

SirPic

Serial & Infra-Red Data Terminal

Interface Guide v2.01



Author : Ravi Deo
Doc Version: 2.01
Updated: 18th August 2005

© Copyright 2004-2005 Ravi Deo
All Rights Reserved
All Trademarks mentioned herein are property of their respective companies.

Contents

1	Introduction	3
1.1	Purpose Of Document.....	3
1.2	Related Documents.....	3
2	Interface Types	4
3	IrTx Interface.....	5
3.1	IrTx Construction.....	5
3.2	IrTx Schematic	5
3.3	IrTx Stripboard Layout.....	6
4	SirTx Interface	7
4.1	SirTx Construction.....	7
4.2	SirTx Schematic	8
4.3	SirTx Stripboard Layout	8
5	SerialPIC Interface.....	10
5.1	SerialPIC Construction.....	11
5.2	SerialPIC Schematic	11
5.3	SerialPIC Stripboard Layout.....	12
6	InfraPIC Interface.....	13
6.1	InfraPIC Construction.....	15
6.2	InfraPIC Schematic	15
6.3	InfraPIC Stripboard Layout.....	16

DISCLAIMER:

THE SIRPIC FAMILY OF SOFTWARE, FIRMWARE AND HARDWARE IS PROVIDED AS IS, WITH NO WARRANTY OF MERCHANTABILITY AND/OR FITNESS FOR ANY PARTICULAR TASK. THE USER ASSUMES ALL RESPONSIBILITY FOR ITS USE. ALL SOFTWARE IS INSTALLED AT USERS OWN RISK. THE AUTHOR, OR ANY OTHER SIRPIC CONTRIBUTORS ARE NOT RESPONSIBLE FOR ANY DAMAGE AND/OR LOSS OF DATA OR PROFITS CAUSED BY THIS SOFTWARE OR HARDWARE. NO REPRESENTATION OR WARRANTY IS GIVEN AND NO LIABILITY IS ASSUMED WITH RESPECT TO THE INFORMATION GIVEN IN THIS DOCUMENT.

1 Introduction

SirPic is a Serial and Infra-red data terminal application for the Palm PDA. It is a versatile tool for sending and receiving data traffic via a wired (RS232) or wireless (infra-red) connection.

SirPic is ideal for PIC developers who require a portable wired or wireless terminal unit for their custom PIC project.

1.1 Purpose Of Document

To use SirPic with a custom PIC microcontroller, requires the use of a hardware SirPic interface. This document provides schematics and construction details on the various types of SirPic Interfaces which are supported by SirPic.

1.2 Related Documents

SirPic users should also read the following documents (available from <http://www.sirpic.com>):

- SirPic User Guide
- SirPic Programmers Guide

2 Interface Types

SirPic supports the following interface types:

- IrTx** : Wireless raw Infra red transmitter
- SIrTx**: Wireless serial to Infra red converter and transmitter
- SerialPIC**: Wired serial (RS232) bi-directional interface
- InfraPIC**: Wireless Infra red bi-directional interface

The following table compares the SirPic hardware interfaces.

Interface Name	Connectivity	Transmit / Receive	API Support		Comments
			PIC Assembler	PIC BASIC	
IrTx	Wireless Infra-red (SIR)	Transmit only	YES	NO	Simple low cost interface using common components.
SIrTx	Wireless Infra-red (SIR)	Transmit only	YES	YES	Requires PS2IR converter. Superseded by bi-directional InfraPIC interface.
SerialPIC	Wired Serial (RS232)	Transmit & Receive	YES	YES	Uses common MAX232 level converter.
InfraPIC	Wireless Infra-red (SIR)	Transmit & Receive	YES	YES	Requires PSXIR converter.

