STANDARD
MICROSYSTEMS
CORPORATION

APPLICATION NOTE 7.6

SMSC IrCC
(Infrared Communications Controller)
Hardware Design Guide

By Neil Winchester

INTRODUCTION

It is the purpose of this document to describe in detailed technical terms the issues and solutions involved in the successful application of IrDA FIR (Fast Infrared) using SMSC Super I/O and Ultra I/O (SIO) devices containing an IrCC (Infrared Communications Controller). The focus is on hardware issues involving the Transceiver and IrCC, including methodologies for design, test, debug and qualification.

Because of the detailed technical nature of the discussion, the intended audience is primarily hardware designers with significant analog and mixed-signal design experience, but some sections will be of interest to others, particularly BIOS authors.

The reader is **strongly encouraged** to obtain the latest data sheets and application notes from the transceiver manufacturer before attempting such a design. In addition, the relevant IrDA specification documents are available on the Internet for download (http://www.irda.org, free of charge) and at least the Physical Layer Specification should be skimmed. Also on the recommended reading list is the SMSC IrCC Data Sheet and the Data Sheet for the specific SIO device containing it.

The very existence of this document is testament to the fact that:

Applying an FIR transceiver is anything but a simple task, and so both the skill and the effort required in order to achieve a properly functioning design should not be underestimated.

The receivers operate with wide bandwidth and sub-microamp full-scale currents, which makes them extremely sensitive to noise, either power supply, conducted or radiated. The transmitters operate at very high peak currents with fast rise and fall times, enabling them to radiate significant EMI, and the power dissipated can result in such excessive temperature rise that it can actually damage the hardware. **Having said this, it *IS* possible to apply these devices successfully, other engineers have done it before, and it is the purpose of this document to provide the knowledge and techniques with which to do it.**

80 Arkay Drive
Hauppauge, NY 11788
(516) 435-6000
FAX (516) 273-3123

TABLE OF CONTENTS

INTRODUCTION 1

HARDWARE AND SOFTWARE REQUIREMENTS 3
BACKGROUND - IrDA, SIR AND FIR 3
THE COMPONENTS IN AN IR SYSTEM 4

THE IR Tx 5

TX CURRENT, POWER AND HEAT 5
TX STUCK-AT-1 PROTECTION 6
SHARP ASK TX 7

THE IR Rx 8

GENERAL LAYOUT 8
POWER SUPPLY FILTERING 9
LOAD CAPACITANCE 10
EMI PREVENTION 10
EMI SHIELDING 10
SWITCHING POWER SUPPLIES 10
FLAT PANEL DISPLAYS 10
DONGLES 11

USING SMSC EVB's WITH A TARGET SYSTEM 12

NULL MODEMS & SMSC FIR TRANSCEIVER EVB'S 12
SMSC IrCC EVB'S WITH TARGET TRANSCEIVERS 13

TESTING AND DEBUGGING 13

TESTING AND DEBUGGING THE IrCC 13
TESTING THE BIOS 15
TESTING AND DEBUGGING THE TRANSCEIVER 15
IR Tx TESTING 16
IR Rx TESTING 17
TRANSCEIVER QUALIFICATION TESTING 18

In order to avoid mistakes before design, rather than try to fix them afterwards, which is not always possible without a complete redesign, it is recommended that this document be at least skimmed **before** the design effort begins. While basic considerations such as the **Hardware Resource Map** might appear obvious, many of the other issues such as **Thermal Management, EMI emission and susceptibility, and component selection and layout considerations** are unique to the application of FIR transceivers.

HARDWARE AND SOFTWARE REQUIREMENTS

The procedures in this document will **require**:

- An oscilloscope with at least 100 MHZ bandwidth (preferably an analog one) and probes with extremely short ground clips. A digital scope can be used, and is even desirable for detecting intermittent noise problems, but it must have at least 2 GSa/s and 100 Kpts. waveform memory if it is to replace the analog scope.

- Two (2) Desktop computer systems, each equipped with (i) an SMSC SIO EVB containing an IrCC, (ii) SMSC FIR Transceiver EVB, and (iii) SMSC IR_UTILS software package. It is unimportant if the specific SIO device is the same as the target system or not, so long as it contains an IrCC. The systems must be Pentium (or faster) with at least 16 MB RAM and an ISA bus with at least 1 free slot bridged from a PCI bus. In order to permit the same machines to be used for Device Driver testing, they should have either spare hard disks, or else the disk should be partitioned so that it can run at least OSR2 or Memphis. The MSDOS.SYS file (hidden file in the root directory) on each machine should be modified for **BOOTGUI=0**, and the setup program run in AutoExec.Bat, as described in the IR_UTILS Release Notes.

BACKGROUND - IrDA, SIR and FIR

The **IrDA** (Infrared Data Association) has defined a set of standards intended to permit the exchange of information between equipment using wireless infrared light as the medium. The Version 1.0 defined operation at up to 115.2 Kbps. More recently, Version 1.1 defines operation at up to 4 Mbps in a way that is fully compatible with existing V1.0 equipment. In both versions, the links are point-to-point (i.e., only 2 stations at a time can communicate), with a distance range of up to 1 meter.

The technique used in V1.0 is called **SIR** (Serial InfraRed), and is based on a scheme similar to a UART operating with 1 start + 8 data + 1 stop. The IR modulation scheme is **RZI** (return to zero inverted), which consists of producing an optical pulse for each 0 bit and no pulse for each 1 bit (RZI). Since each character begins with a 0 start bit, an IR pulse will be present at the start of each character time which the Rx (receiver) requires for synchronization. The pulses can either be (i) $^{3}/_{16}$ duty cycle or (ii) fixed at 1.6 μS (which is D=$^{3}/_{16}$ @ 115.2 Kbps). The characters are assembled into frames that contain special characters (BOF - beginning of frame, EOF - end of frame) in order to establish the frame boundaries. A 2-byte CRC16 is appended to the data before the closing EOF. In order to provide for data transparency, an ESC sequence is defined using byte stuffing, similar to IBM BiSync (an antique character-oriented data link protocol).

When two stations wish to establish a link, they exchange a number of parameters that define their capabilities regarding supported speeds, frame sizes, etc. This negotiation always takes place at 9600 bps, and this is the only required speed; all other speeds (2.4 / 19.2 / 38.4 / 57.6 / 115.2 Kbps) are optional. In order to avoid a situation in which multiple stations interfere with each other, each station is required to monitor the medium before it transmits. The **IrLAP** (Infrared Link Access Protocol) standard defines the procedures which the stations must use in order to gain access to the medium, discover each other and exchange capabilities, establish a link, exchange data over the link, and shut down the link.

