IRTX	62	O	Infrared transmit. IRTX is an output terminal connected to the IR transceiver module and transmits data out from the TIR2000.
$\overline{RD}$	32	I	Read. $\overline{RD}$ is an active-low signal that indicates when the CPU reads data from the TIR2000.
RESET	35	I	Reset. When active (high), RESET is used for system reset operation.
$\overline{RI}$	57	I	Ring indicator. $\overline{RI}$ is a modem-status signal. Its condition can be checked by reading bit 6 (RI) of the MSR. Bit 2 (TERI) of the MSR indicates that $\overline{RI}$ has transitioned from a low to a high level since the last read from the MSR. If the modem-status interrupt is enabled when this transition occurs, an interrupt is generated.
$\overline{RTS}$	59	O	Request to send. When low, $\overline{RTS}$ informs the modem or data set that the ACE is ready to receive data. $\overline{RTS}$ is set low by setting the RTS MCR bit and is set to its inactive (high) level either as a result of a master reset, during loop mode operations, or by clearing bit 1 (RTS) of the MCR. In the auto-$\overline{RTS}$ mode, $\overline{RTS}$ is set low by the receiver threshold control logic.
SIN	53	I	Serial data. SIN is the input from a connected communications device.
SOUT	58	O	Composite serial data output. SOUT is connected to a communication device. SOUT is set to the marking (high) level as a result of master reset.
TC	31	I	Terminal count. TC comes from the DMA controller and indicates the end of the block transfer.
TEST	52	I	Test terminal. TEST is used for test purposes only and should be tied low in normal operation.
VCC	17, 44, 60		5-V or 3-V supply voltage.
$\overline{WR}$	33	I	Write. $\overline{WR}$ is an active low signal that indicates that the CPU is writing data to the TIR2000.
Xin, Xout	49, 51	I/O	Crystal connectors. A 48-MHz crystal should be connected across these terminals.

(This page has been left blank intentionally.)

2 Functional Description

2.1 Modes of Operation

The TIR2000 operates in six different modes. For proper operation, the device must be programmed correctly for each mode. The various modes of operation include:

- Fast-speed infrared (IrDA FIR) mode
- Medium-speed infrared (IrDA MIR) mode
- Slow-speed infrared (IrDA SIR) mode
- SHARP Infrared (IR) mode
- TV mode
- Universal asynchronous receiver transmitter (UART) mode

Basic data communication involves at least two devices. During the data transmit mode, the first device transmits data and the second device receives the transmitted data. During the receive mode, the second device transmits the data and the first device receives the data. When the device is configured for the UART mode, both data transmission and data receive can occur simultaneously, but during the infrared (IR) modes either the data transmit or data receive is possible at any time but not simultaneously. A basic configuration for two communicating devices in the infrared mode is shown in Figure 2–1.

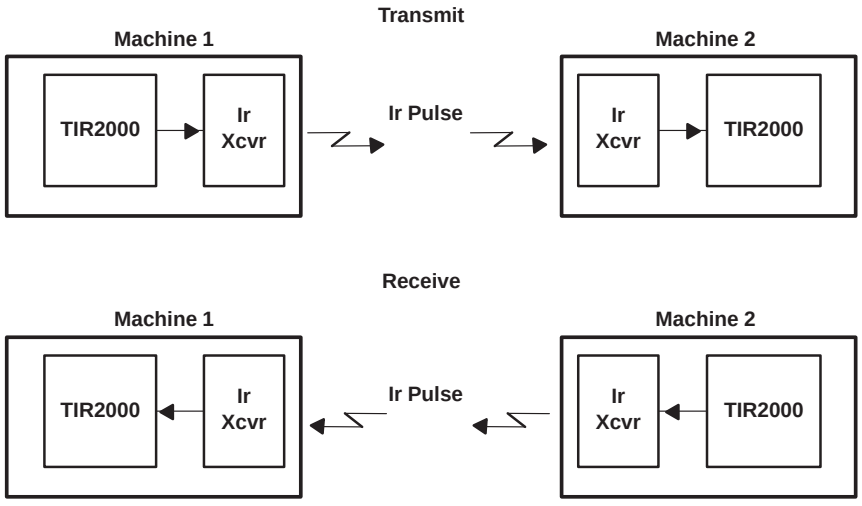


Figure 2–1. Basic Configuration for Ir Communication

The TIR2000 has internal programmable registers. Some of the registers are common to all modes of operation and some are specific to a particular mode of operation. The device is accessed with the industry standard architecture (ISA) address bus SA[15–0]. An outside address decoder uses ISA signals (SA[15–4], AEN) and generates the chip select ($\overline{CS}$) signal for the TIR2000 device. The lower address bits SA[3–0] are directly connected to the device. The internal address decoder decodes SA[3–0] lower address bits to access the different registers.

2.1.1 Fast-Speed Infrared (FIR) mode (IrDA 1.1)

During the FIR mode, data transfer takes place between the CPU and peripheral devices at the baud-rate speed of 4 Mbps. An FIR transmit frame starts with the Preamble, followed by a Start Flag, Frame Data, CRC-32, and Stop Flag. The details of the frame format can be found in the Infrared Data Association (IrDA) document physical layer link specification

Preamble	Start Flag	Frame Data	CRC-32	Stop Flag

The peripheral device that is in the data-transmission mode attaches the Preamble, Start Flag, CRC-32, and the Stop Flag. It also encodes the transmit data into the 4PPM format. During data receive, the peripheral device in the data-receive mode recovers the data-receive clock, removes the Start Flag, decodes the 4PPM incoming data and determines the frame boundary with the reception of a Stop Flag. It also checks for errors such as illegal pulse-position-modulation (4PPM) symbols, cyclic reduncy check (CRC) errors and frame-length errors. At the end of a frame reception, the CPU reads the LSR-status register to find the errors, if any, of the received frame.

Data cannot be transmitted and received by the same device at the same time. When the CPU wants to send data, the peripheral device needs to be programmed for data transmission. When the CPU receives data from the peripheral device, the peripheral device needs to be programmed for data receive. Some registers are common to both data transmit and data receive and need to be programmed only once. Use the following programming sequence for these registers:

1. Program the interrupt configuration register (ICR) in order to:
   - Select the IRQ channel
   - Select either totem-pole or open-drain output
   - Select an active high or an active low for an interrupt when the totem-pole configuration is selected.
2. Write a 6 to the PRESC register. This divides a 48-MHz input clock by 6 and generates a 8-MHz internal clock for the FIR mode of operation.

2.1.1.1 FIR Data-Transmission Mode

The following register programming steps should be performed for transmission in the FIR mode:

1. Program the mode definition register (MDR) in order to :
   - Select the FIR mode
   - Enable/disable the low-power mode. For normal operation, enable the low-power mode.
   - Enable/disable the sleep mode. For normal operation, disable the sleep mode.
   - Enable/disable the store-and-controlled transmission (SCT). For normal operation, disable the SCT.
   - Enable/disable the software control on a 2-µs infrared pulse. For normal operation, disable the software control.
   - Select the frame closing method (frame-length method or set-EOT bit method).
2. Program the preamble length register (PLR) in order to:
   - Select the number of preambles. The default value is 16.
   - Select the TX FIFO trigger level
3. When the frame-length method is selected for frame termination, program the TXFLL and the TXFLH registers for the frame length value.

4. Reset the TX FIFO. This includes:
    - Write to the LCR register so that the LCR[7] bit is a 1. The LCR[7] bit must be a 1 to be able to write to the FCR[5] bit.
    - Program the FCR register to select 64-/16-bytes of the TX FIFO, the non-DMA/DMA mode of operation, select the TX FIFO trigger level and to clear the TX FIFO.
    - Write the LCR register to set the LCR[7] bit to a 0 for normal operation.
5. Program the IER register to enable only the transmitter related interrupts and disable the remaining interrupts. Enable the following:

    For programmed I/O mode
    - Transmitter FIFO below threshold level interrupt enable (IER[1])
    - Transmitter underrun interrupt enable (IER[7])

    For DMA mode
    - Status FIFO time-out interrupt enable (IER[2])
    - Status FIFO threshold interrupt enable (IER[4])
    - Transmitter underrun interrupt enable (IER[6])
6. Write a 1 to the ACREG[7] bit to enable data transmission
7. Write a 1 to the MCR[3] bit. This enables the selected IRQ channel.

2.1.1.1.1 Frame-Closing Methods

There are two ways a transmission frame can be properly terminated: frame-length method and set-EOT bit method. The two methods are described as follows:

- Frame-Length method: This method is selected when the MDR[7] bit equals 0. The CPU writes the frame-length value to the TXFLH and TXFLL registers. The device automatically attaches an end flag to the frame when the number of bytes transmitted becomes equal to the TXFLH and TXFLL value.
- Set-EOT bit method: This method is selected when the MDR[7] bit equals 1. The CPU writes a 1 to the EOT bit of the ACREG[0] register just before it writes the last byte to the TX FIFO. When the CPU writes the last byte to the TX FIFO, the device internally sets the tag bit for that particular byte in the TX FIFO. As the peripheral device in the transmit mode reads bytes from the TX FIFO, the flag-bit information is used to attach an end flag and properly terminate the frame.

2.1.1.1.2 Store and Controlled Data Transmission

In the store and controlled transmission (SCT), the CPU initially writes (stores) the data in the TX FIFO. After the CPU writes a part of the frame (for a bigger frame) or the whole frame (a small frame such as a supervisory frame), it enables another bit (ACREG[2] in this case) to start transmission. The SCT is effective when the CPU writes a 1 to the MDR[5] bit. This method of transmission is different from the normal mode of transmission (MDR[5] = 0) where transmission of data starts right after the CPU writes the first byte to the TX FIFO. The SCT sends short frames without a TX underrun.

2.1.1.1.3 1.6-µs Infrared (IR)-Pulse Select Method

During the MIR and FIR mode of operation, the transmitter sends a 1.6-µs pulse at least once every 500 ms. The purpose of this special pulse is to inform the slow device (in SIR mode) that the high-speed device involved in data transaction is currently occupied. When the MDR[6] bit is a 0 (the default value), the peripheral device in the transmit mode always sends a 1.6-µs pulse at the end of a transmission frame. However, when bit MDR[6] is a 1, the transmission of a 1.6-µs pulse depends on the ACREG[3] bit value. The CPU keeps a timer and sets the ACREG[3] bit at least once in every 500 ms. When the MDR[6] bit is a 1, the peripheral device in the transmit mode sends a 1.6-µs pulse only if the ACREG[3] bit is a 1. The advantage to this approach over the default value sent-always-1.6-µs pulse-at-the-end approach is that the peripheral device in the transmit mode need not send the special 1.6-µs pulse at the end of every frame. Sending a 1.6-µs pulse at the end of every frame may increase overhead, mainly in the MIR mode.

2.1.1.1.4 Data-Transmission Underrun

An underrun during data transmission occurs when the CPU fails to supply the data to the TX FIFO and the TX FIFO becomes empty before the end of the frame is transmitted. When an underrun occurs, the device closes the frame with an end-flag but attaches an incorrect CRC value. The receiving device detects the CRC error and discards the frame. The device sets an internal flag and further transmission of data is disabled. The CPU must reset the TX FIFO and read the RESUME register. This read operation clears the internal flag.

2.1.1.2 FIR Data-Receive Mode

The following register programming steps are required to receive data in FIR mode:

1. Program the mode definition register (MDR) to:
   - Select the FIR mode
   - Enable the low-power mode
   - Disable the sleep mode
2. Program the RXFLL and RXFLH registers for the receive-frame length value. The maximum length of a frame is 2048 bytes. The RXFLL register stores the lower eight bits and the RXFLH register stores the remaining bits. When the intended maximum receive-frame length is n, program the RXFLH and RXFLL registers to be n + 5.
3. Reset the RX FIFO. This includes:
   - Write to the LCR register that the LCR[7] bit is a 1. The LCR[7] bit must be a 1 to be able to write to the FCR[5] bit.
   - Program the FCR register to select 64-/16-bytes of the RX FIFO, the non-DMA/DMA mode of operation, select the RX FIFO trigger level and to clear the RX FIFO.
   - Write the LCR register to set the LCR[7] bit to a 0 for normal operation.
4. Program the IER register to enable only the receive related interrupts and disable the remaining interrupts. Enable the following:
   - RX threshold interrupt (IER[0] bit)
   - The last byte from the RX FIFO interrupt (IER[2] bit)
   - The RX FIFO overrun interrupt (IER[3] bit)
   - The received end of the frame interrupt (IER[7] bit)

5. Write a 1 to the ACREG[6] to enable data receive.
6. Write a 1 to the MCR[3] bit. This enables the selected IRQ channel.

This completes the programming of the registers for data receive and now the device is ready to receive data. The device decodes the serial data, converts it from serial data to parallel data and stores the data bytes in the RX FIFO. When the stored data in the RX FIFO reaches the set threshold level, the device interrupts the CPU. When the CPU is interrupted, it reads the IIR to identify the source of the interrupt. When the source of the interrupt is the IIR[0] bit and not the IIR[7] bit, the CPU goes to the threshold mode.

2.1.1.2.1 Threshold Mode

During threshold mode, the CPU reads a number of bytes (determined by the RX FIFO threshold level) from the RX FIFO. An example: When the RX threshold value is set to be 16, the CPU can perform 16 consecutive read operations from the RX FIFO. When the source of interrupts is the IIR[7] bit, the CPU goes to the byte mode.

2.1.1.2.2 Byte Mode

The CPU will read the majority of the frame in the threshold mode. However, the last bits of data may not fall on the exact threshold boundary and will not be able to interrupt the CPU with the IIR[0] bit. The device interrupts the CPU with the IIR[7] bit when the machine in the data-receive mode detects the end of a frame. The CPU disables the IER[0] bit and reads one byte from the RX FIFO each time and checks for the interrupt (IIR[2]) bit. When the CPU reads the last byte of a frame from the RX FIFO, the device interrupts the CPU with the IIR[2] bit. Once the CPU receives the IIR[2] bit, it should not read further from the RX FIFO and that is the end of the frame. The last four bytes that the CPU read from the RX FIFO are the CRC value and not the actual data. The CPU enables the IER[0] bit. After the CPU reads the last byte of a frame from the RX FIFO, it reads the LSR register to determine the sources or errors, if any, for that frame. When the source of the interrupt is the IIR[3] bit, there has been an overrun while receiving data and the CPU will service the overrun.