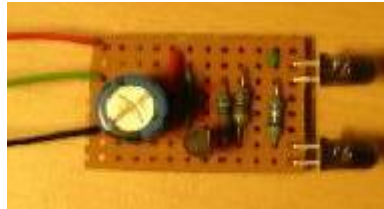
The choice of interface will depend upon the requirements of the PIC application:

- Wired (RS232) or Wireless (Infra-red) interface?
- Transmit only (using SirPic as a data logger), or Transmit & Receive (using SirPic as a terminal)?
- Choice of programming language (Assembler / BASIC /C)?
- Component count and cost?

Each of the interfaces, are discussed in further detail in the following chapters.

3 IrTx Interface

The IrTx interface is a raw infra red transmitter, ideal for PIC Assembler developers who wish to use the SirPic API assembler subroutine for transmitting infra red data to SirPic (as a data logging application).



The modulation of the infra-red signal is done within the PIC microcontroller assembler code which is time critical (using the SirPic API assembler *TxChlr* subroutine). The SirPic assembler subroutines require approximately 40 bytes of code space.

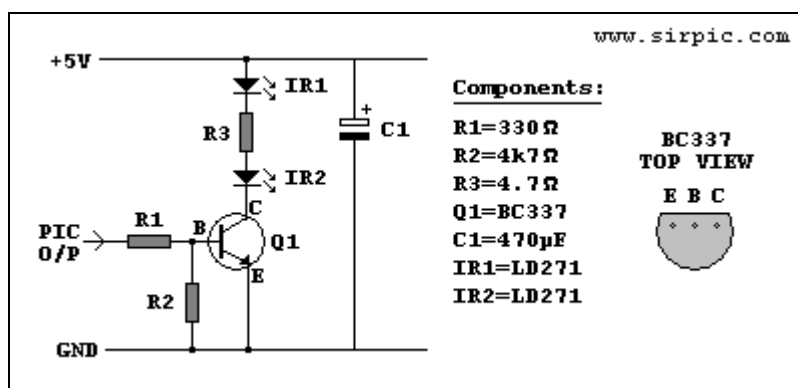
This interface is not suitable for PIC BASIC programmers, as it requires time critical code to perform the modulation. PIC BASIC compilers code generation vary from one vendor to another, and as such is not a reliable means of generating time critical code. PIC BASIC programmers requiring a wireless infra-red connection are advised to use the SIrTx or InfraPIC interface.

3.1 IrTx Construction

The figures below show a constructed IrTx interface.

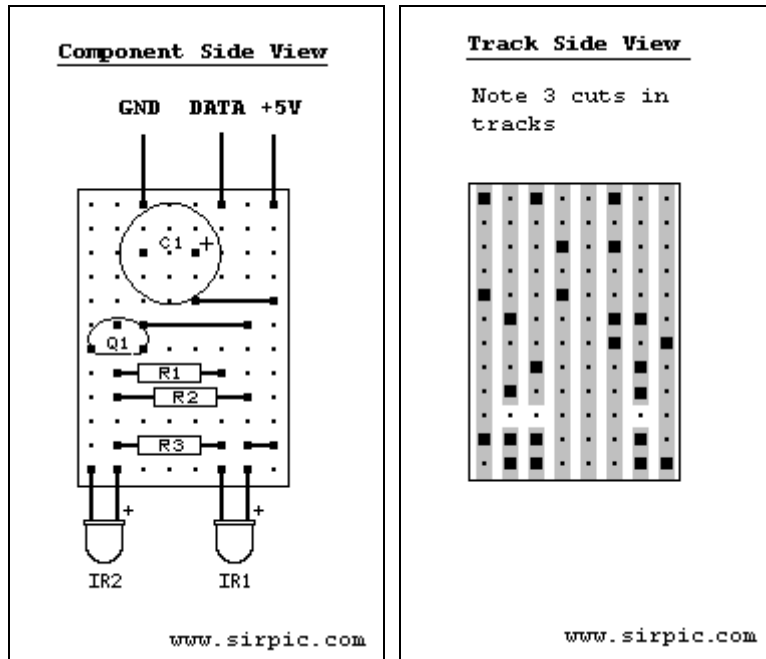


3.2 IrTx Schematic



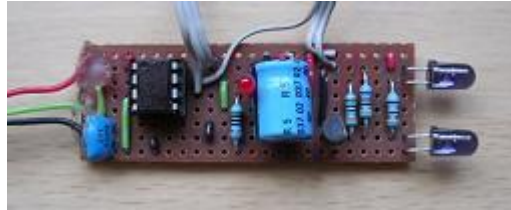
3.3 IrTx Stripboard Layout

For ease of construction the circuit diagram has been transposed onto a piece of stripboard (8 tracks x 12 holes).