The way V1.1 achieves compatibility with V1.0 equipment is twofold. Firstly, the negotiation for V1.1 stations is identical to V1.0 stations, with the exception that the capability to support additional speeds can be exchanged. Secondly, whenever a V1.1 station is operating at an **FIR** (Fast IR) speed, it is required to send SIP's (Serial Interaction Pulses) according to timing that will make SIR stations realize that the media is busy, so they will not interfere.

There are three (3) FIR speeds defined in V1.1: 0.576, 1.152 and 4 Mbps. Both the 0.576 and 1.152 Mbps speeds use a common signaling technique, and are sometimes referred to as MIR (Medium IR), although this term is not defined in the IrDA specifications.

The MIR signaling technique resembles HDLC, which is a modern bit-oriented data link protocol. Exchanges are based on frames, whose start and end boundaries are marked using special bit patterns (called flags) that do not appear within the data portion of the frame. The start of frame is indicated by sending 2 STA bytes (0x7E) and the end of frame is marked with a single STO byte (also 0x7E). Prior to the STO, a 2-byte CCITT CRC16 is inserted. Since the flags contain 6 consecutive 1's, data transparency is provided by the Tx stuffing a 0 bit after a group of 5 consecutive 1's in the data portion of the frame, which the Rx then removes. More important for the purposes of this document is the bit encoding, which is also RZI, like the SIR case. However, for MIR speeds, the duty cycle is fixed at $^1/_4$ (i.e., each zero bit is signaled by issuing an IR pulse whose width is $^1/_4$ of a bit time). Due to the combination of RZI signaling and zero-stuffing, a sufficient number of IR pulses is always present to permit Rx clock synchronization.

At 4 Mbps, the modulation scheme used is called 4PPM (4 Pulse Position Modulation). In this technique, each pair of bits is sent as a symbol. The 500 nS symbol time is divided into 4 non-overlapping subintervals of time, each lasting for 125 nS, which are called chips. Each of the 4 possible values that the double-bits can assume is represented by sending an IR pulse during 1 of the 4 chips. This results in an IR duty cycle of $^1/_4$ on average. Note that if a "11" symbol is followed by a "00" symbol, then 2 consecutive chips will be sent that emit IR; this is referred to as a "double pulse", as opposed to the "single pulse" when a transmitted chip has no IR before or after it. This distinction is important when considering transceiver behavior.

4 PPM data is organized into frames analogously to the MIR case, with start and stop flags, CRC etc. However, it is not necessary to bit-stuff in order to achieve data transparency; this is achieved by using symbol patterns for the flags that are not valid for the data. The CRC is a 4-byte IEEE CRC32.

THE COMPONENTS IN AN IR SYSTEM

An **Infrared Transceiver Module** is used to provide the raw transformation from the electrical to the infrared medium and back. No processing of any kind occurs in the transceiver - it merely converts the signal between the media as best as it can. Of course, no transceiver is perfect, so both the Rx and Tx are distorted to some extent, but the digital controller (the SMSC IrCC) is designed to be tolerant (within limits) to these distortions.

The SMSC IrCC contains two (2) sections: an **ACE** (Asynchronous Communications Element, or UART) section for SIR, and an **SCE** (Synchronous Communications Engine) section for FIR.

The **ACE** section provides a register interface identical to a standard 550A UART, as well as the RZI Modems (Modulator/Demodulator). Since this block provides only the character-level framing for SIR, all additional frame-level functions (e.g., byte stuffing, CRC calculation, BOF's, EOF's, etc.) is provided in software. Like the 550A, there is a programmable baud rate generator, a 16 byte FIFO, and operation is interrupt driven.

The **SCE** provides not only the bit-level modems (also called ENDEC's, or ENcoder/DECoders), but also all of the frame-level functions required by all 3 FIR speeds, including bit stuffing, CRC calculation and checking, etc. There is a 128 byte FIFO in order to buffer bus latency. Data transfer to the ISA bus uses demand-transfer DMA, since it is the only technique with sufficient speed for use at 4 Mbps with an 8-bit ISA device. Operation is interrupt driven. (As an aside, single-transfer DMA and PIO modes are also provided, but are too slow on an ISA bus bridged from PCI to be useful at 4 Mbps, although both techniques do have adequate sustained throughput for MIR operation).

THE IR Tx

Several common problems have been experienced with the IR Tx:

- **The Tx hardware "burns up"**
  - Usually R_{LED} power rating insufficient
  - Excessive temperature rise due to inadequate thermal design
  - Also happens when an IrDA Tx is used for Sharp ASK
  - Sometimes a result of bypass capacitor Irms and ESR ratings
  - TxD signal floating or pulled up to logic 1 at power up andno stuck-at-1 protection, or not working properly

- **The transceiver latches up (i.e., classic CMOS latchup)**
  - Usually AC coupled Tx without a buffer or current limit

- **The Tx exhibits insufficient range**
  - Usually R_{LED} resistance value is incorrect

- **Excessive EMI emission**
  - Usually a combination of PCB layout and bypass capacitors

This section will discuss the general considerations when applying an IR Tx, as well as a discussion of the causes and solutions to the above issues.

Tx CURRENT, POWER AND HEAT

As was mentioned in the IrDA summary, the IR TX must operate with pulse widths up to **78 µS**, and a duty cycle up to **D=1/4**. The exact value varies from device to device, but typical IR LED peak current is about 1/2 **Amp**. That's right -- 1/2 Amp -- 500 mA! In addition, the rise/fall time of the current pulse must be fast (usually **< 10 nS**) because the IR pulse itself must have a rise/fall time < 40 nS in order to meet IrDA specifications.

The first thing that should be realized is that this section will dissipate an unusual amount of power, and provision must be made for this both when specifying the components and the thermal design. In particular, a peak current of 500 mA at 5 V and a duty cycle of 1/4 will dissipate a total power of **625 mW** continuous (2.5 W peak), and provisions must be made in the thermal design to remove this heat without excessive temperature rise.

The next issue related to power dissipation is the specification of the resistor (R_{LED}) that sets the IR LED current. The exact value varies from device to device, but it is typical for the voltage drop (and hence the power dissipation) to split about evenly between the transceiver and the external resistor, which results in a resistor value of about 5 Ω. The power dissipated in this resistor will be about half of the total power, or about 300 mW. Of course, at maximum supply voltage, the current and power will both be substantially higher than this value. Clearly, a single resistor in a small package (e.g., 0603 or 0805) cannot dissipate this much power. In addition, it is often desired for this resistor to exhibit a ± 1% tolerance in order to limit the effect of the resistor tolerance on the final LED current. This is due to the fact that it is often only a small difference between the minimum LED current required in order to meet minimum IrDA Tx intensity (**100 mW/sr**) at minimum power supply, and the maximum current at maximum power supply that the LED can tolerate without damage. Combining the above results in a resistor specification of:

5 Ω ± 1%, 500 mW - Consult the module manufacturer for exact value!