2.1.1.2.3 Data-Receive Overrun

An overrun occurs during data receive if the CPU cannot timely read out data from the RX FIFO and the RX FIFO is overwritten. When an overrun occurs, the device interrupts the CPU with the IIR[3] bit and discards the remaining portion of the frame. When an overrun occurs, the device sets an internal flag and the receive operation of the next frame is disabled. Before the next frame can be received, the CPU must reset the RX FIFO and read the RESUME register. This read operation clears the internal flag.

2.1.2 Medium-Speed Infrared (MIR) mode (IrDA 1.1)

During the medium speed infrared (MIR) mode, data transfer takes place between the CPU and peripheral devices at a speed of 1.15 Mbs. A MIR transmit frame starts with a minimum of two Start Flags followed by the Frame Data, CRC–16 and ends with a Stop Flag.

Start Flag	Frame Data	CRC-16	Stop Flag

The peripheral device in the data transmission mode of operation attaches the Start Flags, CRC-16, and a Stop Flag. It also looks for five consecutive ones in the frame data and automatically inserts a zero after five consecutive ones. This operation is known as bit stuffing. On a receive operation, the machine in the data receive mode recovers the receive clock, removes the Start Flag, de-stuffs the incoming data and determines frame boundary with the reception of the Stop Flag. It also checks for errors such as frame abort, CRC error and frame-length error. At the end of a frame reception, the CPU reads the LSR status register to find out if any errors occurred with the received frame.

The device can both transmit and receive data, but cannot do both at the same time. When the CPU wants to send data, the perpheral device must be programmed for data transmission. When the CPU receives data from the peripheral device, the perpheral device must be programmed for data receive. Some registers are common to both data transmit and data receive and are programmed only once. Use the following programming sequence for these registers:

1. Program the interrupt configuration register (ICR) in order to:
    - Select the IRQ channel
    - Select either totem-pole or open-drain output
    - Select an active high or an active low for an interrupt when the totem-pole configuration is selected.
2. In the MIR mode, the 48-MHz input clock is automatically divided by 1.5 to generate the 28x MIR clock. However, a 6 should be written to the PRES register which is used as the base clock for the status FIFO timeout interrupt.

2.1.2.1 MIR-Data Transmission Mode

The following register programming steps are performed for transmission in the MIR mode:

1. Program the mode-definition register (MDR) to:
    - Select the MIR mode
    - Enable/disable the low-power mode. The low-power mode is enabled for normal operation.
    - Enable/disable the sleep mode. The sleep mode is disabled for normal operation.
    - Enable/disable the store and controlled transmission. The SCT is disabled for normal operation.
    - Enable/disable the software control on the 1.6-μs infrared pulse. The software control is disabled during normal operation.
    - Select the frame closing method (frame-length method or set–EOT bit method)
2. Program the PLR to:
    - Select the number of Start Flags. The default value is two.
    - Select the TX FIFO trigger level.
3. When the frame-length method is selected for frame closing, program the TXFLL and the TXFLH registers for the frame length.
4. Reset the TX FIFO:
    - Write the LCR register so that the LCR[7] bit is a 1. The LCR[7] bit must be a 1 in order to write to the FCR[5] bit.
    - Program the FCR to select 64-/16-bytes of TX FIFO, non-DMA/DMA mode of operation , and to clear the TX FIFO.
    - Write the LCR register so that the LCR[7] bit is a 0 for normal operation.
5. Program the IER register to enable only the transmission related interrupts and disable the remaining interrupts. The TX threshold interrupt and the TX underrun interrupt are enabled.
6. Write a 1 to the ACREG[7] bit to enable data transmission.
7. Write a 1 to the MCR[3] bit to enable the selected IRQ channel.

When the CPU receives an interrupt, it reads the IIR to identify the source of the interrupt. When the source of the interrupt is an IIR[1] bit, the CPU requires the following:

- Disable the IER[1] bit.
- Write data bytes (maximum number limited to the number selected by the TX FIFO trigger level) to the TX FIFO.
- Enable the IER[1] bit.

When the source of the interrupt is the IIR[5] bit, there has been an underrun while transmitting data and the CPU needs to service the underrun. See paragraph 2.1.1.1.4.

2.1.2.2 MIR Data Receive Mode

The following register programming steps are required to receive data in MIR mode:

1. Program the mode definition register (MDR) in order to:
   - Select the MIR mode
   - Disable the low power mode
   - Disable the sleep mode
2. Program the RXFLL and RXFLH registers for maximum receive-frame length value. The maximum length of a frame is 2048 bytes. The RXFLL register stores the lower eight bits and the RXFLH register stores the remaining upper bits. When the intended maximum receive-frame length is n, program the RXFLH and RXFLL registers to be n + 3.
3. Reset the RX FIFO. This includes:
   - Write to the LCR register so that the LCR[7] bit is a 1. LCR[7] bit must be a 1 to be able to write to the FCR[5] bit.
   - Program the FCR register to select 64-/16- bytes of the RX FIFO, the non-DMA/DMA mode of operation, select the RX FIFO trigger level and to clear the RX FIFO.
   - Write the LCR register to set the LCR[7] bit to a 0 for normal operation.
4. Program the IER register to enable only the receive related interrupts and disable the remaining interrupts. Enable the following:
   - RX threshold interrupt (IER[0] bit)
   - The last byte from the RX FIFO interrupt (IER[2] bit)
   - The RX FIFO overrun interrupt (IER[3] bit)
   - The received end of the frame interrupt (IER[7] bit)
5. Write a 1 to the ACREG[6] bit to enable data receive.
6. Write a 1 to the MCR[3] bit to enable the selected IRQ channel.

This completes the programming of the registers for data receive and now the device is ready to receive data.When the CPU is interrupted, it reads the IIR to identify the source of the interrupt. When the source of the interrupt is the IIR[7] bit, the CPU goes to the byte mode. See paragragh 2.1.1.2.2. The CRC value is the last two bytes instead of the last four bytes as in the FIR data-transmission mode.

When the source of the interrupt is the IIR[0] bit and not the IIR[7] bit, the CPU goes into the threshold mode. See paragraph 2.1.1.2.1. If the source of the interrupt is the IIR[4] bit, there has been an overrun while receiving data and the CPU will need to service the overrun. See paragraph 2.1.1.2.3.

2.1.3 Slow-Speed Infrared (SIR) Mode (IrDA 1.0)

The SIR mode of operation is similar to the UART mode except that data communication takes place in a slow-speed half-duplex manner. The method of data transfer is infrared instead of on a wired path. The modem-control register (MCR) and the Modem Status Register (MSR) are not used.

Data transfer occurs with a baud rate of up to 115.2 Kbps. The format of serial data is similar to the UART data format. Each data byte starts with a start bit (0), 1 byte of data, and then ends with at least a stop bit (1). Each serial data bit is encoded before transmission and decoded after being received. A 1 is decoded with no IR pulse and a 0 is decoded by sending 3/16th of one bit time IR pulse. Similarly, the received serial pulse is decoded as a 0 and the absence of an IR pulse is decoded as a 1 (see Figure 2–2).

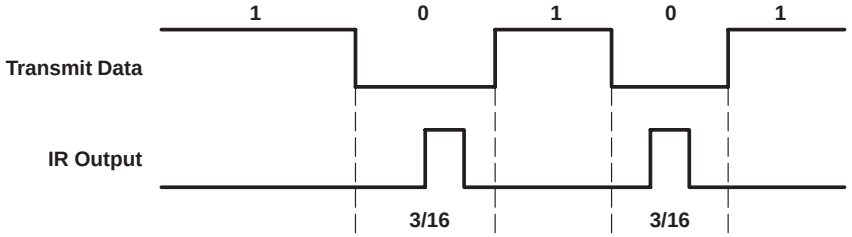


Figure 2–2. SIR Encoding

The following register programming steps are required to receive or transmit data in SIR mode transmissions:

1. Program the ICR register.
2. Program the prescaler (PRESC) register to ensure the prescaler output is approximately 1.843 MHz. The default PRESC value is used for an input clock frequency of 48-MHz.
3. Program the auxiliary baud-rate divisor lower byte (ABDL) and the auxiliary baud-rate divisor higher byte (ABDH) registers to select the desired communication baud rate for the UART mode. The baud rate is given by:

$$\text{Baud rate} = \frac{X_{IN}}{\text{PRESC} \times \text{Divisor} \times 16} \quad (1)$$

4. Program the mode definition register (MDR) to:
   - Select the SIR mode
   - Enable/disable the low-power mode. For normal operation, the low-power mode is enabled.
   - Enable/disable the sleep mode.

2.1.3.1 SIR Data Transmission Mode

The following register programming steps are required for data transmission in the SIR mode:

1. Program the PLR register to select the TX FIFO trigger level.
2. Reset the TX FIFO and program the FCR register.
3. Program the IER register to enable only the data transmission interrupts.
   - Enable the TX holding-register-empty interrupt
   - Enable the TX underrun interrupt
   - Disable the RX FIFO over the threshold interrupt
   - Disable the RX FIFO overrun error interrupt

4. Program the IRCFG[3] bit to select fixed 1.6 μs or 3/16th pulse width
5. Write a 1 to the ACREG[7] bit to enable data transmission.
6. Write a 1 to the MCR[3] bit to enable the selected IRQ channel

2.1.3.2 SIR Data Receive Mode

The following register programming steps are required to receive data in SIR mode:

1. Reset the RX FIFO. This includes:
   - Write the LCR register to set the LCR[7] bit to a 1. The LCR[7] bit must be a 1 in order to be able to write to the FCR(5) bit.
   - Program the FCR to select 64-/16-bytes RX FIFO register, non-DMA mode of operation, select the RX FIFO trigger level and to clear the RX FIFO.
   - Write the LCR register so that the LCR[7] bit = 0 (for normal operation).
2. Program the IER register to enable the receive related interrupts which include:
   - RX threshold interrupt (IER[0]) bit
   - Modem-Status interrupt enable (IER[3])
   - Disable the TX related interrupts (TX FIFO below threshold interrupt and the TX underrun interrupt).
3. Write a 1 to the ACREG[6] bit to enable data receive.

2.1.4 SHARP Infrared (IR) mode

The SHARP IR mode of operation is similar to the SIR mode except the modulation/demodulation of the serial data is different. The SHARP IR mode uses amplitude shift keying (ASK) modulation/demodulation. Data transfer can take place up to a 38.4K baud rate. In the ASK modulation, a 0 is signaled by sending a 500-kHz continuous pulse and a 1 is signaled by the absence of an IR pulse (see Figure 2–3).

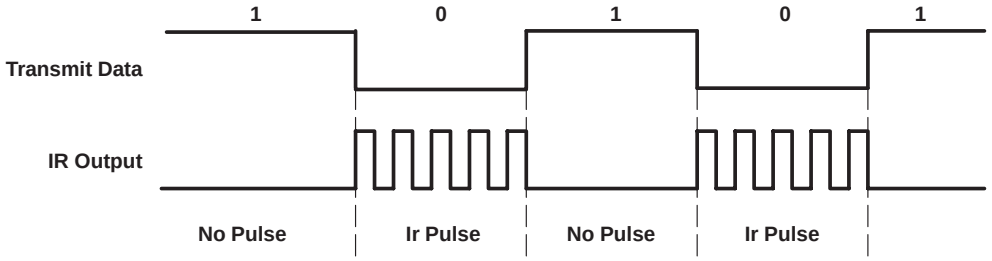


Figure 2–3. SHARP Ir Encoding

The following register programming steps are required for encoding in the SHARP Ir mode:

1. Program the ICR register
2. Program the prescaler (PRESC) register to insure the prescaler output is approximately 1.43 MHz. The default PRESC value is used for an input clock frequency of 48 MHz.
3. Program the auxiliary baud rate divisor lower byte (ABDL) and the auxiliary baud rate divisor higher byte (ABDH) registers to select the desired communication baud rate for the UART mode. The baud rate is given by:

$$\text{Baud rate} = \frac{X_{IN}}{\text{PRESC} \times \text{Divisor} \times 16} \quad (2)$$

4. Program the mode-definition register (MDR) in order to:
    - Select the SHARP Ir mode
    - Enable/disable the low-power mode. For normal operation, the low-power mode is enabled.
    - Enable/disable the sleep mode. For normal operation, the sleep mode is disabled.
5. Program the TV configuration (TVCFG) register in order to:
    - Enable/disable the receiver internal demodulation.
    - Select the transmission carrier-frequency range.
    - Select the receiver carrier-frequency range.

2.1.4.1 SHARP IR Data-Transmission Mode

The following register programming steps are required for data transmission in the SHARP IR mode:

1. Program the PLR to select the TX FIFO trigger level.
2. Reset the TX FIFO. This includes the following:
    - Write to the LCR register to set the LCR[7] bit to a 1. The LCR[7] bit must be a 1 to be able to write to the FCR(5) bit.
    - Program the FCR register to select 64-/16-bytes TX FIFO, non-DMA mode of operation, select the RX FIFO trigger level and to clear the TX FIFO.
    - Write the LCR register so that the LCR[7] bit = 0 (for normal operation).
3. Program the IER register to enable only the data-transmission related interrupts.
    - Enable the transmit threshold interrupt.
    - Enable the transmit underrun interrupt.
    - Disable the RX FIFO over threshold interrupt.
    - Disable the RX FIFO overrun error interrupt.
4. Program the transmit modulation carrier-frequency register (TVMDCFG) to select the desired carrier frequency and pulse duration.
5. Write a 1 to the ACREG[7] bit to enable data transmission.
6. Write a 1 to the MCR[3] bit to enable the selected IRQ channel.