4 SlrTx Interface

The SlrTx interface comprises of the IrTx interface with a pre-liminary "bolt-on" serial to infra red pulse converter. The SlrTx interface unit converts an existing PIC Serial data output signal to an infra red signal, via the custom PS2IR (PIC Serial to Infra Red) converter.

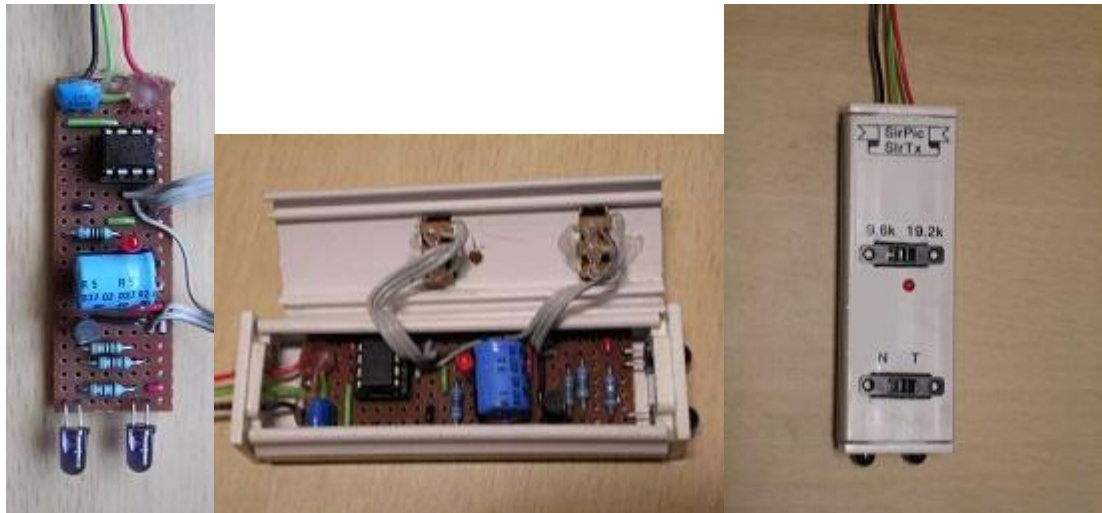


The SlrTx interface has the following advantages:

- ❑ PIC BASIC developers can use the standard SEROUT command to transmit infra-red data from a suitable PIC output pin.
- ❑ PIC developers who have an existing wired PIC serial output data stream (9600 & 19200 baud) can go "wireless" without any PIC code modification.
- ❑ PIC Assembler developers can use the *TxChSer* subroutine to transmit via the conventional "bit-bang" method.
- ❑ PIC developers can save on valuable CPU cycles, by using the PICs onboard hardware UART (as provided by 16F628) to transmit infra red data concurrently in the background.
- ❑ Infra red LEDS are protected from the damaging effects of accidental prolonged constant "on" voltage, extending the component life.