Since ± 1% resistors are not available with resistance less than 10 Ω, or with a power rating above 125 mW, as a standard part (especially in surface mount packages), one solution is to use a parallel combination of standard values. For example, 4 devices in 1206 packages will provide the 500 mW rating, and also provide extreme flexibility in setting the final resistance value if individual values are not exactly equal. For example, if all 4 resistors are 22.6 Ohms, the final resistance is 5.65 Ω, but if one of the resistor is changed to 22.1 Ω, then the final value is 5.62 Ω. **Failure to use the correct resistance value will either result in Tx damage (if too low) or insufficient Tx distance (if too large). Failure to use a resistor of sufficient power rating will result in damage.**

Because of the fast rise/fall times of the current pulse, it is important that the loop area be kept small, and that a high frequency ceramic bypass capacitor be used between the Vcc connection of R_{LED} and the Tx GND.

Failure to observe these requirements can result in rise/fall times that are too slow to meet IrDA specifications, as well as extremely high levels of radiated EMI. In extreme cases, it might also be necessary to shield the Tx power circuits in order to control emissions, but this is usually not necessary, especially since a solid ground plane MUST be used under the entire transceiver circuitry if the Rx is to function properly (which is discussed at length in the IR Rx section).

In addition, the lower frequency current will need to be bypassed as well. A 500 mA pulse train at $D=1/4$ will result in **Irms = 250 mA** ($I_{RMS} = (I_P^2 * D)^{1/2}$). Note that small electrolytic capacitors are *NOT* rated to handle this magnitude of current, and special (and large) low-ESR varieties should be used. As an example, a **220 µF / 0.1 Ω** device will drop 50 mV from the 500 mA Ip (ESR * Ip) flowing through its ESR, will droop about 180 mV after integrating this current for a 78 µS pulse width (Ip * PW / C), and will dissipate about 6 mW of average power (I_{rms}^2 * ESR), which is a reasonable combination. An alternative would be to rely on the low frequency bypassing in the rest of the system, but doing so will cause the 500 mA Ip to flow though the system grounds and any connectors between the transceiver and the capacitors, all of which must be carefully considered.

If an electrolytic capacitor is placed in parallel with the ceramic device as was just described, then ringing of the parasitic resonant circuit they form is a real consideration, which should be investigated and corrected if found. The solution involves damping of the resonant circuit, either series or parallel. Since low series resistance is needed for the filtering to be effective, parallel damping with another capacitor having a specific ESR value is the preferred technique.

Tx STUCK-AT-1 PROTECTION

Most IR Tx's simply turn on an IR LED whenever their TxD pin is active. If the pin is active for an excessive period of time, or for too high a duty cycle, then damage will result to either the IR LED, or the external R_{LED}, or both. One situation in which this can happen is if the pin on the SIO device that is feeding the IR Tx has an internal pullup (e.g., FDC37C669FR HDCS0 pin). When power is first applied to the system, the pin is unconfigured and will produce a static logic 1 of long duration. Eventually, the BIOS will change the pin function and make it a static 0, but each time power is applied the IR Tx is stressed severely, and will likely exhibit significantly reduced lifetime (i.e., MTBF). In other cases, the device pin might not have an internal pullup, but might power-up as an input pin and external leakage currents might pull the pin up to a logic 1 until the pin function is configured, which will result in the same situation. A third possibility is a software malfunction that will cause the TxD signal to present a long static 1. As usual, the solution to the problem varies with the cause.

The simplest solution that addresses all causes of stuck-at-1 faults is to **use a transceiver that contains internal protection** for this. Transceivers with this feature are just beginning to appear in the marketplace at the time of this writing, and should be seriously considered.

The next simplest solution addresses only the case in which leakage currents are pulling the pin up. If the pullup current is small enough, then a pulldown resistor of reasonable value can be used to establish a logic 0 during power-up without adversely affecting V_{OH}. As an example, suppose an I/O pin and the transceiver pin each have a leakage current of 10 µA for a combined total of 20 µA; a 10 KΩ pulldown resistor will reliably establish a logic 0 at 0.2 V, but only reduce I_{OH} by 0.4 mA @ 4V.

For the case in which the pullup current could be large and/or if it is desired to provide protection for software stuck-at-1 faults as well, then the solution involves limiting the pulse width in the TxD signal net. The smallest, simplest, and least expensive way to do this is to introduce the equivalent of a 1-shot (i.e., monostable multivibrator) using a discrete capacitor-resistor. This is also referred to as AC-Coupling the TxD signal. The technique involves placing a capacitor (e.g., 0.1 µF) in series between the SIO device pin and the TxD input to the transceiver, and then placing a resistor (e.g., 10 KΩ) to ground at the transceiver TxD input pin.
[**Aside:** This technique tacitly makes use of the parasitic diode at the input to the Tx pin of the transceiver in order to provide a return path for the current with which to reset the capacitor at the end of each Tx pulse. If there is no diode internal to the module (e.g., HP HSDL-1100/2100), then an external diode must be used, but this causes other problems as will be discussed shortly].

However, if this is all that is done, then **a CMOS transceiver will LATCHUP** because there is no current limit for the reverse current that will be pulled from the Tx pin when the capacitor is fully charged and the I/O pin transitions from a 1 to a 0. In order to address this case, it is necessary to insert some resistance in series with the capacitor to limit this current. For example, a 560 Ω resistor will limit the current to about 10 mA. Although this has been found to work in practice, the manufacturers of transceivers with CMOS input buffers do *NOT*

recommend applying anything less than -0.5V to their TxD pin, no matter what the current limit is. At first it might appear that adding an external Schottky diode would satisfy this requirement, but the only Schottky diodes with guaranteed forward voltage less than 0.5 V are power rectifiers, and these devices have much too high a junction capacitance to use in an application like this. Small signal Schottky diodes have suitable capacitance, but do not have the required V_F rating except at unrealistically low current levels.

The only way to satisfy the manufacturer's specifications and also provide the pulse width limit using AC coupling **is to place a buffer** (e.g., HCT) after the resistor/capacitor/resistor arrangement. HCT buffers **do** have guaranteed latchup current specifications, making them ideal for this purpose. Unfortunately, there is a penalty in size and cost when using a buffer.

When driving a HP transceiver that does not have a buffered CMOS input, the latchup situation is completely avoided. However, the HP TxD input is a direct connection to the base of a Schottky transistor, so there is no parasitic diode at the pin, and an external diode must be used. Unfortunately, just placing such a diode at the input of the module will interfere with the Tx turn-off at the end of the pulse. However, a bigger problem is the current that must flow into the TxD pin -- 6 mA DC is required, and ± 50 mA peak (for about 15 nS during the transitions). The recommended application circuit uses a 560 Ω resistor to limit the current assuming a V**OH** of about 4 V is applied and VBE is about 0.6 V. In addition, a 220 pF "base speed-up" capacitor is placed in parallel with this resistor in order to rapidly inject and remove minority charge carriers from the base-emitter junction, which requires a peak current of about 50 mA for about 15 nS. Because of the incomplete reset of the AC coupling capacitor, combined with droop during the pulse and the effect of the necessary current limiting resistor, it becomes **exceedingly difficult** to both establish the AC coupling function, and the required base drive function, in a single network **without using a buffer** in between. Should anyone wish to seriously pursue this, an HP Applications Note includes the information needed for doing a **Spice Model** of the TxD signal net. The information needed for the SIO device pin could be extracted by making some simple measurements of I_{OH} vs. V_{OH} and measuring the rise/fall time into a couple of capacitive loads.