This completes the programming of registers and now the device is ready for data transmission.

2.1.4.2 SHARP Infrared Data-Receive Mode

The following register programming steps are required for data receive in the SHARP infrared mode:

1. Program the infrared-configuration register (IRCFG). Select the SHARP infrared data-receive terminal to be either IRRVL (IRCFG[1] = 1) or IRRVH (IRCFG[1] = 0).
2. Program the RX demodulation configuration register (TVDMCFG). Select the desired incoming infrared data carrier frequency range.
3. Reset the RX FIFO. This includes:
    - Write to the LCR register to set the LCR[7] bit to a 1. The LCR[7] bit must be a 1 to be able to write to the FCR(5) bit.
    - Program the FCR to select 64-/16-bytes TX FIFO, non-DMA mode of operation, select the RX FIFO trigger level and to clear the TX FIFO.

- Write the LCR register so that bit LCR[7] = 0 (for normal operation).

4. Program the IER register to enable only the receive related interrupts.
5. Write a 1 to the ACREG[6] bit to enable data receive.
6. Write a 1 to the MCR[3] bit to enable the selected IRQ channel.

This completes the programming of registers for data receive and now the device is ready to receive data. The CPU receives an interrupt and reads the IIR to identify the source of the interrupt. If the source of the interrupt is IIR[0] the CPU goes into the threshold mode. See paragraph 2.1.1.2.1. If the source of the interrupt is IIR[4] there has been an overrun while data receiving and the CPU needs to service the overrun. See paragraph 2.1.1.2.3.

2.1.5 TV mode

The consumer TV remote-control (TV) mode supports data transmission between devices at a speed of up to 166.7 Kbps. Data can be transferred in a half-duplex mode. The following registers need to be programmed:

1. Program the interrupt configuration register (ICR) in order to:
   - Select the IRQ channel.
   - Select either totem-pole or open-drain output.
   - Select an active high or an active low for an interrupt when the totem-pole configuration is selected.
2. Program the prescaler (PRESC) register. The PRESC register can be programmed to a value so that:

$$\frac{X_{IN}}{PRESC} \leq 5 \text{ MHz} \tag{3}$$

Where

X_{IN} = Input clock frequency and

PRESC = PRESC register value

For a 48-MHz input clock, a PRESC register value of larger than 0A (hex) is recommended. The default hex value is 1A. This default value is retained if the desired baud rate is under 115.2 Kbps.

3. Program the auxiliary baud-rate divisor lower byte (ABDL) and the auxiliary baud-rate divisor higher byte (ABDH) registers to select the desired communication baud rate for TV mode. The baud rate is determined by:

$$\text{Baud rate} = \frac{X_{IN}}{PRESC \times \text{Divisor} \times 16} \tag{4}$$

$$\Rightarrow$$

$$\text{Divisor} = \frac{X_{IN}}{PRESC \times \text{BaudRate} \times 16}$$

The two 8-bit auxiliary baud-rate divisor registers store the divisor in a 16-bit binary format.

4. Program the mode-definition register (MDR) in order to:
   - Select the TV mode
   - Enable/disable the low-power mode. For normal operation, the low-power mode is enabled.

- Enable/disable the sleep mode. For normal operation, the sleep mode is disabled.
- Enable/disable the store and controlled data transmission. For normal operation, disable the SCT.
- Select the frame-closing method to set-EOT-bit (MDR[7]) to 1.

5. Program the TV configuration register (TVCFG) to:
   - Select the modulation mode (C_PLS, 6_PLS OR 8_PLS).
   - Enable/disable the receiver internal demodulation.
   - Select programmed T-period mode or over-sampling mode. For normal operation select programmed T-period mode.
   - Enable/disable the run-length coding mode.
   - Select the TX carrier-frequency range.
   - Select the RX carrier-frequency range.
   - Write a 0 to the RX-active bit (TVCFG[7]). When set to a 1, this bit sets the device to sample the infrared data even if the carrier frequency is out of range. This bit is normally used to disable the RX when required. The RX automatically enables when infrared is detected and the carrier frequency is in the programmed range.

2.1.5.1 TV Data-Transmission Mode

The following register programming steps are required for data transmission in the TV mode:

1. Program the PLR to select the TX FIFO trigger level
2. Reset the TX FIFO. This includes the following:
   - Write to the LCR register to set the LCR[7] bit to a 1. The LCR[7] bit must be a 1 to be able to write to the FCR(5) bit.
   - Program the FCR register to select 64-/16-bytes TX FIFO, non-DMA mode of operation, select the RX FIFO trigger level and to clear the TX FIFO.
   - Write the LCR register so that the LCR[7] bit = 0 (for normal operation).
3. Program the IER register to enable only the data-transmission related interrupts. This includes the following:
   - Enable the TX threshold interrupt.
   - Enable the TX underrun interrupt.
   - Disable the RX FIFO over threshold interrupt.
   - Disable the RX FIFO overrun error interrupt.
4. Program the TX modulation carrier-frequency register (TVMDCFG) to select the desired carrier frequency and its pulse duration.
5. Write a 1 to the ACREG[7] bit to enable data transmission.
6. Write a 1 to the MCR[3] bit to enable the selected IRQ channel.

This completes the programming of registers and the device is ready for data transmission. The TV data transmission only supports the set-EOT-bit method, thus the CPU must write a 1 to the ACREG[0] bit prior to writing the last byte of a frame to the TX FIFO. The CPU receives an interrupt and reads the IIR to identify

the source of the interrupt. If the source of the interrupt is IIR[1] (TX FIFO below the threshold interrupt), the CPU must:

- Disable the IER[1] bit
- Write data bytes to the TX FIFO. The maximum number of data bytes is limited to the number selected by the TX FIFO trigger level.
- Enable the IER[1] bit

If the source of the interrupt is a IIR[5] bit, an underrun has occurred while transmitting and the CPU must service the underrun. See paragraph 2.1.1.1.4.

2.1.5.2 TV Receive Mode

The following register programming steps are required for data receive in the TV mode:

1. Program the infrared configuration register (IRCFG). Select the TV infrared receive terminal to be either IRRVL (IRCFG[1] = 1) or IRRVH (IRCFG[1] = 0).
2. Program the RX demodulation configuration register (TVDMCFG). Select the desired incoming infrared data carrier frequency range.
3. Reset the RX FIFO. This includes the following:
   - Write to the LCR register to set the LCR[7] bit to a 1. The LCR[7] bit must be a 1 in order to be able to write to the FCR(5) bit.
   - Program the FCR to select 64-/16-bytes TX FIFO, non-DMA mode of operation, select the RX FIFO trigger level and to clear the TX FIFO.
   - Write the LCR register so that bit LCR[7] = 0 (for normal operation).
4. Program the IER register to enable only the receive related interrupts which include:
   - RX threshold interrupt (IER[0].
   - RX FIFO overrun (IER[3]).
   - Disable the TX related interrupts (TX FIFO below threshold interrupt and the TX underrun interrupt).
5. Write a 1 to the ACREG[6] bit to enable data receive.
6. Write a 1 to the MCR[3] bit to enable the selected IRQ channel.

This completes the programming of the registers for data receive and now the device is ready to receive data. The CPU receives an interrupt and reads the IIR to identify the source of the interrupt. If the source of the interrupt is IIR[0], the CPU goes to the threshold mode. See paragraph 2.1.1.2.1. If the source of the interrupt is IIR[4], an overrun has occurred while receiving data and the CPU needs to service the overrun. See paragraph 2.1.1.2.3.

7. To stop data receive after the End-Of-Frame flag (EOF) is detected by the CPU, write a 0 to the TVCFG[7] bit.

2.1.6 Universal Asynchronous Receiver Transmitter (UART) mode

The UART mode is the default mode of operation after power up and system reset. This mode uses a wired interface for serial communication with a remote device or a modem. The TIR2000 can operate in a full-duplex mode, i.e. data transmission and reception can take place simultaneously.

The TIR2000 in the UART mode works as a regular serial asynchronous communication controller that converts the parallel data received from the CPU or the DMA controller to serial data. It also converts the serial data received on the serial input terminal to parallel data.

2.1.6.1 Auto-flow Control (See Figure 2–4)

Auto-flow control is composed of auto-$\overline{\text{CTS}}$ and auto-$\overline{\text{RTS}}$. With auto-$\overline{\text{CTS}}$, $\overline{\text{CTS}}$ must be low before the transmit FIFO register can emit data (see Figure 2–4). With auto-$\overline{\text{RTS}}$, $\overline{\text{RTS}}$ becomes high when the receiver is empty or the threshold has not been reached. When $\overline{\text{RTS}}$ is connected to the $\overline{\text{CTS}}$, data transmission does not occur unless the receiver FIFO register has empty space. Overrun errors are eliminated when UART1 and UART2 are TIR2000 or other autoflow control devices with enabled flow control. When UART1 and UART2 are not autoflow control devices with enabled flow control an overrun error occurs if the transmit data exceeds the data receive FIFO-read capacity.

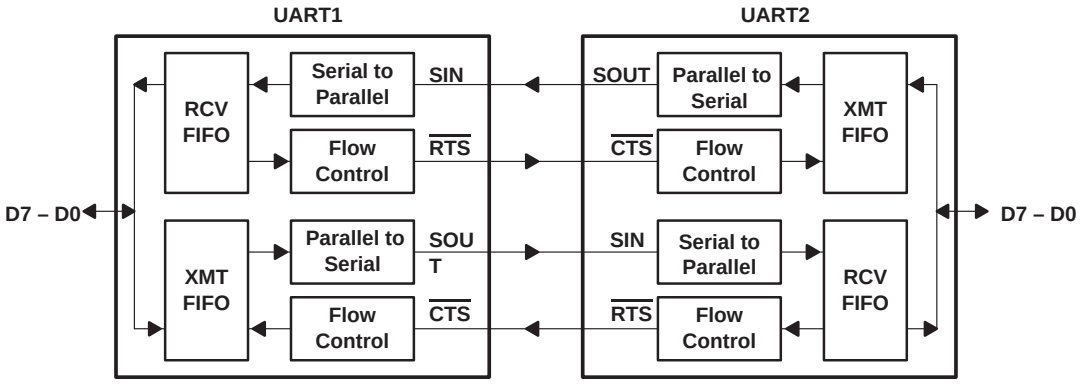


Figure 2–4. Example of Auto-flow Contol (Auto-$\overline{\text{RTS}}$ and Auto-$\overline{\text{CTS}}$)

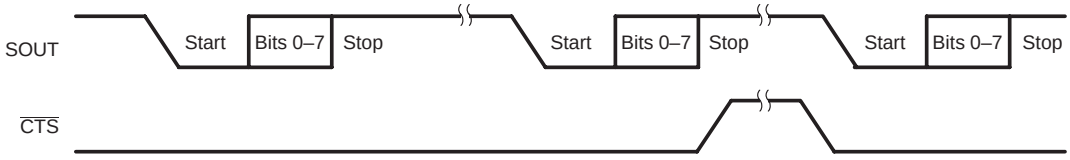
2.1.6.2 Auto-$\overline{\text{RTS}}$ (See Figure 2–4)

Auto-$\overline{\text{RTS}}$ data-flow control originates in the receiver timing and control block and is linked to the programmed RCV-FIFO trigger level. When the receiver FIFO trigger level reaches a trigger level of 1, 4, 8, or 14 in the 16-byte mode or 1, 16, 32, or 56 in the 64-byte mode, $\overline{\text{RTS}}$ is de-asserted. The sending UART may send an additional byte after the trigger level is reached (assuming the sending device has another byte to send) because it may not recognize the de-assertion of $\overline{\text{RTS}}$ until after it begins to send the additional byte. $\overline{\text{RTS}}$ is automatically reasserted when the RX FIFO register is emptied by reading the receiver-buffer register (RBR). The reassertion signals the sending UART to continue transmitting data.

2.1.6.3 Auto-$\overline{\text{CTS}}$ (See Figure 2–4)

The transmitter circuitry checks $\overline{\text{CTS}}$ before sending the next data byte. If $\overline{\text{CTS}}$ is low, the transmitter sends the next byte. To stop the transmitter from sending the following byte, $\overline{\text{CTS}}$ must be released before the middle of the last stop bit that is currently being sent. The auto-$\overline{\text{CTS}}$ function reduces interrupts to the host system. When the flow control is enabled, the $\overline{\text{CTS}}$ level changes and does not trigger host interrupts because the device automatically controls its own transmitter. Without auto-$\overline{\text{CTS}}$, the transmitter sends any data present in the transmit FIFO register and a receiver overrun error can result.

To enable the auto-$\overline{\text{RTS}}$ and auto-$\overline{\text{CTS}}$ modes of operation bit 5 of the modem-control register must be set to a 1.

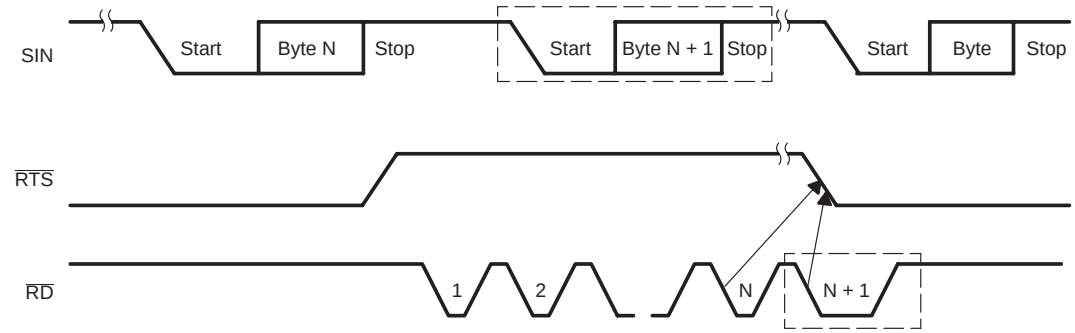


NOTES: A. When $\overline{\text{CTS}}$ is a low, the transmitter keeps sending serial data out.
B. When $\overline{\text{CTS}}$ transitions to a high before the middle of the last stop bit of the current byte, the transmitter finishes sending the current byte but it does not send the next byte.
C. When $\overline{\text{CTS}}$ transitions from a high to a low, the transmitter begins sending data again.