4.1 SlrTx Construction

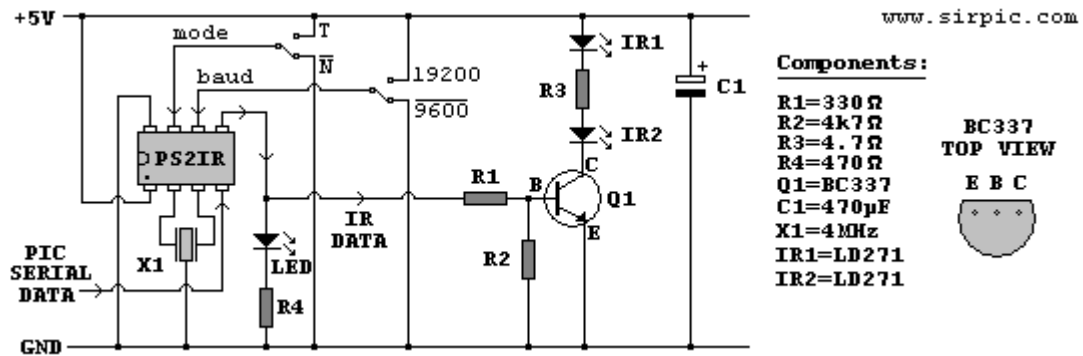
The figures below show a constructed IrTx interface.



The SlrTx interface houses two switches for changing baud rate (9600/19200) and serial mode (N/T). However if it is only ever intend to use one serial baud mode (eg N9600), then the corresponding PS2IR connections can be hardwired via jumper wires.

In the pictures above, there are some minor component layout changes to fit the enclosure (capacitor mounted axially, and the GND and +5V connections for one of the switches has been taken from lower down the track).

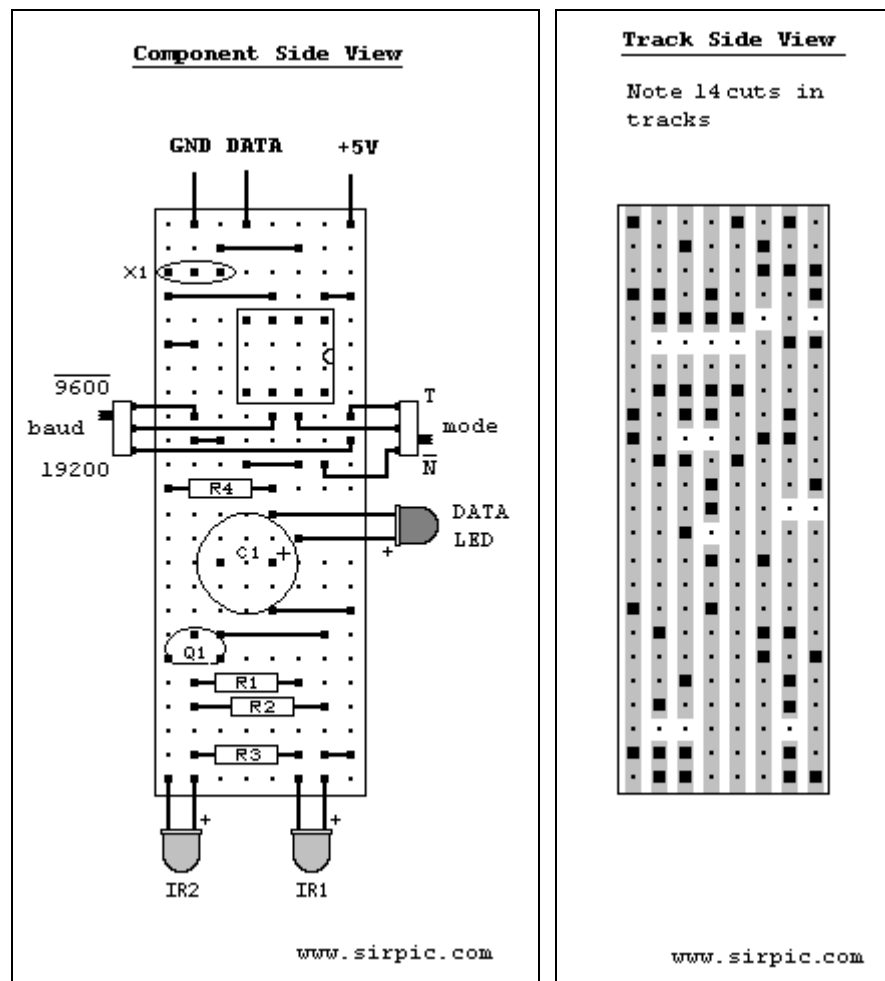
4.2 SirTx Schematic



4.3 SirTx Stripboard Layout

To aid construction, the circuit diagram has been transposed onto a piece of stripboard (8 tracks x 24 holes).