SHARP ASK Tx

The Sharp ASK is another IR communication standard. It involves using a 500 KHZ carrier with PCM. Since the carrier duty cycle is 50%, a Tx designed for D=$^1/_4$ IrDA communications ***CANNOT* be used for Sharp ASK!** Doing so will **burn out the transmitter!** In order to operate at D=$^1/_2$, the transmitter must be specifically designed for this purpose.

One approach to achieving ASK operation is to use an external switch (e.g., a power PFET) to switch half the Tx current off when operating in ASK mode. A disadvantage to this approach is that the loop area will cause increased EMI emissions, and the PFET will need to have extremely low RDS in order to avoid materially affecting the value of the series resistor. A far more significant problem is that no standard Device Driver will be aware of the switching mechanism, since there is no standard way to do this within the IrCC specification, so **custom Device Drivers will be needed** with such an approach.

The more direct way to achieve ASK operation is simply to design the Tx so that it can tolerate a 50% duty cycle, but the most straight forward way to do this is to provide 2 Tx's, each operating at half the normal current. Considerations of size and cost make this approach undesirable as well.

Yet another way to achieve ASK operation involves the use of a single Tx at half the normal current, but this will cause the unit to not meet IrDA specifications for Tx intensity, and will reduce the distance capability of the IrDA link by 30%.

Unfortunately, there is just no good way to use a single IR interface for both IrDA and Sharp ASK communications. Perhaps the least objectionable way is to operate the IrDA Tx at half current, and then construct an additional Tx to provide the other half of the Tx power. The additional Tx can be reasonably inexpensive, since it consists of just a buffer gate, a power NFET, an IR LED and a current-setting resistor.

THE IR Rx

Several common problems have been experienced with the IR Rx:

- **The RxD signal contains significant noise on it**
    1. By far the most common problem
    2. Usually conducted from Rx P/S and/or radiated EMI
    3. Sometimes due to gain setting resistor with IBM or Temic modules
        A. Wrong value or missing
        B. Connected to "dirty" digital Vcc instead of "clean" analog Vcc

- **Rx fails at 4 Mbps**
    1. Excessive load capacitive on RxD signal from long cable or PCB trace
    2. RxD pull up resistor either wrong value or missing

- **Rx fails @ 1.152 Mbps with HP**
    Incorrect values for squelch capacitor (Cx3 on HP HSDL-1100/2100 Data Sheets) causes distortion of first bit of first STA byte

- **Rx Erratic**
    Usually power supply; check for both noise and correct DC voltage

- **Range too small**
    Usually with IBM or Temic, wrong value of gain setting resistor

Since Rx noise is the single most common problem, both a definition and an extended discussion is provided here: **when there is no IR incident on the Rx, the RxD signal MUST be a solid static logic 1**. The most common cause of Rx noise is insufficient power supply filtering, followed closely by an improper PCB layout and radiated EMI coupling. In many cases, all of these phenomena exist in the same design.

In order for the Rx to function properly, the noise on the P/S at the Rx itself must be less than about **10 mVpp** with no significant high frequency content. The type of filtering necessary to achieve this depends to a great extent on the nature of the power from the rest of the system. Because of this, only general guidelines can be offered here, rather than specific recommendations, although the examples should provide a useful starting point.

GENERAL LAYOUT

The most basic guidelines are:

- A solid ground plane should be used under the entire Transceiver circuit and associated components.

- This ground plane should be broken from the rest of the ground plane on the PCB and only connected to the rest of the plane in a discrete area along a single edge. For example, suppose the placement of the transceiver components is approximately rectangular. The ground plane should be isolated from the rest of the board around this entire rectangular area, with the exception of a small region along one edge, which should be connected to the ground plane from the rest of the board. All signals into and out of the transceiver circuitry should be routed over this ground plane connection, and no signals should be routed over the breaks in the plane.

- NO other signals, especially any fast digital ones, should be routed through the transceiver region of the board.

- ONLY the transceiver Rx power supply pin (and gain setting resistor, if used) should connect to the output of the Rx power supply filter and NOT the Tx circuitry or any pullup resistor on the RxD signal.

- If a gain setting resistor is used (e.g., with an IBM or Temic device), then this resistor MUST connect to the (quiet) Rx power, and NOT to the (noisy) Tx or system power.

The combination of these techniques will go a long way towards eliminating conducted noise as a problem. In extreme cases, it might also be necessary to isolate both the power and ground lines to the transceiver section at high frequencies by using ferrite beads, but the use of a bead in the ground line should be considered as a last resort because it leaves no high frequency return path for the signals going to and from the transceiver. However, a bead on the Vcc line is always a good thing, and should be considered in every case.

POWER SUPPLY FILTERING

The simplest power supply filter is an RC low-pass filter consisting of a 10 Ω series resistor with a shunt capacitor. The resistor cannot be much larger than 10 Ω because of the voltage drop it would cause. The capacitor will usually need to be a parallel combination of a high frequency ceramic device (e.g., 0.1-1 μF) and an electrolytic device (e.g., 22-220 μF) with low ESR. Whenever 2 or more different capacitor types are connected in parallel like this, ringing of the parasitic resonant circuit they form is a real consideration, which should be investigated and corrected if found. The solution involves damping of the resonant circuit, either series or parallel. Since low series resistance is needed for the filtering to be effective, parallel damping with another capacitor having a specific ESR value is the preferred technique.

Often there is significant high frequency content in the power as it arrives at the transceiver subassembly, and a ferrite bead in series with the resistor is an effective solution. Usually, the rise/fall time of switching converter waveforms is about 30-50 nS, resulting in noise that is bandlimited to about 10 MHZ. Up to this frequency range, 0.1 μF ceramic devices in surface mount packages are effective, but a fairly large ferrite bead is needed to be effective at such "low" frequencies. If even higher frequencies are present, then a smaller value ceramic capacitor should be placed in parallel with the others.

Once the high frequency noise is attenuated to acceptable levels, there remains the problem of the low frequency noise. The 10 Ω series resistor will form an attenuator with the ESR of the electrolytic capacitor above its zero frequency (Fz). For example, a 220 μF / 0.1 Ω capacitor has an Fz at about 7 KHZ and will provide 40 dB of attenuation above that frequency until the contribution of its ESL makes it no longer effective at the higher frequencies. A 22 μF / 1 Ω capacitor will also have Fz at 7 KHZ, but will provide only 20 dB of attenuation when used with a 10 Ω series resistor. These examples represent the extremes of suitable electrolytic capacitors to use in most cases since less than 20 dB of attenuation is rarely sufficient, and the 0.1 Ω ESR capacitor is about the largest practical device.

