



Chilling out

One common question with serial RapidIO (sRIO) boards is how to support hot swap. Generically speaking, hot swapping should be easy – simply insert or remove the hardware.

The sRIO protocol is targeted for high speed chip to chip communication and data transfer. Compared to gigabit Ethernet and PCI-Express, sRIO has minimal latency, less header overhead, higher efficiency and a guaranteed delivery advantage. Ironically, these key benefits are the crux of the hot swap headache.

The devil is in the detail. If the user doesn't understand the details, then packets stop, ports go into error states, ports stop and, potentially, the system goes down. To achieve guaranteed delivery, sRIO uses Acknowledge Identification (AckID) counters to track each

Taking the frustration out of hot swapping. By Trevor Hiatt and Raymond Ho.

packet and ensure it has been delivered – ultimately to its destination.

AckID is located in the physical layer, not in the packet header. Each port keeps track of its AckID counters and these values are used in the packet acknowledgment control symbol to update the packet transmit status between devices.

Each port has inbound, outbound and outstanding AckIDs. Inbound AckID is the next expected AckID value. When the input port receives a packet, it will use that value as the packet ID, send a response and indicate whether or not the packet with packet ID has been accepted.

Outbound AckID is the output port's next packet AckID value. When the device sends a packet, outbound AckID will increase from 0 to 1, but the outstanding AckID will not increase until it receives a response. On the receiving side, the device will use inbound AckID to

respond. If the packet accepts, then inbound AckID will increase by 1 for the next expected packet. AckID is a 5bit counter and, when it reaches 31, will rollover to 0.

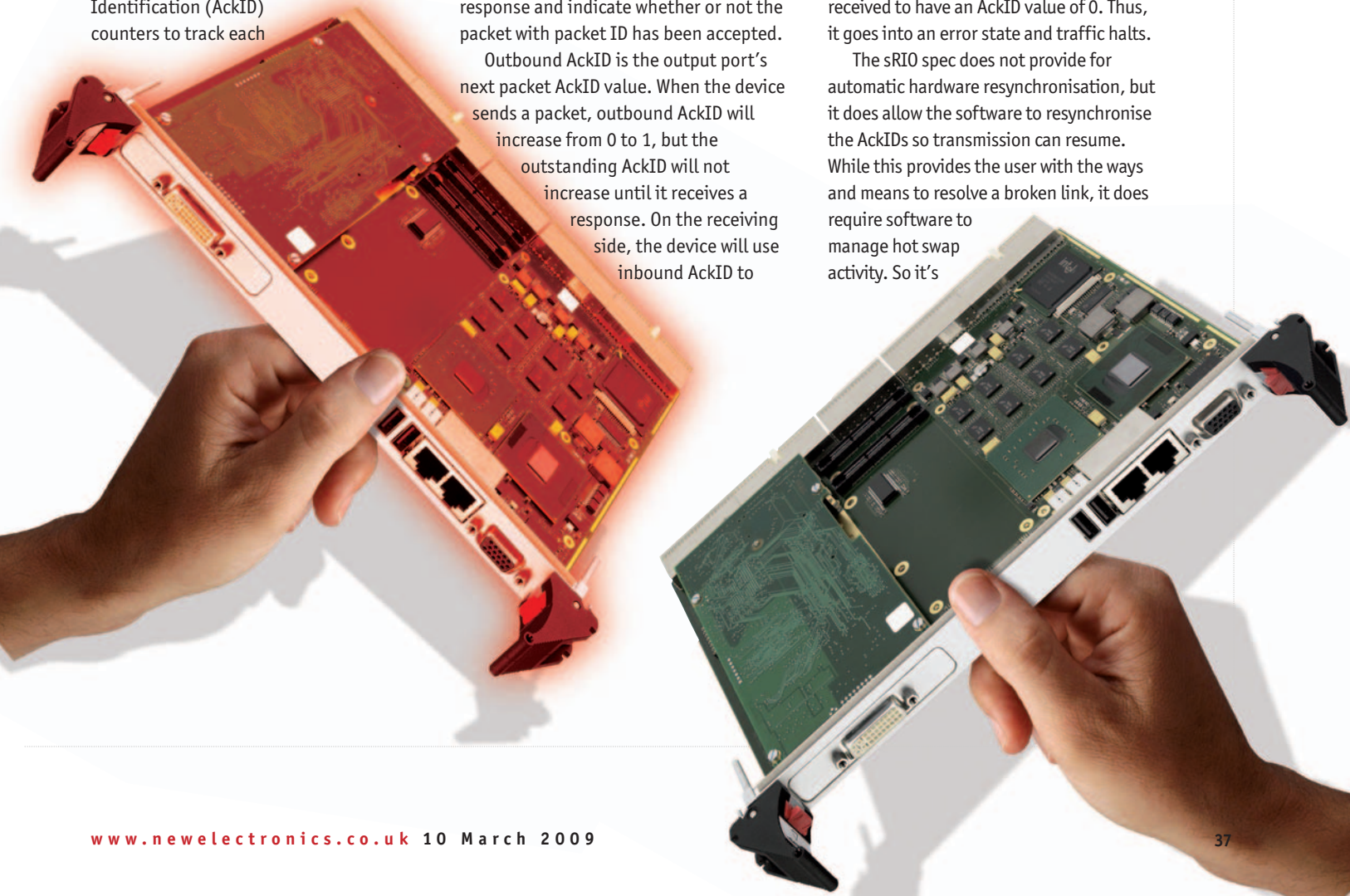
When the sending port receives a packet accept control symbol, the sender will compare response AckID with outstanding AckID. If it matches, it will increase outstanding AckID by 1. If not, it will report 'unexpected AckID error'. The only requirement to support hot swap is simply to match the inbound and outbound AckIDs on both sides of the link.

