New Technical Notes
Macintosh

Developer Support

Serial I/O Port Q&As
Hardware

Revised by: Developer Support Center May 1993
Written by: Developer Support Center October 1990

This Technical Note contains a collection of Q&As relating to a specific topic—questions you've sent the Developer Support Center (DSC) along with answers from the DSC engineers. While DSC engineers have checked the Q&A content for accuracy, the Q&A Technical Notes don't have the editing and organization of other Technical Notes. The Q&A function is to get new technical information and updates to you quickly, saving the polish for when the information migrates into reference manuals.

Q&As are now included with Technical Notes to make access to technical updates easier for you. If you have comments or suggestions about Q&A content or distribution, please let us know by sending an AppleLink to DEVFEEDBACK. Apple Partners may send technical questions about Q&A content to DEVSUPPORT for resolution.

Which Macintosh models support serial port GPi line
Date Written: 1/16/92
Last reviewed: 1/20/93

I need an updated list indicating which Macintosh models have a functional serial port GPi line.

—

Here's a Macintosh product line GPi support summary:

```
Macintosh Plus         no
Macintosh Classic      no
Macintosh LC           no
Macintosh IIsi         yes
Macintosh Classic II   yes
Macintosh IIci         yes
Macintosh IIfx         yes
Macintosh Quadra       yes
Macintosh IIvx         yes

PowerBook 100          yes
PowerBook 140          yes
PowerBook 145          yes
PowerBook 160          yes
PowerBook 170          yes
PowerBook 180          yes
```

```
PowerBook 210          yes
PowerBook 230          yes
```

We recommend use of the Gestalt call with the gestaltSerialAttr environment selector to locate machines implemented with the GPi line.

Macintosh maximum sustainable serial baud rate

Date Written: 8/28/91
Last reviewed: 10/8/91

When my hardware device sends data to a Macintosh at 57.6 kilobaud, characters can get lost if LocalTalk is on and a file server is mounted. Does Apple know about the problem? Is there a solution?

—

The coherency of a standard serial connection is not guaranteed at any baud rate. Guaranteed delivery requires more complicated protocols like those employed by AppleTalk or by file transfer protocols like Kermit. At high baud rates, particularly over 19.2 kilobaud, a Macintosh serial connection may not be reliable depending on a number of factors.

Every single character that's sent or received by the Macintosh serial port requires an interrupt to alert the processor that a character is available at the receiver or that the transmitter is ready to send a character. A nasty drawback of interrupts is that they can be masked—equal or higher priority interrupts may be in service or other pieces of code running on the machine can disable interrupts altogether. If the period of time interrupts are not serviced becomes too great, the 3-byte receive buffer in the serial chip overflows with incoming data and characters are lost. This is known as a hardware overrun.

AppleTalk works a little differently. When a packet comes across the network, an interrupt alerts the processor to that event. AppleTalk code then disables interrupts and polls in the entire packet without multiple successive interrupts. This is necessary because of the high speed at which data is received: 230.4 kilobaud. The overhead of processing an interrupt for each byte of data would be prohibitive.

A "worst case" situation is that a 603-byte packet (the largest possible AppleTalk packet) sent to the LocalTalk port takes approximately 26 milliseconds to receive. Compare this with asynchronous serial data transmitted at 9600 baud, which would take only 1 millisecond to receive. A very realistic situation arises in which 26 bits could be lost during a concurrent AppleTalk/Serial Device transmission. This possibility is increased when there are routers on the network, which send out large RTMPs (Routing Table Maintenance Packets).

During the time that interrupts are disabled by AppleTalk, AppleTalk code attempts to poll regular serial data as well (from the modem port only) to prevent the hardware overrun problem. Unfortunately, at very high serial baud rates there may not be enough time to do both; characters may be lost anyway.

This same loss of characters may also occur in conjunction with the Sony driver. If a disk is inserted that requires special attention (like formatting), the odds of losing characters is increased. The Sony driver has built-in checks like the AppleTalk driver (although not as

frequent). So you can still lose data at high transfer rates if a disk is inserted during the transfer.

The character dropout also occurs on systems that aren't currently running AppleTalk but are receiving serial transmissions from high-speed devices (57.6 kilobaud) and are performing CPU tasks that require a high amount of memory access (such as a CPU like the Macintosh IIsi or IIci running on-board video in 8-bit mode, or a CPU using virtual memory).

This is simply a performance problem, asking the machine to process more data over two separate ports than the processor speed and hardware architecture can allow. Any number of factors affect serial performance at high baud rates. Turning off AppleTalk helps. Turning on 32-bit addressing helps because it reduces interrupt handler overhead. Turning on VM hurts performance for the opposite reason. Running on a faster machine helps. Running on a Macintosh IIfx or Macintosh Quadra 900 with a serial I/O Processor (IOP) helps a lot because the IOP handles the serial port regardless of whether the main processor is busy doing something else.

There is, however, a way to ensure reception of high-speed serial data without character loss —through the use of a Serial NB card or a third-party NuBus™ card with a dedicated processor necessary to provide higher speeds. These cards in essence operate as dedicated port-handling circuitry. They're able to perform necessary buffering while the processor is servicing other interrupts. This is the reason that network cards such as EtherTalk and TokenTalk can accomplish transactions without data loss; they do at least some of their own buffering until being serviced to avoid interference with other operations such as receiving serial data.