If excess low frequency noise is still present, then the resistor could be replaced with an inductor having the equivalent Rs, or the external resistor value could be reduced by the Rs of the inductor if it is less than 10 Ω. An inductor with low stray capacitance should be selected, or else it might introduce problems at high frequencies again. The 10 Ω series resistance will be sufficient to damp the resonant circuit formed, but only if the resonant impedance $((L/C)^{1/2})$ is low enough (i.e., it is a series resonant circuit). In other words, if the inductance value is too large for the capacitance being used, it will ring, and there can also be a resonant voltage rise at power-up which could either degrade or destroy the transceiver depending on its severity. Because of these considerations, as well as size and cost, the use of an inductor is considered undesirable unless there is no other way to achieve sufficient low frequency attenuation.

LOAD CAPACITANCE

Given that the Rx now has a ground layout that should prevent conducted noise from being an issue, and the power supply is nice and quiet, the next consideration is capacitive load. The output buffers on the RxD signal are generally rated for about 50 pF maximum C_L. If anything more than a few inches of PCB trace is being driven, then a buffer will likely be needed. The usual symptom from excessive C_L is that operation will be correct except at 4 Mbps. Operation at 4 Mbps will fail because the excessive C_L increases the pulse width.

EMI PREVENTION

The Rx is sensitive to radiated, as well as conducted, EMI; this is due to the extremely small signal (< 1 μA) from the PIN photodiode at maximum distance. Prevention is usually the best way to deal with EMI: consider the possibility of an EMI issue early in the design cycle and either incorporate, or permit the later incorporation of, methods that will rectify it. There are 3 things that are required in order to have an EMI problem, and reducing any one of them, or any combination of them, will be effective in eliminating the problem:

1. A device which radiates EMI
2. A device which is susceptible to EMI
3. A coupling medium

Perhaps the easiest requirement to consider in the early design is the coupling medium, which is affected by the relative position of the Rx to likely sources of EMI, such as switching power supplies and the ribbon cable going to an FPD (Flat Panel Display). To the extent possible, the Rx should be positioned as far away from likely EMI sources as it can be.

EMI SHIELDING

Another way to reduce the effectiveness of the coupling medium is through the use of a shield. Without getting into a detailed discussion of shielding theory and techniques, the EMI problem is usually of the near-field electric variety, for which a grounded conductive shield is an effective solution. Although mu-metal is most effective at the lower frequencies, plated steel is almost as effective, is easier to work with, and is more economical. This shield should cover the entire Rx circuitry, except for small openings to permit the IR to enter and leave, which should be as small as possible. The transceiver should be slightly recessed behind this optical port in the shield. It might be desirable to place the R_{LED} outside of the shield (maybe on the opposite side of the PCB) to minimize the thermal impedance.

SWITCHING POWER SUPPLIES

There are numerous ways in which a switching power supply can radiate EMI and the solutions vary with the cause. Without getting in a detailed discussion of the possibilities, if the converter is the source of the EMI, then it might be possible to get the power supply designer to consider the use of different magnetics (e.g., toroids or pot cores instead of rods or EI/ETD cores), or improved filtering, bypassing, etc. depending on the situation.

FLAT PANEL DISPLAYS

It is sometimes unavoidable that the ribbon cable to the FPD passes near the Rx. In order to minimize the emissions from such a cable, the very best case is to use a shielded cable with twisted pairs and differential line drivers and receivers, but the display designers will almost certainly not agree to this. Probably the best that they will agree to is to use a ground-signal-ground arrangement in their flex cable assembly, and maybe even a ground-plane construction of the cable.

DONGLES

For IR applications, the term "dongle" is usually applied to a device that plugs into a standard RS-232 COM port and contains both an IrDA SIR Modulator/Demodulator (Modem) and an IR Transceiver. For the purposes of this discussion, the term will be used to apply to a transceiver mounted at the end of a cable or PCB trace whose length is more than a few inches.

Once the cable length exceeds a few inches, terminations will definitely be required on all 3 signals because of the transmission line effects caused by the combination of the electrical length of the cable and the fast rise/fall time of the signals. In addition, at least the RxD(s) will definitely require the use of buffers. The ideal case is to use differential drivers and receivers on all 3 signals, but the signals output from the IrCC can be simply series terminated and connected to the cable instead (i.e., single-ended transmission). If the signals on the cable are single-ended, then each signal must be in a twisted-pair with an AC ground, or else a ground-signal-ground approach on a ribbon cable can be used over shorter distances. Of course, if the connection is external to an enclosure in a commercial system, then both ESD protection and EMI/RFI control will need to be provided as well.

The power supply bypassing at the transceiver end of the cable must be sufficient to handle the Tx current pulses at high frequency, and the DC resistance of the cable must be low enough to prevent significant shift in ground reference between the two ends of the cable, especially if single-ended techniques are used. As an example, consider the case of a 500 mA Tx pulse, 1 Vcc line and 2 DC grounds. Suppose it is desired to have a maximum of 100 mV of ground shift and 200 mV of Vcc drop. The resulting Vcc at the dongle will be 300 mV low (-6% of 5V) during the Tx pulses, which is probably the most that can be tolerated. The DC cable resistance will need to be 200 mV / 500 mA = 400 mΩ, which is equivalent to about 6 feet of 28 AWG (7/36, 65 mΩ/ft.) cable. Bear in mind that this is an extreme example, and it would be extremely desirable to keep the cable resistance below this level!

The very simplest approach is used by the SMSC EVB's. The signals output by the SIO device are series terminated and applied to a single row connector with alternating AC grounds. Two DC grounds and one Vcc pin are used in the cable. The cable assembly consists of 5 feet of Category 5 (100 Mbps Ethernet, 24 AWG) cable, which uses a twisted-pair construction. Each signal is twisted with an AC ground. On the dongle side, all signals are received and driven with HCT buffers. The HCT driver on the RxD pin is also series terminated at the dongle. All input pins on both ends of the cable have weak pull-up / pull-down resistors in order to provide defined logic levels for all of the signals in the event that either the cable is disconnected, or the SIO device pins are not configured for IR operation. A 220 μF / 0.1 Ω low-ESR capacitor provides the low-frequency bypassing for the Tx pulses on the dongle. Ground plane and microstrip layout techniques are used at both ends of the cable.

From the above discussion, it should be realized that:

- It **IS** possible to locate a transceiver at the end of a meaningful length of cable.
- It is **NOT** possible to simply connect the transceiver to the end of a meaningful length of cable and expect it work.
- Differential techniques, while superior, are not always required, and single-ended techniques **CAN** be used, but require greater care in their application.