NOTE: The custom PS2IR converter PIC is plugged unconventionally with pin 1 in the top right corner of the DIL socket.



The custom PS2IR converter is a programmable microchip device, and the hex file can be programmed onto either a PIC12F629 or PIC12F675 device. The corresponding hex files are available from the SirPic website (<http://www.sirpic.com>).

5 SerialPIC Interface

The SerialPIC interface is a bi-directional serial TTL to RS232 voltage level converter. This uses the popular MAX232 IC to convert TTL levels (0 and +5V) to RS232 levels (+10 and –10V) and vice-a-versa.



The SerialPIC interface:

- ❑ Allows simple bi-directional data transfer (transmit and receive), with no flow control.
- ❑ Allows PIC developers to save on valuable CPU cycles, by using the PICs onboard hardware USART (as provided by 16F628) to transmit/receive serial data concurrently in the background.
- ❑ Can be used to communicate directly to a PDA or PC equipped with a serial RS232 port. Supports direct connection of the Palm HotSync serial cable (or other NULL modem cable), as well as straight serial cables for connection to a PC or laptop.
- ❑ Conforms to RS232 voltage specifications.
- ❑ SirPic API also supports the *bit-bang* method for PICs, which do not have onboard USARTs. The bit bang method produces a *True* mode bit patterns, as produced by the PIC USART
- ❑ Can be used by PIC Basic programmers for True mode output (see note below).
- ❑ Enables use of third party data loggers or terminal emulation software, as well as SirPic for data logging.

PIC BASIC Programmers Note: The SerialPIC interface is designed to be used in *True* mode for PIC output, and should not be used for TTL inverted outputs (eg T9600 PIC serial mode is valid, whereas N9600 mode is invalid).

True mode implies +5V during data inactivity (idle), 0V pulses, for start bit, stop bit and zero data bits.

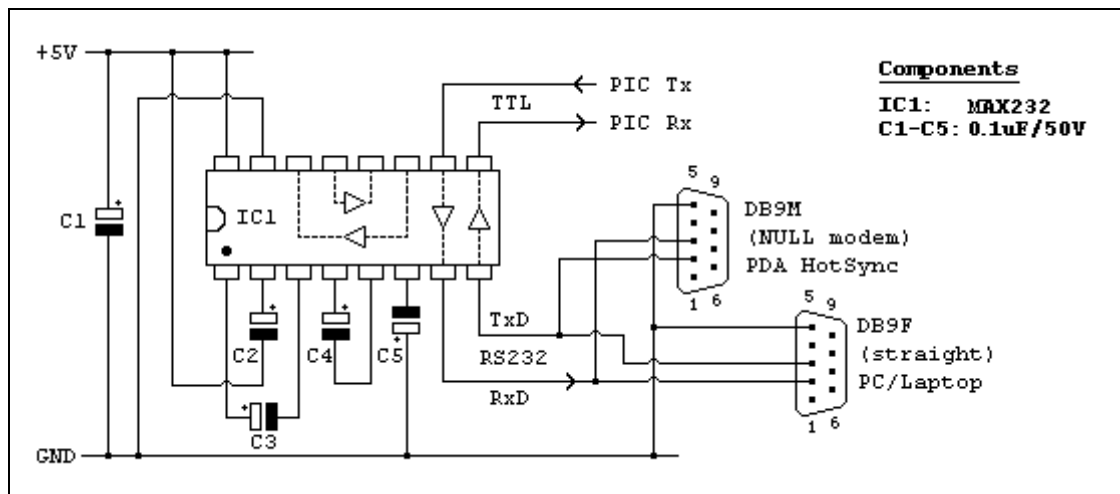
5.1 SerialPIC Construction

The figures below show a constructed SerialPIC interface.