Figure 2–5. $\overline{\text{CTS}}$ Functional Timing Diagram

2.1.6.4 Enabling Auto-$\overline{\text{RTS}}$ and Auto-$\overline{\text{CTS}}$

The receiver FIFO trigger level can be set to 1, 4, 8, or 14 bytes for the 16-byte mode and 1, 16, 32, or 56 bytes for the 64-byte mode.



NOTES: A. N = receiver FIFO trigger level
B. The two blocks in dashed lines cover the case when an additional byte is sent as described in auto-CTS.

Figure 2–6. $\overline{\text{RTS}}$ Functional Timing, RCV-FIFO Trigger

2.1.6.4.1 Programming UART Related Registers

All UART related registers must be programmed for proper data communication. The following register programming steps are required for data communication:

1. Program the interrupt configuration register (ICR) in order to:
   - Select the IRQ channel.
   - Select either totem-pole or open-drain output.
   - Select an active high or an active low for an interrupt when the totem-pole configuration is selected.

2. Program the baud-rate divisor lower byte (DLL) register and the baud-rate divisor higher byte (DLM) register to select the desired communication baud rate for the UART mode. The baud rate is given by:

$$\text{Baud rate} = \frac{X_{IN}}{\text{PRESC} \times \text{Divisor} \times 16} \quad (5)$$

so the formula for the Divisor is:

$$\text{Divisor} = \frac{X_{IN}}{\text{PRESC} \times \text{BaudRate} \times 16}$$

Where

X_{IN} = Input clock frequency to the chip
PRESC = PRESC register value
Baud Rate = Desired baud rate

The two 8-bit baud-rate divisor registers (DLL and DLM) store the divisor in a 16-bit binary format. The device can be changed to the UART mode from any other mode of operation by writing to the DLL or the DLM registers. This changes the MDR register MDR[2–0] bits to the UART mode automatically (known as the fall-back mechanism). The PRESC register also automatically changes to the default value.

3. Program the prescaler (PRESC) register so that the prescaler output is approximately 1.843 MHz. For an input frequency of 48 MHz, use the default PRESC value (26).
4. Program the mode-definition register (MDR) in order to:
   - Select the UART mode.
   - Enable/disable the low-power mode. For normal operation, the low-power mode is enabled.
   - Enable/disable the sleep mode. For normal operation, the sleep mode is disabled.
5. Program the line-control register (LCR) in order to:
   - Select the word length.
   - Select the number of stop bits.
   - Enable/disable parity generation.
   - Select even/odd parity.
   - Enable/disable stick parity.
   - Enable/disable the break control.
6. Program the LCR to select the TX FIFO trigger level.
7. Reset the TX FIFO. This includes the following:
   - Write to the LCR register to set the LCR[7] bit to a 1. The LCR[7] bit must be a 1 in order to be able to write to the FCR(5) bit.
   - Program the FCR register to select 64-/16-bytes TX FIFO register, non-DMA mode of operation, select the RX FIFO trigger level and to clear the TX FIFO.
   - Write the LCR register so that the LCR[7] bit = 0 (for normal operation).
8. Program the IER register to enable interrupts.
9. Write a 1 to the MCR[3] bit to enable the selected IRQ channel.

2.1.7 DMA Operation

The DMA mode of data transfer is used to achieve faster data transfer when the device is operating in the FIR or in the MIR modes. In a multi-application environment, where multiple applications are running at the same time, the DMA mode should be used to keep up with the high rate of data transfer without having underrun during data transmission or overrun during data receive. Frames can be transmitted and received back-to-back. The device must be programmed for data transmission and data receive. The DCSR register is programmed to select and enable the DMA channel for data transmit and data receive.

2.1.7.1 DMA Data-Transmit Mode

Packets of various sizes can be transmitted. Packets, smaller than 2k bytes, can be properly terminated through the TC signal from the DMA controller or through the (TXFLH, TXFLL) registers value. Packets, larger than 2k bytes, can be automatically fragmented into equal-sized smaller frames (decided by the TXFLH and TXFLL registers value) and properly transmitted. The last portion of the packet, which may be smaller than the (TXFLH, TXFLL) register value, is terminated by the TC signal. The DMA controller asserts the TC signal when it sends the last byte of that packet. When the DMA asserts the TC signal, an interrupt IIR[6] bit is generated to inform the CPU that the DMA data transmission is complete.

The DMA controller uses the Demand Transfer mode to transfer data from memory to the TXFIFO. The peripheral device asserts the DRQ signal when the number of bytes in the TXFIFO falls below the set threshold level and de-asserts the DRQ when the TXFIFO becomes almost full.

2.1.7.2 DMA Data-Receive Mode

The DMA controller controls the data reception of back-to-back frames. The back-to-back frames are transferred to memory and the status of each received frame is stored in the Status FIFO which can hold up to 8 entries. Each entry in the Status FIFO corresponds to one received frame. Each entry stores the length of each received frame and the error-status of that frame. The CPU reads the Status FIFO entries to locate the frame boundaries and status of individual frames inside memory.

The Status FIFO has 4 interrupt levels, which are 1, 4, 7, and 8. In a data receive transaction, the receiver peripheral expects to receive 1 to 7 frames. When the number of received frames becomes equal or greater than the set threshold value in the Status FIFO, the peripheral device generates an interrupt IIR[4] bit. After the CPU receives the Status FIFO interrupt (IIR[4] bit), it reads the Status FIFO Register High (SFREGH) and then the Status FIFO Register Low (SFREGL) to determine the length of the frame. The SFLSR register is then read to determine the error-status of that frame. The CPU checks the content of the SFLSR[4] bit which is set to a 1 when the Status FIFO becomes empty. If this bit is found to be a 1, the CPU should stop reading the SFREFH, SFREGL, and SFLSR registers.

The time-out interrupt is set at 1 ms. This is useful when the number of frames in a data receive transaction is less than the set Status FIFO threshold level. A time-out interrupt (IIR[2] bit) will be generated when the inter-frame gap between back-to-back frames is 1 ms. When the CPU receives a Status FIFO time-out interrupt, it should read the LSR register (LSR[1] bit) to find out the empty status of the Status FIFO. If the Status FIFO is empty, the CPU disregards the interrupt. If the Status FIFO is not empty, the CPU reacts the same as it does after receiving a Status FIFO threshold interrupt.

The DMA controller uses the demand transfer mode to transfer data from the RXFIFO to memory. The peripheral device asserts the DRQ signal when the number of bytes in the RXFIFO is equal to or greater than the set threshold level or when the end of a receive frame is detected by the RX state machine. The DRQ is de-asserted when the RXFIFO becomes empty. An overrun in the RXFIFO in the DMA mode is handled differently than in the programmed I/O mode. When an overrun occurs in the RXFIFO, during any frame, data reception is terminated and the RX state machine waits for the next frame. It continues to receive the further frames. The overrun error (OE bit) is set to a 1 in the SFLSR register for the frame in which a RXFIFO overrun has occurred.

(This page has been left blank intentionally.)

3 Register Definitions for the UART, SIR and SHARP Modes

3.1 Receiver-Buffer Register (RBR) – (read only)

The receiver buffer register is used to access the receiver-holding register when the RX FIFO register is disabled, or the RX FIFO when the RX FIFO is enabled. Address: 0 (hex) – LCR[7] bit should be a 0.

3.2 Transmitter Holding Register (THR) – (write only)

The transmitter holding register is used to access the transmitter (TX) holding register when the TX FIFO is disabled, or the TX FIFO when the TX FIFO is enabled. Address: 0 (hex) – LCR[7] bit should be a 0.

3.3 Interrupt Enable Register (IER) – (read and write)

Address: 1 (hex) – LCR[7] bit should be a 0.

Bit definitions are as follows:

BIT NO.	DEFAULT	DESCRIPTIONS
0	0	Receiveddata-available interrupt enable (1 = enable, 0 = disable). When set, bit 0 enables an interrupt when the receiver FIFO is equal to or above its threshold level.
1	0	Transmitter holding-register-empty interrupt enable.
2	0	Receiver line-status interrupt enable.
3	0	Modem-status interrupt enable.
4	0	Not used
5	0	Transmitter underrun interrupt enable
6	0	0
7	0	0

3.4 Interrupt Identification Register (IIR) - (read only)

Address: 2 (Hex)

Bit Definitions:

BIT NO.	DEFAULT	DESCRIPTION
0	1	Interrupt pending
1	0	Interrupt ID0
2	0	Interrupt ID1
3	0	Interrupt ID2
4	0	Reserved (always 0)
5	0	FIFO enabled 1
6	0	FIFO enabled 2
7	0	FIFO enabled 3 (bit 7 bit 6 bit 5) = 000 FIFO disabled = 110 16-byte FIFO = 111 64-byte FIFO

INTERRUPT ID REGISTER				PRIORITY LEVEL	INTERRUPT TYPE	INTERRUPT SOURCE	INTERRUPT RESET METHOD
BIT 3	BIT 2	BIT 1	BIT 0				
0	0	0	1	None	None	None	None
0	1	1	0	1	Receiver line status	Overrun error, parity error, framing error or break interrupt	Read the line-status register
0	1	0	0	2	Received data available	Receiver data available or trigger level reached in the FIFO mode	Read the receiver buffer register
1	1	0	0	2	Character time-out indication	No characters have been removed from or input to the receiver FIFO during the last 4 character times, and there is at least 1 character in it during this time.	Read the receiver buffer register
0	0	1	0	3	Transmitter holding-register empty	Transmitter holding register is empty	Read the interrupt identification register or write to the transmitter holding register.
0	0	0	0	4	Modem status	Clear to send, data-set ready, ring indicator, or data-carrier detect	Read the modem-status register

3.5 FIFO-Control Register (FCR) – (write only)

Address: 2 (Hex)

Bit definitions:

BIT NO.	DEFAULT	DESCRIPTIONS
0	0	FIFO enable. When set to 1, bit 0 enables both the transmitter and receiver FIFOs. Bit 0 must be 1 when setting other FCR bits. Changing bit 0 clears the FIFO registers.
1	0	Receiver FIFO reset. When set to 1, bit 1 clears all bytes in the receiver FIFO and resets its counter to 0. The shift register is not cleared. A logic 1 written to this bit is self clearing.
2	0	Transmitter FIFO reset. When set to 1, bit 2 clears all types in the transmitter FIFO and resets the counter to 0. The shift register is not cleared. A logic 1 written to bit 2 is self clearing.
3	0	DMA mode select. Set to 1 to enable DMA or 0 to disable DMA.
4	0	Reserved
5	0	FIFO size selection. When set to 1, 64-byte mode is selected. When set to 0, 16-byte mode is selected. A write instruction to FCR5 is ignored when line-control register (LCR[7]) is 0. LCR[7] needs to be a 0 for normal operation.
6	0	Receiver FIFO trigger level selection (LSB)
7	0	Receiver FIFO trigger level selection (MSB)

Bit 6 and Bit 7 trigger levels are as follows:

BIT 7	BIT 6	16-BYTE RECEIVER FIFO TRIGGER LEVEL (BYTES)	64-BYTE RECEIVER FIFO TRIGGER LEVEL (BYTES)
0	0	01	01
0	1	04	16
1	0	08	32
1	1	14	56

3.6 Line Control Register (LCR) - (read and write)

Address: 3 (Hex)

Bit definitions:

BIT NO.	DEFAULT	DESCRIPTION
0	0	Word length select bit 0 (WLS0)
1	0	Word length select bit 1 (WLS1)

Bits 0 and bit 1 specify the number of bits in each transmitted or received serial character.

BIT 1	BIT 0	WORD LENGTH
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

2 0 Number of stop bits. Bit 2 specifies either 1, 1.5, or 2 stop bits in each transmitted character. The number of stop bits generated is defined as in the following table. The receiver clocks only the first stop bit, regardless of the number of stop bits selected.

Bit 2 bit specified word lengths and number of stop bits generated are as follows:

BIT 2	WORD LENGTH SPECIFIED BY BITS 0 & 1	NUMBER OF STOP BIT GENERATED
0	Any word length	1
1	5 bits	1.5
1	6 bits	2
1	7 bits	2
1	8 bits	2

3 0 Parity enable bit. When bit 3 is a 1, a parity bit is generated in data transmitted between the last data word bit and the first stop bit. In the receiver, the parity bit is checked when bit 3 is set. When bit 3 is a 0, no parity is generated or checked.

4 0 Even parity select bit. When bit 3 is set, a logic 1 in bit 4 produces even parity and a logic 0 in bit 4 produces odd parity.

5 0 Stick-parity bit. When bits 3, 4 and 5 are a 1, the parity bit is transmitted and checked as a logic 0. When bits 3 and 5 are a 1 and bit 4 is a 0, the parity bit is transmitted and checked as a logic 1. When bit 5 is a 0, stick parity is disabled.

6 0 Break control bit. Bit 6 is set to a 1 to force a break condition; i.e., a condition where the serial output (SOUT) is forced to spacing (0) state. When bit 6 is set to 0, the break condition is disabled. Bit 6 only affects the SOUT.

7 0 Divisor-latch-access bit (DLAB). Bit 7 must be set to a 1 to access the divisor latches of the baud rate generator or bit 5 of the FIFO control register (FCR). Bit 7 must be set to 0 to access the receiver buffer register, the transmitter holding register, or the interrupt enable register.