USING SMSC EVB's WITH A TARGET SYSTEM

NULL MODEMS & SMSC FIR TRANSCEIVER EVB's

It is sometimes desired to evaluate the IrCC in a target system either before the transceiver hardware is available or in order to isolate the cause of a system problem (i.e., to determine if the cause is either the transceiver assembly or something else in the system). In this case, either a Null Modem (i.e., a cable assembly) can be used to interface the target system to an SMSC SIO EVB (Evaluation Board) directly, or an SMSC FIR Transceiver EVB can be substituted for the target transceiver assembly.

Since the target system probably does not have series terminations on its IrCC output pins (the way the SMSC SIO EVB's do), the cable used with either method should be kept short, definitely under one foot, and probably under 6 inches. If longer cables are desired, then the techniques discussed in the Dongles section must be used. The connections for the Null Modem vary with the transceiver type used in the target:

HP	IBM/TEMIC	SHARP	SMSC SIO EVB
TxD	TxD	TxD	RxD (pin 1)
RxD-A	RxD	RxD	TxD (pin 3)
RxD-B	-----	------	TxD (pin 3)

Notes:

1. The connections in the table above assume that the SMSC SIO EVB is configured in its default state with the IR3 pin set as a Mode Control output (i.e., the third IR pin is configured for the IRMODE function, as opposed to the IRRX3 function, and it is programmed as an output, as opposed to an input, pin direction). The pin numbers indicated for the SMSC SIO EVB are for the IR Transceiver connector on the EVB.
2. The module manufacturers names indicate the type of module that the target system is designed to use.

In addition to the cable design and construction, the IR Options Register value on one or both ends of the link will need to be changed because of the difference in the polarities between the Null Modem and an FIR Transceiver. This value is easy to change on the EVB end because it can be done with a command line switch on the setup program. To make use of this feature, first run the setup program the normal way and record the value it displays for the IR Options register. To invert both the Tx and Rx polarities, simply compliment the 2 LSB's of the IR Options register value. Run the setup program again using the command line switch to set the new value.

Note: Be careful not to connect a Transceiver Assembly when the Tx polarity is inverted!

In order to connect an SMSC FIR Transceiver EVB to a target system, the original cable assembly should first be removed, and a short cable with a suitable connector (if desired) should be used instead. If the target system was designed for an IBM/Temic transceiver, then full operation is provided. However, target systems designed for HP or Sharp transceivers will not operate the SD/Mode pin, so operation at 4 Mbps will not be possible, while all other speeds will operate normally. The connections are as follows:

HP	IBM/TEMIC	SHARP	SMSC XCEIVER
RxD-A & RxD-B	RxD	RxD	RxD (pin 1)
TxD	TxD	TxD	TxD (pin 3)
ground	SD/Mode	ground	SD/Mode (pin 5)

Note:

Target systems designed for HP and Sharp transceivers will operate at all speeds EXCEPT 4 Mbps; those designed for IBM or Temic modules will work at all speeds. The module manufacturers names indicate the type of module that the target system is designed to use.

SMSC IrCC EVB's WITH TARGET TRANSCEIVERS

It is sometimes desired to exercise a target FIR Transceiver assembly before the rest of the target system becomes available, or in order to isolate the cause of a problem (i.e., to determine if the cause is either the transceiver assembly or something else in the system). In this case, the target Transceiver Assembly can be connected to an SMSC IrCC EVB. As in the cases just discussed, a short cable should be used with a suitable connector (if desired). The connections are as follows:

HP	IBM/TEMIC	SHARP	SMSC SIO EVB
RxD-A	RxD	RxD	RxD (pin 1)
TxD	TxD	TxD	TxD (pin 3)
RxD-B	SD/Mode	(n.c.)	Mode/Irrx (pin 5)

Note: When using an HP module, use the **/Ir3in** switch on the setup program. The module manufacturers names indicate the type of module used on the target Transceiver Assembly.

Exercising the target Transceiver Assembly in this way can be useful from many different points of view:

- The Assembly is often more accessible for probing when it is attached to an ISA EVB than when it is positioned in the target system.

- Since the electrical environment in the target system is usually more hostile than in this connection, if the Assembly does not work well this way, it is unlikely to be suitable for the target system, so the design almost certainly will need to be revised. (Unfortunately, the converse is true as well: even if the Assembly does work well this way, that is no guarantee that it will work as well in the target system because of the most hostile electrical environment).

- Since experience has shown that it is often necessary to revise the Transceiver design a few times, it is possible to get an early start on that work by testing the first version(s) with an EVB before the target system is available.

- If the Transceiver misbehaves in the target system, then it can be useful to know if an EMI situation in the Target is the cause, which is a likely conclusion if operation with an EVB is correct.

- If something about the transceiver Assembly could cause damage to the system that it is connected to, it is probably better for that to be a relatively inexpensive, and easily replaceable, EVB system rather than an expensive prototype system.

TESTING AND DEBUGGING

TESTING AND DEBUGGING THE IrCC

In a target system, it is the responsibility of the PnP BIOS to handle all Configuration Mode issues related to the SIO device that contains the IrCC. This includes not only PnP hardware resource assignments, but also the programming of any device pin functions, directions and polarities. It is also the responsibility of this configuration software to set the IR Options Register for IrDA SIR operation.

With an SIO EVB, a setup program is used to perform these functions, and is usually executed in AutoExec.Bat. Before a complete BIOS is available on a target system, sometimes the setup program for the SIO EVB's can be used instead. Sometimes a custom setup program is needed either (i) because the setup program for the EVB does not program all of the registers that need to be programmed (e.g., it never sets up the core logic registers, which are unique to each system), or (ii) it does not support the required values. Sometimes a setup program is not possible because some of the registers that need to be programmed are not accessible to software running on the host processor.

A number of the programs in the SMSC IR_UTILS software package (a set of DOS-based IR utility programs) can be used to determine if the IrCC has been properly configured. Note that all of the tests will search for an SIO device and use its Configuration Register contents to determine hardware resource assignments as a default behavior. This works fine as long as the registers contain a value that is numerically equal to the actual ISA assignment (i.e., the resources are mapped "straight-through"). However, in a number of situations, mostly with

IRQ's and DMA's, the Configuration Register value is *NOT* the same as the ISA value, and so an INI file (IR_UTILS.INI) containing the actual ISA values must be used. The most basic programs with which to test for proper IrCC configuration are:

FindChip: Will search for the device Configuration registers, and display useful information about the SIO itself, as well as the present values of the IrCC Configuration Registers, and any override values provided by an INI file. If any of these register or resource values are incorrect, then the test programs will *NOT* operate properly (if at all), so it is **extremely important** to take care of this problem before proceeding!

UartChk: Will search the legacy COM port I/O addresses, identify any UART's found, and exercise their IRQ function.