Alternatively, you may want to develop a custom card yourself that exactly fits the needs of your product. In this case you should look into the Macintosh Coprocessor Platform (MCP) and Apple Real-Time Operating System Environment (A/ROSE) as a possible basis of this line of development. Development packages for both these products—the Macintosh Coprocessor Platform Developer's Kit (#M0793LL/B) and the A/ROSE Software Kit (#M0794LL/B)—are available from APDA.

How to keep power on the Macintosh Portable SCC
Date Written: 8/8/91
Last reviewed: 8/13/91

How do I tell the SCC on the Macintosh Portable to stay on (powered)?

—

The Macintosh Portable has a second CPU known as the Power Manager. The Power Manager controls power to each subsystem of the Portable in order to best conserve power on those parts that are not currently in use. The Power Manager is documented in *Inside Macintosh* Volume VI as is a trap (_SerialPower) that can be used by a driver to turn on (and off when done) the SCC.

Conditions degrading 57.6 Kbaud transmission

Date Written: 8/8/91
Last reviewed:

We are experiencing intermittent character dropout at 57.6 Kbaud on a Macintosh IIsi in 256-color mode, but if we drop down to 16-color mode, everything works hunky dory.

MultiFinder on or off makes no difference. What's the connection between the number of colors and the serial port?

—

It turns out that various additions to the system cause performance to degrade, such that serial operation at 57.6 Kbaud cannot reliably be maintained.

There are various factors here, including the video mode (which requires more memory accesses and wait states to produce the same video rate, thus incurring a higher processor overhead). VM, if turned on, exacts an additional performance penalty. These problems are not limited to the Macintosh IIsi; we've had reports that they can also occur on the Macintosh IIci when running the on-board video in 256-color mode without a cache card.

Where to find information on Coaxial card Twinax connector
Date Written:5/7/91
Last reviewed: 7/25/91

Is there any documentation on how to gain access to the Twinax connector on the Macintosh Coaxial card?

—

Andrew/KMW Systems has developed a device interface for the Twinax connector on the Macintosh Coaxial card for their product. For more information, you can contact them at the following address:

Andrew/KWM Systems
6034 West Courtyard Drive
Suite 100
Austin, TX
78730-5014
AppleLink: D0500

For additional documentation, you might want to contact National Semiconductor for a copy of their technical documentation on the Biphase Communications Processor - DP8344A. Also, some Twinax technical documentation might be available from IBM.

Where to get connectors for making Apple-compatible cables
Date Written: 5/3/89
Last reviewed: 12/17/90

Where can I get connectors for making Apple-compatible cables?

—

Here are some suppliers of the mini-DIN connectors. This list is provided for information only—Apple does not endorse any of the vendors.

Hosiden America/Harmonix Sales
Mike Ferrera

10090 Pasadena Avenue, Suite A-2
Cupertino, CA 95014
408-725-2424

CompuCable
Bob Mickey
180 Vallecitos De Oro
San Marcos, CA 92069
619-744-2789
800-222-2332 (outside CA)

TRW
Illinois
Gene Kazmarek aka Kaz
408-720-8727

HB Associates
Hank Lorta
415-487-3933
Union City, CA

Harbor Electronics
Ron Marsilio
650 Danbury Road
Ridgefield, CT 06877
203-438-9625

Also try AMP; they may supply the 8-pin variety.

SCC and AppleTalk timing
Date Written:5/7/91
Last reviewed: 7/25/91

The AppleTalk spec claims a data rate of 230.4 kbaud, which should require a 3.6864 MHz input to the SCC, but RTxCB on the Macintosh carries a 3.672 MHz clock. How does the AppleTalk driver reconcile this and what frequency should I use?

—

The SCC contains a phase-locked loop which is able to lock on and synchronize with AppleTalk transmissions whose clock rates are not exactly to specifications, so everything is fine as long as both ends of the communication are using approximately the same clock frequency. If you are designing your own AppleTalk hardware from scratch, it is easiest to use a 3.6864 MHz oscillator and a Z8530. This has been tested and works just fine.

Macintosh IIfx Serial Switch 'cdev' for SCC direct access

Date Written:6/29/90
Last reviewed: 12/17/90

My application requires direct access to the SCC and I do not find the Macintosh IIfx Serial Switch 'cdev' to be good solution. Is there a way for me to get the support I need from the Serial Driver, or do I need to continue breaking the rules?

—

The Serial Switch 'cdev' is a compromise between not allowing developers any solution or Apple redesigning the serial driver and waiting for developers to implement it. For the moment, this Serial Switch 'cdev' is the best solution available; however, Apple is working on improving the Serial Driver to solve this problem. Unfortunately, it was not possible to produce this improved Serial Driver in time for the Macintosh IIfx introduction.

Where to get detailed Zilog Z8530 SCC documentation
Date Written:3/9/90
Last reviewed: 12/17/90

I need to find documentation for the Zilog Z8530 SCC beyond *Inside Macintosh.* Do you how I can contact Zilog or another vendor for this documentation?

—

You can get documentation on the SCC from the following source:

Advanced Micro Devices (AMD Z8530)
901 Thompson Place
P.O. Box 3453
Sunnyvale, CA 94088