3.7 Modem Control Register (MCR) - (read and write)

Address: 4 (Hex)

Bit Definitions:

BIT NO.	DESCRIPTION
0	Data terminal ready (DTR). Bit 0 controls $\overline{\text{DTR}}$.
1	Request to send (RTS). Bit 1 controls $\overline{\text{RTS}}$.
2	Reserved
3	Enable IRQ. When set to a 1, the selected IRQ channel is enabled.
4	Loop back mode enable. When set to 1, bit 4 enables the loop back mode, i.e., it provides a local loop-back feature for diagnostic testing of the UART.
5	Flow-control enable (AFE).
6	Reserved
7	Reserved

3.8 Line Status Register (LSR) - (read only)

Address: 5 (Hex)

Bit Definitions:

BIT NO.	DESCRIPTION
0	Data-ready (DR) indicator for the receiver. Bit 0 is set to a 1 when a complete incoming character is received and transferred into the receiver buffer register or the FIFO. Bit 0 is reset to 0 by reading all of the data in the receiver buffer register or the FIFO register.
1	Overrun-error (OE) indicator. When bit 1 is set to a 1, it indicates that before the character in the receiver buffer register was read, it was overwritten by the next character transferred into the register. Bit 1 is reset every time the CPU reads the contents of the line-status register. When the FIFO-mode data continues to fill the FIFO beyond the trigger level, an overrun error occurs only after the FIFO is full and the next character has been completely received in the shift register.
2	Parity-error (PE) indicator. When bit 2 is set to 1, it indicates that the parity of the received data character does not match the parity selected in the line-control register (bit 4). Bit 2 is reset every time the CPU reads the LSR. In the FIFO mode, this error is associated with the particular character in the FIFO to which it applies. This error is revealed to the CPU when its associated character is at the top of the FIFO.
3	Framing-error indicator. When bit 3 is set to a 1, it indicates that the received character does not have a valid (1) stop bit. Bit 3 is reset every time the CPU read the contents of the LSR. In the FIFO mode, this error is associated with the particular character in the FIFO to which it applies. This error is revealed to the CPU when its associated character is at the top of the FIFO. The UART tries to re-synchronize after an framing error (FE). To accomplish this, it is assumed that a FE is caused by the next start bit. The UART samples this start bit twice and then accepts the input data.
4	Break-interrupt (BI) indicator. When bit 4 is set to a 1, it indicates that the received data input was held low for longer than a full-word transmission time. A full-word transmission time is defined as the total time to transmit the start, data, parity, and stop bits. Bit 4 is reset every time the CPU reads the contents of the LSR. In the FIFO mode, this error is associated with the particular character in the FIFO to which it applies. This error is revealed to the CPU when its associated character is at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character transfer is enabled after SIN goes to the marking state for at least two RCLK samples and then receives the next valid start bit.
5	Transmitter holding register empty (THRE) indicator. Bit 5 is set to 1 when the transmitter holding register is empty, indicating that the UART is ready to accept a new character. If the THRE interrupt is enabled when bit 5 is a 1, an interrupt is generated. Bit 5 is set to 0 when the contents of the transmitter holding register are transferred to the transmitter shift register. Bit 5 is reset to 0 concurrent with the loading of the transmitter holding register by the CPU. In the FIFO mode, bit 5 is set when the transmit FIFO is empty; it is cleared when at least one byte is written to the transmit FIFO.
6	Transmitter empty (TEMT) indicator. Bit 6 is set to 1 when the transmitter holding register and the transmitter shift register are both empty. In the FIFO mode, bit 6 is set to 1 when the transmitter FIFO and shift register are both empty.
7	Error in receiver FIFO (ER_RCV_FIFO). Bit 7 is set to a 1 if there is at least 1 framing-error, parity-error or break-indication in the receiver FIFO. If the FIFO is disabled, this bit is always 0. Bit 7 is cleared when the CPU reads the LSR if there are no other errors in the receiver FIFO.

3.9 Modem Status Register (MSR) - (read only)

Address: 6 (Hex)

Bit Definitions:

BIT NO.	DESCRIPTION
0	Δ clear-to-send (CTS) indicator.
1	Δ data-set-ready (DSR) indicator.
2	Trailing edge ring-indicator (RI) detector.
3	Δ data-carrier-detect (DCD) indicator.
4	CTS. Complement of $\overline{\text{CTS}}$.
5	DSR. Complement of $\overline{\text{DSR}}$.
6	RI. Complement of $\overline{\text{RI}}$.
7	DCD. Complement of $\overline{\text{DCD}}$.

3.10 Scratch Register (SCR) - (read and write)

Address: 7 (Hex)

SIR/SHARP Mode

Bit definitions:

7	6	5	4	3	2	1	0
SCR[7]	SCR[6]	SCR[5]	SCR[4]	SCR[3]	SCR[2]	SCR[1]	SCR[0]

3.11 Baud-Rate Divisor Latch LSB (DLL) - (read and write)

Address: 0 (Hex) - When LCR[7] = 1

Bit Definitions:

7	6	5	4	3	2	1	0
DLL[7]	DLL[6]	DLL[5]	DLL[4]	DLL[3]	DLL[2]	DLL[1]	DLL[0]

3.12 Baud-Rate Divisor Latch MSB (DLM) - (write only)

Address: 1 (Hex) - When LCR[7] = 1

Bit Definition:

7	6	5	4	3	2	1	0
DLM[7]	DLM[6]	DLM[5]	DLM[4]	DLM[3]	DLM[2]	DLM[1]	DLM[0]

4 Register Definitions for the MIR, FIR and TV Modes

4.1 Receiver Buffer Register (RBR) – (read only)

Address: 0 (Hex)

LCR[7] bit must be a 0 to read this register. This register is used to read data from the RX FIFO register.

4.2 Transmitter Holding Register (THR) – (write only)

Address: 0 (Hex)

LCR[7] bit must be a 0 to read this register. This register is used to read data from the TX FIFO register.

4.3 Interrupt Enable Register (IER) – (read and write)

Address: 1 (Hex) - LCR[7] bit must be a 0 to access this register.

MIR/FIR (Programmed I/O) Mode

Bit definition:

BIT NO.	DEFAULT	DESCRIPTIONS
0	0	Received–data–available interrupt enable (1 = enable, 0 = disable). If bit 0 is a 1, it enables an interrupt when the receiver FIFO is equal to or above the set threshold level.
1	0	Transmitter FIFO below threshold level interrupt enable. If bit 1 is a 1, it enables an interrupt when the transmitter FIFO is below the set threshold level.
2	0	Last byte from RX FIFO enable. (1 = enable, 0 = disable) If bit 2 is a 1, it enables an interrupt when the CPU reads the last byte of a frame from the RX FIFO.
3	0	RX FIFO overrun interrupt enable.
4	0	Write 0
5	0	Transmitter underrun interrupt enable.
6	0	Write 0
7	0	Received end–of–frame interrupt enable.

MIR/FIR (DMA) Mode

Bit definition:

BIT NO.	DEFAULT	DESCRIPTIONS
0	0	Write 0
1	0	Write 0
2	0	Status FIFO time-out interrupt enable. When set to a 1, an interrupt is generated when the interframe gap between successive frames is 1 ms.
3	0	Write 0
4	0	Status FIFO threshold interrupt enable. When set to a 1, bit 4 enables the Status FIFO threshold interrupt, i.e. When the Status FIFO level is equal to or above its threshold level, an interrupt is generated.
5	0	Transmitter underrun interrupt enable.
6	0	DMA terminal count interrupt enable.
7	0	Write 0

Consumer Remote Mode

Bit definition:

BIT NO.	DEFAULT	DESCRIPTIONS
0	0	Received-data-available interrupt enable (1 = enable, 0 = disable). If bit 0 is set, it enables an interrupt when the receiver FIFO is equal to or above its threshold level.
1	0	Transmitter FIFO below threshold level interrupt enable. If bit 1 is set, it enables an interrupt when the transmitter FIFO is below its threshold level.
2	0	write 0
3	0	RX FIFO overrun interrupt enable
4	0	write 0
5	0	Transmitter underrun interrupt enable.
6	0	write 0
7	0	write 0

4.4 Interrupt Identification Register (IIR) – (read only)

Address: 2 (Hex)

MIR/ FIR (Programmed I/O) Mode

IIR_IR

Bit No	7	6	5	4	3	2	1	0
Definition	EOF detected	Not used	TX underrun	Not used	RX FIFO overrun	Last byte read from FIFO	TX FIFO below threshold	RX FIFO over threshold
Default	0	0	0	0	0	0	1	0

Bit definition:

BIT NO.	DESCRIPTIONS
0	Receiver interrupt pending. Bit 0 is set to a 1 when the receiver FIFO level is equal to or above its threshold level.
1	Transmitter interrupt pending. Bit 1 is set to a 1 when the transmitter FIFO level is below its threshold level.
2	Bit 2 is set to a 1 when the CPU reads the last byte of a frame from the receiver FIFO. Bit 2 is cleared when it is read. The CPU uses bit 2 along with bit 7 to locate the end of the frame.
3	Receiver FIFO overrun interrupt. Bit 3 is set to a 1 when an overrun occurs in the RX FIFO. Overrun occurs when the CPU cannot read data fast enough from the RX FIFO and there is no empty space available in the RX FIFO register to store frame data. If an overrun occurs, the CPU has to service the overrun before the TIR2000 can receive data further.
4	Not used
5	Transmit underrun interrupt pending. Bit 5 is set to a 1 when an underrun occurs in transmit FIFO. Bit 5 is cleared by servicing the underrun.
6	Not used
7	Received end-of-frame interrupt pending. Bit 7 is set to a 1 when the receiver detects the last byte of a frame. This is different from bit 2. Bit 2 is set to a 1 when the CPU reads the last byte of a frame from the FIFO. Bit 7 is set prior to bit 2. Bit 7 is cleared to a 0 when it is read.

MIR/FIR (DMA) Mode

Bit No	7	6	5	4	3	2	1	0
Definition	Not used	DMA TC asserted	TX underrun	Status FIFO threshold	Not used	Status FIFO time-out	Not used	Not used
Default	0	0	0	0	0	0	0	0

Bit definition:

BIT NO.	DESCRIPTIONS
0	Not used
1	Not used
2	Status FIFO time-out interrupt pending. Bit 2 is cleared to 0 by reading the Status FIFO.
3	Not used
4	Status FIFO threshold interrupt pending. Bit 4 is cleared to 0 by reading the Status FIFO.
5	Transmit underrun pending. Bit 5 is set to a 1 when an underrun occurs in the data transmit FIFO. Bit 5 is cleared to 0 by servicing the underrun.
6	DMA terminal count occurred interrupt pending. In the DMA mode, bit 6 is set to 1 when the terminal count (TC) is asserted. Bit 6 is cleared to 0 by reading the Status FIFO.
7	Not used

TV Mode

IIR_TV

Bit No	7	6	5	4	3	2	1	0
Definition	0	0	TX underrun error	0	RX FIFO overrun error	0	TX FIFO below threshold	RX FIFO over threshold
Default	0	0	0	0	0	0	1	0

Bit definition:

BIT NO.	DESCRIPTIONS
0	Receiver interrupt pending. Bit 0 is set to a 1 when the receiver FIFO level is equal to or above its threshold level.
1	Transmitter interrupt pending. Bit 1 is set to a one when the transmitter FIFO level is below its threshold level.
2	always 0
3	Receiver FIFO overrun interrupt. Bit 3 is set to a one when an overrun occurs in the data RX FIFO.
4	always 0
5	Transmit underrun interrupt pending. Bit 5 is set to a one when an underrun occurs in the data transmit FIFO.
6	always 0
7	always 0

4.5 FIFO Control Register (FCR) – (write only)

Address: 2 (Hex)

MIR, FIR And Consumer Mode

Bit definition:

BIT NO.	DEFAULT	DESCRIPTIONS
0	0	FIFO enable. When set to 1, Bit 0 enables both the transmitter and the receiver FIFOs. Bit 0 must be a 1 when setting other FCR bits. Changing bit 0 clears the FIFO.
1	0	Receiver FIFO reset. When set to 1, bit 1 clears all bytes in the receiver FIFO and resets its counter to 0. The shift register in the TV mode is not cleared. A 1 written to bit 1 is self clearing.
2	0	Transmitter FIFO reset. When set to 1, bit 2 clears all bytes in the transmitter FIFO and reset its counter to 0. The shift register in the TV mode is not cleared. A 1 written to bit 2 is self clearing.
3	0	DMA mode select. DMA mode is selected when bit 3 is set to 1.
4	0	Not used
5	0	When set to 1, 64–byte transmitter and 64–byte receiver FIFO are selected. When set to 0, the 16–byte transmitter and the 16–byte receiver FIFO are selected. LCR[7] bit must be a 1 in order to write to bit 5. After writing to bit 5, the LCR[7] bit should be changed to a 0.
6	0	Receiver FIFO trigger level selection.
7	0	Receiver FIFO trigger level selection.

Bit 6 and bit 7 trigger levels are as follows:

BIT 7	BIT 6	16–BYTE RECEIVER FIFO TRIGGER LEVEL (BYTES)	64–BYTE RECEIVER FIFO TRIGGER LEVEL (BYTES)
0	0	01	01
0	1	04	16
1	0	08	32
1	1	14	56

4.6 Line Status Register (LSR) – (read only)

Address: 5 (Hex)

MIR, FIR Mode

LSR

Bit No	7	6	5	4	3	2	1	0
Definition	TX empty	Status FIFO full	Received last byte from RX FIFO	Frame length error	PHY error	CRC error	Status FIFO empty	RX FIFO empty
Default	1	0	0	0	0	0	1	1

Bit definition:

BIT NO.	DESCRIPTIONS
0	Receiver FIFO empty. Bit 0 indicates the empty status of RX FIFO register. When the RX FIFO becomes empty bit 0 becomes a 1 and when RX FIFO is not empty, it becomes a 0.
1	Port 1 is set to a 1 when the Status FIFO is empty.
2	CRC error. When a bad CRC is detected on data receive, bit 2 is set to 1. Bit 2 is cleared to 0 when the LSR register is read.
3	PHY error. In FIR mode, bit 3 is set to a 1 when an illegal symbol is received during reception. In the MIR mode, bit 3 is set to a 1 when an abort pattern (at least 7 consecutive 1's) is received during reception. Bit 3 is cleared to a 0 when the LSR register is read.
4	Frame–too–long. Bit 4 is set to 1 when a frame exceeding the maximum length (set by RXFLH register and the RXFLL register) is received. When this error is detected, current frame reception is terminated. Reception is stopped until the next BOF is detected. Bit 4 is cleared to a 0 when the LSR register is read.
5	Received last byte from FIFO. Bit 5 is set to a 1 when the CPU reads the last byte of a frame. Bit 5 can be used for software polling to determine the frame boundary. It is cleared to a 0 when the CPU reads the LSR register.
6	Bit 6 is set to a 1 when the Status FIFO is full.
7	Transmitter empty. When TX FIFO is empty and the transmitter front-end is idle, this bit is set to a 1.