PostSir: Will check the functioning of the SIR (ACE) portion of the IrCC, including UART identification, register test, internal loopback, and external IRQ.

PostFir: Will check the functioning of the FIR (SCE) portion of the IrCC, including ID check, PIO data test, FIFO test, internal loopback, external IRQ and DMA.

SirTx24: Will send bytes of SIR data at 2400 bps. This can be used to check that the TxD pin is programmed correctly for function, direction and polarity, that the IrCC mode is programmed for IrDA SIR operation, and that the TxD signal makes it all of the way to the Transceiver assembly.

Set4Mbps: Will pulse the SD/Mode and TxD pins in such a way as to toggle an IBM or Temic transceiver between fast and slow modes. This can be used to verify that the IR3 signal (SD/Mode) is programmed correctly for function, direction and polarity, as well as confirming that the signal makes it all of the way to the Transceiver assembly. For HP or Sharp designs, this pin is either an input or is not used, so this program does not provide any new information, since SirTx24 has already been used to verify the TxD signal.

A number of the other test programs can be run without regard to the transceiver, if desired, since these programs do not attempt to receive anything, including: MaxTxRst, MaxTxDma, FirTxPio, FirTxDma. However, these programs will provide little information that has not already been obtained by running the tests listed above.

Null Modem or Transceiver EVB: The best way to be sure that everything about the IrCC in the Target System is working correctly is to use either a Null Modem to directly attach the target to an SMSC SIO EVB, or to connect the target to an SMSC FIR Transceiver EVB. An earlier section of this document described the procedures. In either case, the testing need only exercise the IrCC, without regard to the subtleties of the transceiver, and so it is brief: run **MastSir/SlaveSir** @ 115.2 Kbps, and **MastFir/SlaveFir** @ 4 Mbps (or 1.152 Mbps if 4 Mbps is not supported as a result of using an SMSC Transceiver EVB with a target system designed for an HP or Sharp module). If both of these pass, in addition to all of the previous tests, then the IrCC and all 3 IR signals are properly configured, and the signals are all making it to the transceiver. Since the only thing not verified with the previous tests was the RxD(s) signals, any failure here is likely caused by a problem with those pin(s).

IMPORTANT: If any of these tests indicate that the IrCC is not configured or functioning properly, then these issues MUST be resolved before moving on to the evaluation of the transceiver.

TESTING THE BIOS

As was previously mentioned, it is the responsibility of the PnP BIOS to handle all of the hardware resource assignments for the IrCC. In addition, the BIOS has the same responsibility for many other devices. While the various Device Driver Release Notes should be consulted for a more complete discussion, it is required that:

- The devNode for the IrCC must return ALL of the FIR hardware resource assignments. It must have a deviceID of SMSCF010, and (optionally) a compatibility ID of PNP0511. The TYPE bytes must be all zeroes, since it does not belong to any of the legacy type definitions.
- The Current Resource Assignments must be one of the Possible Resource Assignments.
- The Current Resource Assignments must not conflict with any other devNode.
- The IrCC must actually be configured for the Current Resource Assignments and it must be Activated.
- If the IrCC is replicated in a docking station, then the docking and undocking event handler must enable and disable the appropriate IrCC and configure its resource assignments.

The **BiosDump** test program can be used to display the complete information for all of the devNodes enumerated by the BIOS. This report can then be reviewed to ensure that the above requirements are satisfied. If the IrCC devNode appears correct, then the **PnpIrcc** test program can be used to confirm that software which accesses the devNode the way a Bus Enumerator or Device Driver would can correctly obtain the Current Resource Assignments.

TESTING AND DEBUGGING THE TRANSCEIVER

After the above tests, it is known that the IrCC is enabled at the proper hardware resource assignments, it is set for IrDA SIR mode, its IRQ and DMA are functioning, its IrCC is at least basically functional, and the TxD pin (and SD/Mode, if used) are properly configured and the resulting signals are making it to the Transceiver Assembly. It is critical to verify these assertions as fact by using the test methods described above before proceeding with this section!

An even better test is if a Null Modem or Transceiver EVB is used, because it is then known that any problems experienced at this point are due ONLY to the transceiver, and not to the IrCC, its configuration, or to any other part of the system.

If the target Transceiver Assembly is available before the rest of the system (or before the above issues are resolved), then it is possible to interface the Transceiver to an SMSC SIO EVB (described in a previous section), and the following tests can first be conducted with such an arrangement.

Even if a target system is available, it might not be a bad idea to first test the Transceiver with an EVB -- if something about the Transceiver will cause damage to the system to which it is attached, it is probably a lot better for that system to be a cheap production computer with an EVB than an expensive and difficult to replace prototype system.

You will need an oscilloscope with at least 100 MHZ bandwidth and probes with **extremely short** ground clips. An analog scope is preferred to a digital one for most of the testing; a logic analyzer is completely useless.

OK -- this is it -- it's time to power-up the Transceiver for the first time. Quickly probe the Tx and Rx power supplies and confirm that they are not faulted with a short circuit or latchup, and touch all of the components checking for excessive temperature rise. Using a scope probe with a **very short ground clip**, probe the Rx P/S and confirm that the noise is < 10 mVpp with no significant high frequency content. If there is excessive noise, a spectrum analyzer can be useful for understanding the frequency characteristics of it in order to determine the solution. If an IBM or Temic transceiver is used, confirm that the SD/Mode pin is low. For all transceiver types, confirm that the TxD pin is low, and that with no IR incident on the module, inspect the RxD signal(s) and confirm that they are static logic 1's with *NO* random noise pulses on them. For HP modules, look at BOTH the RxD-A and RxD-B pins. For IBM or Temic modules, use FirRxDma at 4 Mbps to check Fast mode and any other speed to check Slow mode.

If noise is experienced on the RxD pin, understand that you WILL need to fix it. There might be some value in proceeding through the rest of the tests, just to see if there's anything else wrong, but do *NOT* expect that any of the tests that use the Rx will pass. See the section on the IR Rx for ideas of some of the possible causes of, and solutions to, Rx noise problems.

For most of the tests which follow, you will need a known-good station with which to communicate. The simplest way to obtain a known-good station is to obtain 2 sets of SMSC IrCC and Transceiver EVB's and place them in production desktop computers. If they can communicate with each other by passing MastSir/SlaveSir and MastFir/SlaveFir, then they are both known-good stations.

IR Tx TESTING

SirTx24: This program will produce maximum IrDA width (78 μS) Tx pulses. Check RLED to see that the Tx is getting keyed, and that the pulse width is about the same as the input (78 μS). Place a known-good transceiver in range of the Tx and check its RxD signal to make sure the Tx is actually producing IR. Check the maximum distance at which the Rx continues to receive in order to determine that the Tx intensity is approximately correct.