The SerialPIC interface has four wired connections (GND, +5V, PIC_TX, PIC_RX). On the RS232 side, two DB9 ports are provided (male and female). This provides flexibility for connecting "straight" to a PC serial port or "crossed" to the PDA hotsync cable (which is effectively a NULL modem cable). The DB9 male port allows the PDA hotsync cable to be directly plugged into the SerialPIC interface.

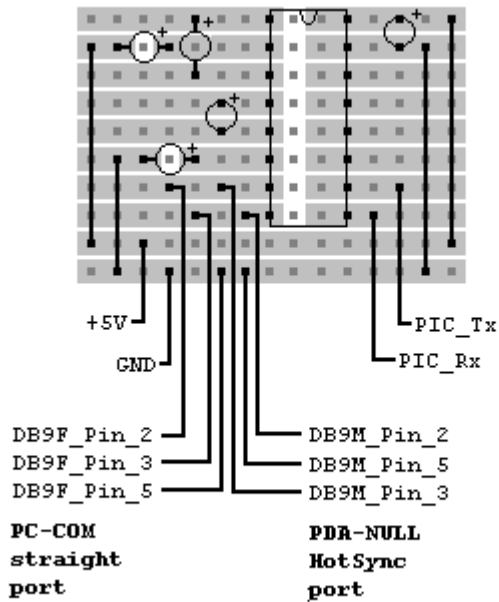
5.2 SerialPIC Schematic



5.3 SerialPIC Stripboard Layout

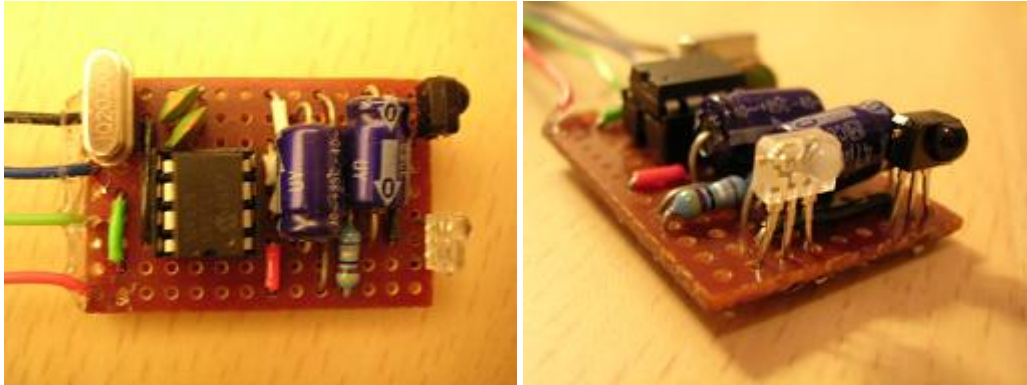
To aid construction the circuit diagram has been transposed onto a piece of stripboard (10 tracks x 15 holes). The component layout shown below is the plan view from the component side, with the underlying tracks shown as semi-transparent.

NOTE: Polarity of electrolytic capacitors, and 10 breaks on the underlying track side.



6 InfraPIC Interface

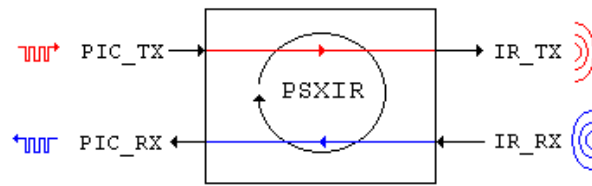
The InfraPIC interface is a bi-directional serial TTL to Infra-red pulse shape converter. The InfraPIC interface unit converts infra-red input/output signals to TTL, using the custom PSXIR bi-directional converter.