LSR in TV Mode

LSR_TV

Bit No.	7	[6–2]	1	0
Description	TX FIFO empty	Not relevant	TV_IR pulse detected	RX FIFO register empty
Default	1	Always 0	0	1

Bit definition:

BIT NO	DESCRIPTION
0	When bit 0 is a 1, the RX FIFO register is empty. A 0 in bit 0 indicates the RX FIFO register is not empty.
1	When bit 1 is a 1, the valid IR pulse (carrier frequency within programmed range) has been detected. Bit 1 is reset to a 0 when read.
[6–2]	Not relevant
7	When bit 7 is a 1, the TX FIFO register is empty. A 0 in Bit 7 indicates the TX FIFO register is not empty.

4.7 Auxiliary Baud–Rate Divisor Lower Register (ABDL) – (write only)

Address: E (Hex) – Bank 3

Baud-rate divisor lower 8–bit for TV, SIR and SHARP mode

Bit No.	7	6	5	4	3	2	1	0
Description	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Default	0	0	0	1	1	0	0	0

4.8 Auxiliary Baud–Rate Divisor Higher Register (ABDH) – (write only)

Address: F (Hex) – Bank 3

Baud-rate divisor higher 8–bit for the TV, SIR and SHARP modes.

Bit No.	7	6	5	4	3	2	1	0
Description	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Default	0	0	0	0	0	0	0	0

4.9 Mode Definition Register (MDR) – (read and write)

Address: 8 (Hex)

MDR

Bit No.	7	6	5	4	3	2	1	0
Description	Frame end mode select	2-µs Ir pulse select method	SCT mode enable	Low power mode enable	Sleep mode enable	Mode select		
Default	0	0	0	0	0	000		

Bit definition:

BIT NO.	DESCRIPTIONS
0–2	Mode of operation. Bits 0–2 select the mode of operation as follows: 000 UART 001 SIR 010 SHARP 011 MIR 100 FIR 101 TV 110 Reserved 111 Reserved
3	Sleep-mode enable. When bit 3 is set to 1, sleep mode is enabled. When set to 1 and other conditions are satisfied, the device goes into the sleep mode. See paragraph 4.9.1.
4	Low power mode enable. When bit 4 is set to 1, bit 4 enables the low power mode. When set to 1 and other conditions are satisfied, the device goes to low-power mode. See paragraph 4.9.2.
5	Store and controlled transmission enable. When bit 5 is set to 1, store and controlled transmission mode is enabled.
6	2-µs Ir pulse select method. Bit 6 selects the 2-µs Ir pulse select method. Set 1 to select 2-µs Ir pulse.
7	When set to a 1, bit 7 selects the Set-EOT-bit method. When a 0, bit 7 selects the frame-length method. See paragraph 2.1.1.1

4.9.1 Sleep Mode

When the MDR[3] bit is a 1 and other conditions are satisfied, the device goes into the sleep mode. In sleep mode the oscillator is shut off and there is no clock running inside the TIR2000. This feature is useful for saving power consumption. The scope of sleep mode depends on the mode of operation as listed below:

UART mode: The device is always awake when there is a byte in the transmitter, activity on SIN, or when the device is in the loopback mode. The device is also awake when either delta-CTS, delta-DSR, delta-DCD, or TERI = 1.

MIR mode: Sleep mode is applicable only during data transmission. The CPU must disable the sleep mode during data reception by writing a 0 to the MDR[3] bit. When the transmitting peripheral device is idle and the transmitter FIFO is empty the device goes into the sleep mode. When the CPU writes data to the TX FIFO register the oscillator is turned on and the clock becomes alive. However it takes 1ms for the oscillator to be stable and the clock to be valid.

FIR mode**:** Sleep mode is applicable both during data transmission and data reception. The device goes into the sleep mode when the transmitting peripheral device is idle, the TX FIFO register is empty and the receiver is also idle. The clock becomes alive when the CPU writes data to the TX FIFO register or the receiver starts receiving data. However, it takes 1ms for the oscillator to be stable and the clock to be valid.

SIR and SHARP IR mode: The scope is similar to UART mode and operates for both data transmission and data receive.

TV mode: Sleep mode is applicable for both data transmission and data receive.

4.9.2 Low-Power Mode

The mechanism of low-power mode is the same as the sleep mode except that the low-power mode is controlled by the MDR[4] bit and the oscillator keeps running. In low-power mode, the oscillator is not shut off; only the clock inside the device is disabled. In sleep mode it takes 1 ms to change from sleep mode to normal mode, whereas the device can change from low-power mode to normal mode immediately.

4.10 Configuration Control Register (CCR) – (write only)

Address 9 (Hex)

CCR

Bit No.	[7–6]	[5–0]
Description	BANK SELECT	Not used
Default	00	

There are four register banks with addresses from A to F. To write to/read from a register in any bank, the corresponding bank must be selected by writing to the CCR bits[7–6].

BITS CCR[7–6]	BANK SELECTION
00	Bank 0
01	Bank 1
10	Bank 2
11	Bank 3

4.11 BANK 0 REGISTERS

4.11.1 Transmit Frame-Length Register Low (TXFLL) – (write only)

Address: A (Hex) – Bank 0

Bit No.	7	6	5	4	3	2	1	0
Description	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Default	0	0	0	0	0	0	0	0

4.11.2 Transmit Frame-Length Register high (TXFLH) – (write only)

Address: B (Hex) – Bank 0

Bit No.	[7–6]	5	4	3	2	1	0
Description	Not used	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Default		0	0	0	0	0	0

The TXFLL register along with the TXFLH register stores the value for the number of bytes of a data transmission frame. The TXFLL register stores the lower 8 bits and the TXFLH register stores the upper bits.

4.11.3 Received Frame-Length Register Low(RXFLL) – (write only)

Address: C (Hex) – Bank 0.

Bit No.	7	6	5	4	3	2	1	0
Description	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Default	0	0	0	0	0	0	0	0

4.11.4 Received Frame-Length Register High (RXFLH) – (write only)

Address: D (Hex) – Bank 0.

Bit No.	[7–6]	5	4	3	2	1	0
Description	Not used	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Default		0	0	0	0	0	0

While receiving data, the maximum length of a frame is limited to the value, written to the RXFLL and the RXFLH registers. The RXFLL register stores the lower 8 bits and the RXFLH register stores the upper bits. Any frame greater then the set maximum value will be reported as a frame-length-exceed error. If n is the intended receive-frame maximum length, then program the (RXFLL, RXFLH) registers to be (n + 3) in the MIR mode and (n+5) in the FIR mode of operation.

4.11.5 Preamble Length Register (PLR) – (write only)

Address: E (Hex) – Bank 0.

PLR

Bit No.	[7–6]	[5–4]	[3–0]
Description	Preamble length	TX FIFO trigger	Start flags
Default	00	01	0010

Bit[3–0]: Number of start flags (relevant only in the MIR mode). The number of start flags to be transmitted at the beginning of a frame is equal to the PLR[3–0] value. The minimum value is 2.

Bit[5–4]: Data Transmission FIFO trigger level selection or empty space available selection.

BIT [5–4]		16-BYTE RECEIVER FIFO EMPTY SPACE AVAILABLE (BYTES)	64-BYTE RECEIVER FIFO. EMPTY SPACE AVAILABLE (BYTES)
0	0	Not used	Not used
0	1	12	48
1	0	08	32
1	1	2	8

Example: Bit[5–4] are set to 01 and the TX FIFO register is configured for 64 bytes. When the available empty space in the TX FIFO register increases to 48 bytes, the device interrupts the CPU with the IIR[1] bit. The CPU can safely write up to 48 bytes to the TX FIFO without the possibility of overwriting the TX FIFO.

Bit[7–6]: These two bits decide the number of preambles (a fixed pattern of 16 bits) to be transmitted at the beginning of each frame. The default value of PLR[7–6] = 00 which is equal to 16 preambles.

BIT [7–6]	NO. OF PREAMBLES
00	16
01	4
10	8
11	32

4.11.6 Auxiliary Control Register (ACREG) – (read and write)

Address: F (Hex)

Bit No.	7	6	5	4	3	2	1	0
Description	TX enable	RX enable	Short time-out	MIR half mode	Send Ir pulse	Deferred TX start	Frame abort	EOT bit
Default	0	0	0	0	0	0	0	0

Bit definition:

BIT NO.	DESCRIPTIONS
0	End of transmission (EOT) bit. The CPU writes a 1 to bit 0 just before it writes the last byte to the TX FIFO register. Bit 0 automatically gets cleared to a 0 in the next CPU write instruction to the device. See paragraph 2.1.1.1.
1	Frame abort. The CPU can intentionally abort data transmission of a frame by writing a 1 to bit 1. The default value is a 0. Neither the end flag nor the CRC bits are appended to the frame. The receiver will find the frame with the abort pattern (1111_111) in the MIR mode and a Phy–error in the FIR mode. The CPU must reset the TX FIFO and reset the ACREG[1] bit by writing a 0 to bit 1 before next frame can be transmitted.
2	Store and controlled TX start. When the MDR[5] bit equals to a 1 and the CPU writes a 1 to the ACREG[2] bit, the transmitting peripheral device starts frame data transmission. Bit 2 is self clearing. See paragraph 2.1.1.2.
3	Send 2-μs Ir pulse. When the MDR[6] bit equals to a 1 and the CPU writes a 1 to the ACREG[3] bit, the transmitting peripheral device sends a 2-μs IR pulse at the end of the frame. Bit 3 clears automatically by the transmitting peripheral device at the end of 2-μs IR pulse data transmission. See paragraph 2.1.1.3.
4	MIR half mode. When bit 4 is set to a 1, the operating speed in the MIR mode changes to 0.58 Mbs.
5	Short time-out. When bit 5 is set to a 1, the time-out interval in the Status FIFO time-out interrupt is changed from 1 ms to 128 μs.
6	RX enable. Bit 6 must be set to a 1 to enable data receive in all IR modes.
7	TX enable, Bit 7 must be set to a 1 to enable data transmission in all IR modes.

4.11.7 Status FIFO Line Status Register (SFLSR) – (read only)

Address A (Hex) – Bank 0

Bit No.	[7–5]	4	3	2	1	0
Definition	Not used	Status FIFO empty	Overrun errors	Frame length error	PHY error	CRC error
Default		1	0	0	0	0

Bit definition:

BIT NO.	DESCRIPTIONS
0	CRC Error . Bit 0 is set to a 1 when a bad CRC is detected on data receive for that particular frame.
1	PHY error. In the FIR mode, bit 1 is set to a 1 when an illegal symbol is received during data reception for that particular frame. In the MIR mode, bit 1 is set to a 1 when an abort pattern (at least 7 consecutive 1's) is received during data reception.
2	Frame-too-long. Bit 2 is set to a 1 when a frame exceeding the maximum length (set by the RXFLH and RXFLL registers) has been received.
3	Overrun error. Bit 3 is set to a 1 when an overrun in the RXFIFO occurs.
4	Bit 4 is set to a 1 when the Status FIFO is empty. The CPU uses bit 4, not the LSR[1] bit, to determine the empty status when it receives a Status FIFO threshold interrupt.

4.11.8 Status FIFO Register Low (SFREGL) – (read only)

Address C (Hex), – Bank 0

4.11.9 Status FIFO Register High (SFREGH) – (read only)

Address D (Hex) – Bank 0

The length of each frame is stored in the Status FIFO register. When the CPU executes a read transaction with address C (Bank 0), status FIFO[7–0] bits are loaded to the bus. When the CPU executes a read transaction with address D (Bank 0), status FIFO[11–8] bits are loaded to the bus.

4.11.10 Resume Register (RESUME) (read only)

Address B (Hex) – Bank 0

When transmit underrun or receive overrun occurs this register should be read to resume normal operation.