Note that most Rx's will produce pulses much more narrow than the IR pulse, so do not be concerned about Rx pulse width, as long as it is appropriate for the Rx being used. Also, most transceivers will talk back on their own Rx when their Tx is running, so do not be concerned with Rx activity while transmitting. Also, all transceivers take a period of time after transmission for their Rx's to recover, and it is not unusual to see spurious Rx activity during this recovery time, as long as it is gone by the time specified in the transceiver data sheet.

Back at the Tx, if any AC coupling is used in the TxD net, confirm the VIH margin at the end of the pulses. Check the Tx P/S (using a **very short ground clip**) for any ringing or overshoot on the edges, excessive droop, and any ground bounce to the rest of the system. Check the pulses at RLED (using a **very short ground clip**) and confirm that they are clean (no overshoot or ringing) and that the rise/fall time is much less than 40 nS (usually about 10 nS).

If an IBM or Temic transceiver is being used, run **Set4Mbps** again, and check the SD/Mode pin at the transceiver module (using a **very short ground clip**) for any overshoot, ringing, noise etc. Since the timing of this signal is not critical, it is possible to use a low-pass filter on it, but this should not be necessary in a proper design. However, if there is an appreciable length of cable or PCB trace to the IrCC, a series terminator at the IrCC might be required to mitigate any transmission line effects.

Run **MaxTxDma** for at least 1 hour and check all components for excessive temperature rise, but especially the transceiver module itself and RLED.

EXTREMELY IMPORTANT! If there is not Tx PW protection in the design, then **do *NOT* execute MaxTxRst** on it - you will risk damaging the hardware. Where possible, redesign to include Tx PW protection, either internal or external to the transceiver module, should be considered.

If there is any AC coupling in the TxD net, run **MaxTxRst** and confirm both that the Tx pulse is effectively limited (by inspecting the R_{LED} waveform) and that the transceiver does not latch up. If a transceiver with internal TxD pulse width limiting is used, then also run MaxTxRst and check the RLED waveform to confirm that the protection is functioning properly.

Run **FirTxDma** with data = 0, frame size = 2048, and speed = 4 Mbps. Check the RLED waveform and the Tx P/S (using a **very short ground clip**) for any ringing or overshoot, and for rise/fall time at RLED. On the known-good station, run FirRxDma at 4 Mbps, and confirm reception. Check the maximum distance in order to confirm that the Tx intensity is approximately correct.

Congratulations! You have now confirmed that:

- The RxD signal(s) is quiet
- The Tx sends IR of approximately the correct intensity and rise/fall time at both extremes of SIR (2400 bps) and FIR speeds (4 Mbps).

- The stuck-at-1 protection circuitry is working correctly and does not cause the transceiver to latchup.
- The power supply bypassing is working correctly and does not ring.
- Both the TxD and SD/Mode (if used) signal nets are working correctly and do not exhibit transmission line effects.
- The thermal design is adequate to avoid excessive temperature rise while transmitting.

IR Rx TESTING

Run **SirTx24** on the known-good system and check the SIR RxD signal on the target system Transceiver. Confirm that the Rx is responding and check the maximum distance in order to verify that the sensitivity is approximately correct. Run **MastSir/SlaveSir** at least at 9.6 and 115 Kbps briefly (maybe 10 frames) in order to confirm basic functionality.

Run **FirTxDma** on the known-good system and **FirRxDma** on the target system. Confirm operation and check distance at least at 1.152 and 4 Mbps. Run **MastFir/SlaveFir** at both speeds (at least 1.152 and 4 Mbps) for a few seconds each in order to confirm basic functionality.

Run **WhichRx** for a few seconds and confirm that it makes no mistakes identifying the type of transceiver module being used.

Run **ManMast** on the known-good system and **ManSlave** on the target system. Confirm that the tests pass at all 9 IrDA speeds.

Congratulations! You have now confirmed:

- The Rx is functioning with approximately the correct sensitivity at both SIR and FIR speeds.
- The entire IR subsystem, both Tx and Rx, is at least basically functional at all 9 IrDA speeds.
- The Transceiver Auto-Detection algorithm used by Device Drivers executes correctly on the target system.

If desired, the remaining tests should also execute properly (e.g., **FirTxPio**) but will not provide much new information. Note that, unless the Tx was specifically designed for it, IrDA Tx's should *NOT* be used with **MastAsk/SlaveAsk**. However, those tests run at low duty cycle, so they are **relatively unlikely to damage the hardware**, even though such damage is definitely possible.

TRANSCEIVER QUALIFICATION TESTING

It is the purpose of Qualification Testing to confirm that the entire IR subsystem is capable of meeting IrDA error rate specifications. Note that an excessive error rate can result in applications difficulty ranging from poor performance in minor cases to an actual inability to either establish or maintain an IR link in extreme cases. The basis of the testing consists of running the **MastFir/SlaveFir**, **MastSir/SlaveSir**, and **WhichRx** programs for an extended period of time (e.g., a 16 hour overnight test interval) and checking the accumulated error statistics. Specific testing strategies will be discussed later in this section, but first, some generalities assuming a 16 hour test interval will be presented.

The WhichRx program should not make a single mistake identifying the module type. Considering the speed of the SIR tests and the required BER, there should be no failures for these tests either. With the FIR tests, a error rate of 1 frame in 3,000 is both common and sufficient. Note that higher failure rates are usually indicative of an Rx noise problem. Usually, the 4 Mbps speed has the highest error rate, which is not surprising considering that it has the widest bandwidth Rx, so the noise will be worst if a relatively flat noise spectral density is assumed.

Note that there are a total of 10 tests, which can take a long time if they are all performed for 16 hours each on a single pair of systems. However, several observations can be made that will materially reduce the test time. First of all, if the FIR tests are intended to test the hypothesis that the frame error rate is 1 in 3,000 or better, then testing 30,000 frames should be sufficient to test or reject this hypothesis, and this will take far less than 16 hours for MastFir/SlaveFir to transfer. The rationale for 1 frame in 3,000 is based on the assumption that a frame error is caused by a single bit error, the frames consist of 2048 bytes sent round trip, and the desired BER is 10^{-8}. It can be argued that 4 Mbps and 1.152 Mbps are by far the most common speeds, and that if these both pass, then a failure at 576 Kbps is extremely unlikely, and so that speed (576 Kbps) does not need to be tested. The combination of these arguments suggests that the FIR tests can be completed in a single day, with WhichRx then run overnight.

By similar reasoning, it can be argued that by far the most common SIR speeds are 9.6 Kbps and 115.2 Kbps, that passing at these speeds makes failing at the other speeds extremely unlikely, and so the other speeds do not have to be tested either. However, the slow speed of SIR transfers makes it difficult to argue that a sufficient number of bits with which to estimate BER can be obtained in a very short time. For example, at 9600 bps, it would take 29 hours to transfer 10^9 bits if the transfers were continuous.