The InfraPIC interface:

- ❑ Allows simple bi-directional (transmit and receive) half-duplex data transfer at 9600 baud (modes: 8N1, 7E1, 7O1).
- ❑ Can be used to communicate directly to a PDA or PC equipped with an infra-red port, and able to receive/transmit IrDa (SIR) data.
- ❑ PIC Assembler developers can use both “bit-bang” or USART routines to transmit and receive data (as per usual for serial 9600 mode)
- ❑ PIC developers to save on valuable CPU cycles, by using the PICs onboard hardware USART (as provided by 16F628) to transmit/receive serial data concurrently in the background.
- ❑ Can be used by PIC Basic programmers for True mode output. The InfraPIC interface is designed to be used in *True* mode for PIC input and output, and should not be used for TTL inverted outputs (eg T9600 PIC serial mode is valid, whereas N9600 mode is invalid).

NOTE: The InfraPIC interface provides a simple half-duplex communication link, and does not perform any flow control or data validation or retransmissions. This responsibility is placed upon the PIC application developer, where required. Users requiring full IrComm functionality onboard should consider using a Microchip MCP2150 based offering.



The diagram above illustrates the data flow to and from the custom PSXIR microchip. The custom PSXIR converter is a programmable microchip device, and the hex file can be programmed onto a PIC12F629 or PIC12F675 device. The corresponding hex files are available from the SirPic website (<http://www.sirpic.com>).

The PSXIR microchip:

- ❑ Polls both infra red input and serial output whilst idle (until PIC_TX or IR_RX becomes active).
- ❑ Disable infra red reception (IR_RX) whilst transmitting infra-red data (PIC_TX --> IR_TX).
- ❑ Disables serial transmission (PIC_TX) whilst receiving infra-red data (IR_RX --> PIC_RX).
- ❑ Maximum conversion delay of 52us (PIC_TX --> IR_TX, IR_RX --> PIC_RX)
- ❑ Ideal for PIC application based upon a simple request/reply mechanism.

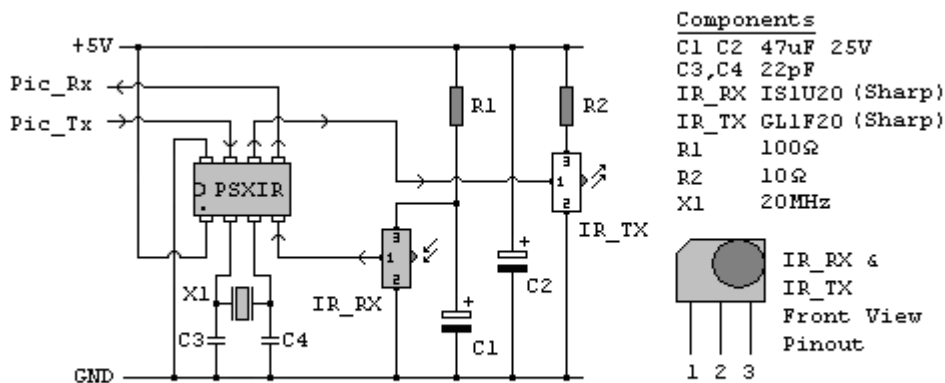
6.1 InfraPIC Construction

The figure below show a constructed InfraPIC interface, boxed in a familiar transparent enclosure.



The InfraPIC interface has four wired connections (GND, +5V, PIC_TX, PIC_RX). On the infra-red side, an infra-red transmitter (left side) and receiver (right-side) enable bi-directional communication (half-duplex).

6.2 InfraPIC Schematic



Note the use of a 20 MHz crystal oscillator with a pair of ceramic capacitors. It is not advisable to use a 3-pin 20 Mhz ceramic resonator, due to poor stability, which could lead to a higher reception error rate.

6.3 InfraPIC Stripboard Layout

To aid construction the circuit diagram has been transposed onto a piece of stripboard (9 tracks x 14 holes). The component layout shown below is the plan view from the component side, with the underlying tracks shown as semi-transparent.

NOTE: Polarity of electrolytic capacitor, and 8 breaks on the underlying track side.