4.12 Bank 1 Registers

4.12.1 TV Configuration Register (TVCFG)

Address: A (Hex) – Bank 1

Bit No.	7	6	5	4	3	2	[1:0]
Description	RX active	Run length coding enable	0 = programmed T mode 1 = over-sampling mode	RX carrier frequency range 0 = 30K–56K 1 = 400K–500K	RCVR internal demodulation disable	TX carrier frequency range 0 = 30K–56K Hz 1 = 400K–500K Hz	Modulation mode 00 = C_PLS 01 = 8_PLS 10 = 6_PLS 11 = Reserved
Default	0	0	0	0	0	0	00

Bit definitions:

BIT NO.	DESCRIPTIONS
0	Bit 0~1: Transmitter modulation mode.
1	00: C_PLS modulation mode 01: 8_PLS modulation mode 10: 6_PLS modulation mode 11: Reserved
2	TX High–speed carrier frequency select 0: 30–56KHz 1: 400–500KHz
3	Receiver internal demodulation disabled. 0: enabled 1: disabled
4	RX High–speed carrier frequency select 0: 30–56KHz 1: 400–500KHz
5	RX sampling mode 0: Programmed T-period 1: Oversampled mode The oversampled mode is used to catch the snapshot of an incoming signal so as to determine the period of the carrier signal. The programmed T-period mode is used to receive data at a lower sampling rate (compared to “oversampled” mode) which matches the bit rate of incoming signal. Thus programmed T-period reduces the amount of data used to represent the incoming signal. In normal operation, the “programmed T-period” mode should be chosen.
6	Run length encoding/decoding enable 0: disabled 1: enabled
7	1: Indicate the RX is active 0: RX is inactive Bit 7 is automatically set when valid infrared pulses are detected

4.12.2 TV Demodulation Configuration Register (TVDMCFG)

Address: B (Hex) – Bank 1

Bit No.	[7–6]	[5–4]	[3–0]
Description	Not used	Delta frequency	Center frequency
Default		00	0000

Table 4–1. RX Demodulation Carrier Frequency (low range TVCFG[4] bit = 0)

TVDMCFG [3–0] 3 2 1 0	TVDMCFG [5–4] = 00		TVDMCFG [5–4] = 01	TVDMCFG [5–4] = 10	TVDMCFG [5–4] = 11
0000	25	±2	±4	±6	±8
0001	30	±2	±4	±6	±10
0010	35	±2	±5	±8	±10
0011	40	±2	±5	±10	±15
0100	45	±2	±8	±15	±20
0101	50	±2	±8	±15	±20
0110	55	±2	±8	±15	±20
0111	60	±2	±10	±15	±25

(Freq. In kHz)

Table 4–2. RX Demodulation Carrier Frequency (high range TVCFG[4] bit = 1)

TVDMCG [3–0] 3 2 1 0	TVDMCFG [5–4] = 00		TVDMCFG [5–4] = 01	TVDMCFG [5–4] = 10	TVDMCFG [5–4] = 11
0000	400	±20	±50	±100	±150
0001	450	±20	±50	±100	±150
0010	480	±20	±50	±120	±200

(Freq. In KHz)

4.12.3 TV Modulation Configuration Register (TVMDCFG)

Address: C (Hex) – Bank 1

Bit No.	[7–5]	[4–0]
Description	Carrier pulse duration	Carrier frequency.
Default	000	00000

Table 4–3. TX Modulation Carrier Frequency

FREQUENCY RANGE	TVMDCFG Bits 4 3 2 1 0	FREQUENCY (kHz)
TVCFG[2]=0 (low range)	00000 00001 00010 … 11010 other	30 31 32 … 56 invalid
TVCFG[2]=1 (high range)	00000 00001 00010 00011 other	400 450 480 500 invalid

Table 4–4. TX Modulation Carrier Pulse Duration

FREQUENCY RANGE	TVMDCFG bits 7 6 5	PULSE WIDTH (µs)
TVCFG[2]=0 Low Range	000 001 010 011 other	6.0 7.0 9.0 10.6 Invalid
TVCFG[2]=1 High Range	000 001 010 011 other	0.7 0.8 0.9 1.0 Invalid

4.13 Bank 2 Registers

4.13.1 Prescaler Register (PRESC)

Address: A (Hex) – Bank 2

Bit No.	[7–6]	[5–0]
Description	Not used	Prescaler Value
Default		011010

Bits[5–0]: Prescaler Value. The prescaler unit divides the 48-MHz input clock by the value in the PRESC register. The 48-MHz input clock is divided by integers from 1 to 31. When PRESC is a 0 there is no output clock. The default value for the PRESC register is 1A(Hex).

4.13.2 Interrupt Configuration Register (ICR) – (write only)

Address: B (Hex) – Bank 2

Bit No.	7	6	[5–4]	[3–0]
Description	IRQ active level select	IRQ output mode select	Status FIFO threshold level	IRQ channel select
Default	0	0		0001

Bits [3–0]: IRQ channel select. There are 11 IRQ channels. Any particular IRQ channel can be selected by programming the ICR[3–0] bits.

ICR[3–0]	IRQ channel	ICR[3–0]	IRQ channel
0001	IRQ3	0111	IRQ10
0010	IRQ4	1000	IRQ11
0011	IRQ5	1001	IRQ12
0100	IRQ6	1010	IRQ14
0101	IRQ7	1011	IRQ15
0110	IRQ9	Others	None

Bits [5–4]: These two bits decide the threshold level needed:

BIT [5–4]	THRESHOLD LEVEL
00	1
01	4
10	7
11	8

Bit 6: IRQ output mode select. IRQ output can be a totem-pole or an open-drain output. A 0 on bit 6 selects the open-drain output and a 1 selects the totem-pole output. The default value is 0 (open drain). In the totem-pole output configuration, the output can be a 1, a 0 or a Z, whereas in the open-drain configuration, the output can be only a 0 or a Z.

Bit 7: In the totem-pole configuration. IRQ output can be an active high or an active low. When the IRQ[7] bit is a 0, active high is selected and when the IRQ[7] bit is a 1, active low is selected. The default value is a 0 (active high). Bit 7 is not relevant when open drain is selected.

4.13.3 DMA CHANNEL SELECT REGISTER (DCSR) – (write only)

Address: C (Hex) – Bank 2

Bit No.	7	6	5	4	3	2	1	0
Definition	Status FIFO reset	Enable DRQ	RX DMA channel select			TX DMA channel select		
Default	0	0	000			000		

Bits [2–0]: TX DMA channel select. Set these bits as shown to select the data transmission DMA channel for interrupt and acknowledge.

DCSR[2–0]	DMA channel
001	DRQ0/$\overline{\text{DACK0}}$
010	DRQ1/$\overline{\text{DACK1}}$
100	DRQ3/$\overline{\text{DACK3}}$

Bits [5–3]: RX DMA channel select. Set these bits as shown to select the data receive DMA channel for interrupt and acknowledge.

DCSR[5–3]	DMA channel
001	DRQ0/$\overline{\text{DACK0}}$
010	DRQ1/$\overline{\text{DACK1}}$
100	DRQ3/$\overline{\text{DACK3}}$

NOTE:

The same channel cannot be selected for both data transmission and data receive at the same time. If selected, the DMA channel output will be disabled (driven to a high impedance state). For the UART application, where both data transmission and data receive interrupts are active simultaneously, separate channels have to be assigned for data transmit and data receive. However for the infrared mode of operation, where data transmit and data receive are not active at the same time, a particular channel can be selected alternately for data transmit and data receive.

Bit 6: When bit 6 is set to a 1, it enables the selected DRQ channel.

Bit 7: Status FIFO is reset by writing a 1 to this bit. Bit 7 is self clearing.

4.14 Bank 3 Registers

4.14.1 General Purpose I/O Direction Register (GPIODIR) – (write only)

Address: B (Hex) – Bank 3

Bit no.	7	6	5	4	3	2	1	0
Definition	The corresponding bit is the direction bit for that GPIO terminal. When a 0, the GPIO terminal is an output and when a 1, the GPIO terminal is an input.							
Default	0	0	0	0	0	0	0	0

4.14.2 General Purpose I/O Data Register (GPIODAT) – (read and write)

Address = C, Bank 3

Bit No.	7	6	5	4	3	2	1	0
Definition	GPIO8 data	GPIP7 data	GPIO6 data	GPIO5 data	GPIO4 data	GPIO3 data	GPIO2 data	GPIO1 data
Default	0	0	0	0	0	0	0	0

When a GPIO terminal is in the input mode, the value at the GPIO terminals are latched into the GPIODAT registers, which the CPU can read. But when a GPIO terminal is in the output mode, internal data is available at the GPIO terminal. The value at the output terminal depends on the GPIODAT register value, the test pin of the device and the GPFSR value.

4.14.3 General Purpose Function Select Register (GPFSR) – (write only)

Address D , Bank 3

Bit No.	7	6	5	4	3	2	1	0
Definition	GPFSR [7–0]							
Default	0	0	0	0	0	0	0	0

When the TEST pin is a 0, the value at the GPIO terminal depends on the GPFSR value. When the GPFSR bit is a 0, the corresponding GPIODAT register bit is available in the GPIO terminal. When, the GPFSR bit is a 1, some internal signals are available at the GPIO terminals. Refer to the following:

TEST PIN	GPFSR [7–0]	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0
1	X	FIR/MIR RX clock	FIR/MIR TX clock	RX state[1]	RX state[0]	TX state[3]	TX state[2]	TX state[1]	TX state[0]
0	1111_ 1111	0	0	0	0	0	DDIS	Presc.clk out	Baudout Z
0	0000_ 0000	GPIODAT [7]	GPIODAT [6]	GPIODAT [5]	GPIODAT [4]	GPIODAT [3]	GPIODAT [2]	GPIODAT [1]	GPIODAT[0]

4.14.4 Ir Mode Configuration Register 1 (IRCFG) – (read and write)

Address A, Bank 3

IR Configuration Register (IRCFG)

Bit No.	[7–4]	3	2	1	0
Definition	Not used	SIR pulse duration 0: 3/16 of bit rate 1: 1.6-us	Not used	Select IRRVL/IRRVH in TV mode and SHARP mode	TEMIC bit
Default		0		0	0

Bit definition:

BIT NO.	DESCRIPTIONS
0	Temic IBM bit. Bit 0 is used for automatic mode selection (High speed/Low speed) in Temic IBM transceivers
1	Select IIRVL/IRRVH. Bit 1 selects the IIRVL and IRRVH in the TV and SHARP modes.
2	Not used
3	SIR pulse duration select. Bit 3 will determine the selection of either a 1.6 µs or a 3/16 of baud-rate pulse width.

Table 4–1. TIR 2000 Registers

ADDRESS	REGISTERS	
	LCR[7] = 0	LCR[7] = 1
0	RBR (R) , THR (W)	DLL
1	IER	DLM
2	IIR (R) FCR (W)	IIR (R) FCR (W)
3	LCR	
4	MCR	
5	LSR	LSR
6	MSR	
7	SCR	
MODE	UART, SIR & SHARP	MIR, FIR & TV

ADDRESS	REGISTERS							
8	MDR							
9	CCR							
	BANK 0		BANK 1		BANK 2		BANK 3	
	WR	RD	WR	RD	WR	RD	WR	RD
A	TXFLL	SFLSR	TVCFG		PRESC		IRCFG	
B	TXFLH	RESUME	TVDMCFG		ICR		GPIODIR	
C	RXFLL	SFREGL	TVMDCFG		DCSR		GPIODAT	
D	RXFLH	SFREGH					GPFSR	
E	PLR	ACREG					ABDL	
F	ACREG						ABDH	
CCR [7–6]	00		01		10		11	

5 Electrical Specifications

5.1 Absolute Maximum Ratings Over Operating Free Air Temperature Range (Unless Otherwise Noted)†

Supply voltage range, V_{CC} . . . . . –0.5 V to 6 V
Input voltage range, V_I . . . . . –0.5 V to V_{CC} + 0.5 V
Output voltage range, V_O . . . . . –0.5 V to V_{CC} + 0.5 V
Operating free-air temperature range, T_A . . . . . 0°C to 70°C
Storage temperature range, T_{stg} . . . . . 0°C to 150°C

† Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

5.2 Recommended Operating Conditions

Low voltage (3.3 V nominal)

		MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}		3	3.3	3.6	V
Input voltage, V_I		0		V_{CC}	V
High-level input voltage, V_{IH} (see Note 1)		0.7 V_{CC}			V
Low-level input voltage, V_{IL} (see Note 1)				0.2 V_{CC}	V
Output voltage, V_O (see Note 2)		0		V_{CC}	V
High-level output current, V_{OH}	I_{OH} = –12 mA, See Note 4	V_{CC} – 0.8			V
	I_{OH} = –4 mA, See Note 5	V_{CC} – 0.8			
	I_{OH} = –2 mA, See Note 6	V_{CC} – 0.8			
Low-level output current, V_{OL}	I_{OL} = 12 mA, See Note 4			0.5	V
	I_{OL} = 4 mA, See Note 5			0.5	
	I_{OL} = 2 mA, See Note 6			0.5	
Input capacitance, C_I				15	pF
Operating free-air temperature, T_A		0	25	70	°C
Virtual junction temperature range, T_J (see Note 3)		0	25	115	°C
Oscillator/Clock speed				48	MHz
Clock duty cycle			50%		
Jitter specification				±100	ppm

NOTES: 1. Meets TTL levels, $V_{IH(min)}$ = 2 V and $V_{IL(max)}$ = 0.8 V on nonhysteresis inputs.
2. Applies for external output buffers.
3. These junction temperatures reflect simulated conditions. Absolute maximum junction temperature is 150°C. The customer is responsible for verifying junction temperature.
4. These parameters apply for D7–D0, IRQ3–IRQ15, DRQ0, DRQ1 and DRQ3.
5. These parameters apply for GPIO0–GPIO7, XSOUT, $\overline{\text{XRTS}}$, $\overline{\text{XDTR}}$, XIR–TXD.
6. These parameters apply for XOUT.

Standard voltage

		MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}		4.75	5	5.25	V
Input voltage, V_I		0		V_{CC}	V
High–level input voltage, V_{IH}		0.7 V_{CC}			V
Low–level input voltage, V_{IL}				2 V_{CC}	V
Output voltage, V_O (see Note 2)		0		V_{CC}	V
High-level output current, V_{OH}	I_{OH} = –12 mA, See Note 4	V_{CC} – 0.8			V
	I_{OH} = –4 mA, See Note 5	V_{CC} – 0.8			
	I_{OH} = –2 mA, See Note 6	V_{CC} – 0.8			
Low-level output current, V_{OL}	I_{OL} = 12 mA, See Note 4			0.5	V
	I_{OL} = 4 mA, See Note 5			0.5	
	I_{OL} = 2 mA, See Note 6			0.5	
Input capacitance					pF
Operating free-air temperature, T_A		0	25	70	°C
Virtual junction temperature range, T_J (see Note 3)		0	25	70	°C
Oscillator/Clock speed				48	MHz
Clock duty cycle			50%		
Jitter specification				±100	ppm
I_{CC} at FIR mode			3		mA

NOTES: 2. Applies for external output buffers.
3. These junction temperatures reflect simulated conditions. Absolute maximum junction temperature is 150°C. The customer is responsible for verifying junction temperature.
4. These parameters apply for D7–D0, IRQ3–IRQ15, DRQ0, DRQ1 and DRQ3.
5. These parameters apply for GPIO0–GPIO7, XSOUT, XRTS, XDTR, XIR–TXD.
6. These parameters apply for XOUT.

5.3 System Timings

SYMBOL	PARAMETER	MIN	MAX	UNIT
$t_{(AR)}$	Valid address to read active	10		ns
$t_{(AW)}$	Valid address to write active	10		ns
$t_{(DH)}$	Data hold	5		ns
$t_{(DS)}$	Data setup	15		ns
$t_{(RA)}$	Address hold from inactive read	20		ns
$t_{(RD)}$	Read pulse width	40		ns
$t_{(RDH)}$	Read data hold	10		ns
$t_{(RVD)}$	Active read to valid data	35		ns
$t_{(WA)}$	Address hold from inactive write	10		ns
$t_{(WR)}$	Write pulse width	40		ns

5.4 Timing Diagrams

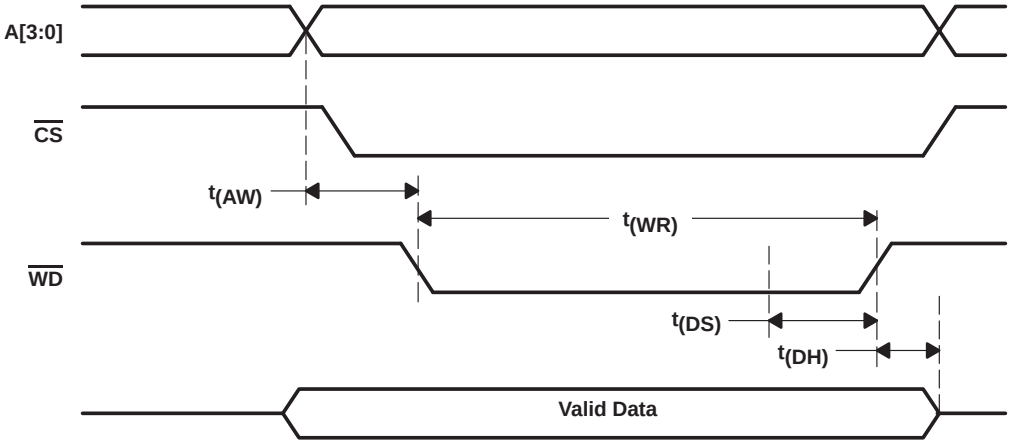


Figure 5–1. Standard ISA Write-Cycle Timing Diagram

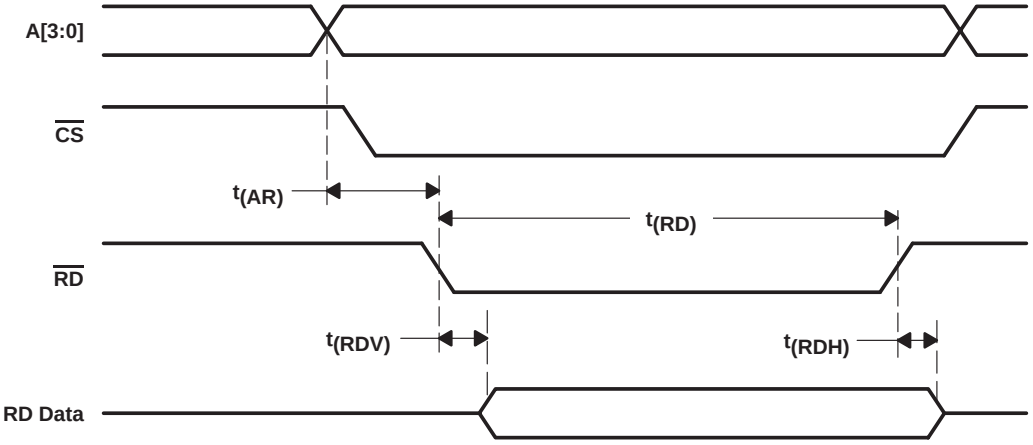


Figure 5–2. Standard ISA Read-Cycle Timing Diagram

(This page has been left blank intentionally.)

6 Application Information

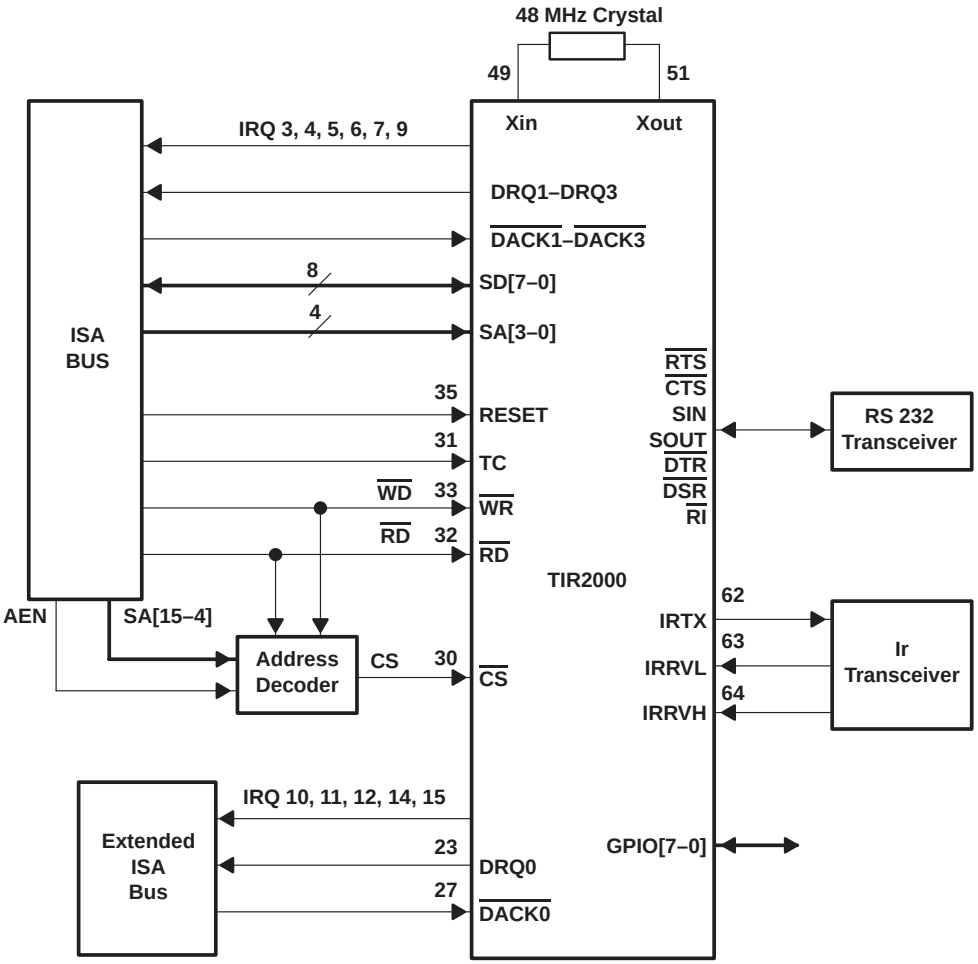


Figure 6–1. Typical TIR2000 Configuration

6.1 Transceiver Connections

The TIR2000 can be connected to an HP or TI transceiver as shown in Figure 6–2. The HP and TI transceivers which are pin-to-pin compatible have one transmit pin (TXD) and two receive pins (RXD-A and RXD-B). TXD pin of the transceiver should be connected to the IRTX pin of TIR2000 device. RXD-A pin of the transceiver should be connected to the IRRVL pin of the TIR2000 device and is used to receive data in the slow speed IR and TV mode. RXD-B pin of the transceiver should be connected to the IRRVH pin of the TIR2000 device and is used to receive data in the medium speed IR, high speed IR and SHARP ASK modes.

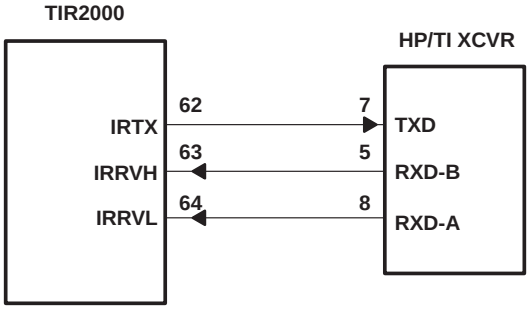


Figure 6–2. HP/TI Transceiver Connection

The TIR2000 can also be connected to a Temic transceiver as shown in Figure 6–3. Connect TXD pin of the transceiver to IRTX pin of TIR2000 device. The transceiver has only one RXD pin which is used for both high speed IR as well as slow speed IR operation. Connect RXD pin of the transceiver to IRRVH and IRRVL of the TIR2000 device. The SD pin of the transceiver needs to be programmed correctly so that it will operate properly in high speed IR mode. Follow the programming steps for different modes as described next.

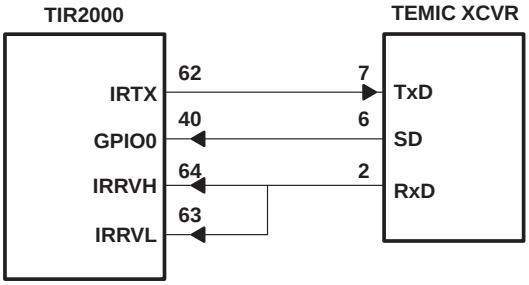


Figure 6–3. Temic Transceivers

6.2 Operating Modes

To Operate in SIR (115 kb/s) IrDA Mode:

1. Set mode to SIR mode
2. Program GPIO0 pin as an output pin
3. Write 1 to GPIODAT[0] bit
4. Write 0 to GPIODAT[0] bit

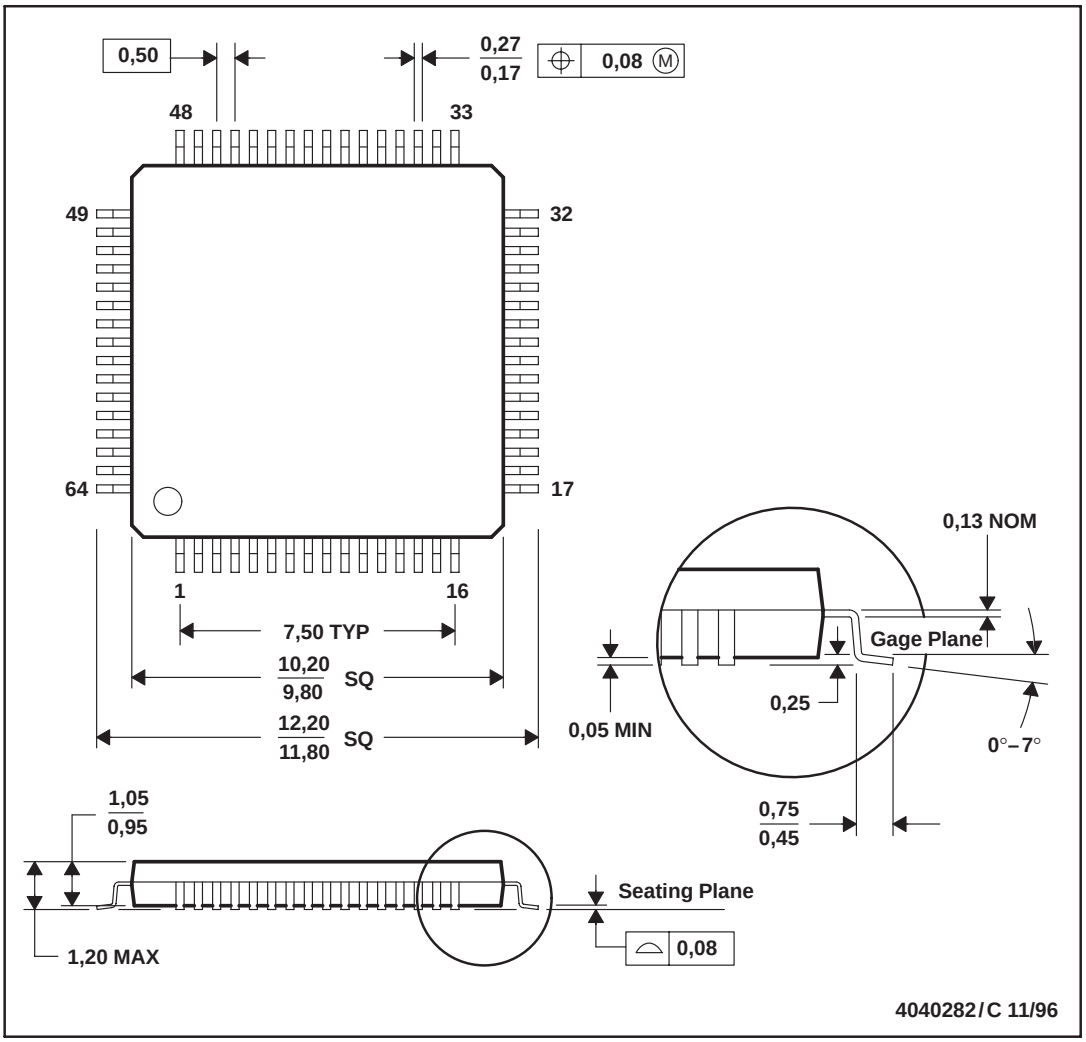
To Operate in FIR (4 Mb/s) or MIR (1.152 Mb/s) IrDA Mode:

5. Set mode to FIR or MIR mode
6. Program GPIO0 pin as an output pin
7. Write 1 to GPIODAT[0] bit
8. Write 1 to IRCFG[0] bit
9. Write 0 to GPIODAT[0] bit
10. Write 0 to IRCFG[0] bit

(This page has been left blank intentionally.)

Appendix A
Mechanical Data

PAG (S-PQFP-G64) **PLASTIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-026

(This page has been left blank intentionally.)

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.