Expecting a packet

Serial RapidIO ports on the device that has been hot swapped or where power has been cycled or reset, will start to transmit with an AckID value of 0 and will expect to receive packets from the connected port beginning with an AckID value of 0.

If the connected port was previously transmitting and receiving, it is unlikely it will resume transmission with an AckID value of 0 or expect the next packet received to have an AckID value of 0. Thus, it goes into an error state and traffic halts.

The sRIO spec does not provide for automatic hardware resynchronisation, but it does allow the software to resynchronise the AckIDs so transmission can resume. While this provides the user with the ways and means to resolve a broken link, it does require software to manage hot swap activity. So it's





likely those boards that lock up during hot swap in the lab do not have the application layer software support running.

Challenges and solutions

The hot swap challenge is further complicated because not all devices have the same level of support or even the same implementation for Software Assist Error Recovery registers. These registers are optional, so some devices may not allow local reprogramming of AckIDs or querying of a link partner's AckIDs. Further, some devices implement Software Assist Error Recovery registers differently.

System topology also plays a role. Which is the board that gets swapped (and whose AckIDs will zero)? Which remains powered (and whose AckIDs will be non zero)? On which board is the host responsible for reprogramming AckIDs?

In general, there are two solutions

“Generically, hot swapping should be easy – simply insert or remove the hardware.”

Trevor Hiatt and Raymond Ho, IDT

(see figures 1 and 2). Solution 1 requires a hardware reset of inbound/outbound AckID to 0. This is simple and direct, but packets in the output buffers may be lost.

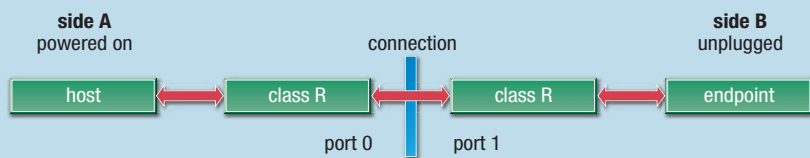
Multiple ways exist to reset AckID. Methods a and b require the link partner to change its AckID to 0 after unplug and before replug. Method c will reset the link partner's AckID automatically.

Method a: For some implementations, disabling and enabling the port will reset inbound and outbound AckID to 0. The local device will also have to set its own AckID values back to zero.

Method b: Hard or soft reset of the full device is not ideal in a live system, since all functionality is affected and traffic will not route. It requires reprogramming the switch after reset. The local device will have to set its own AckID values back to zero.

Method c: For most implementations, 'per port reset' allows the local port to be reset by sending a reset control symbol using the

Figure 1: Hot swap between two class R switches



condition:

two class R switches, one host on one switch. side B will power off or reset during unplug

solution:

- step 1: disable port 0 on side A
- step 2: plug in the link or power up side B
- step 3: enable the port 0 on side A to clear all AckIDs to 0

Software Assist Error Recovery registers.

In solution 2, the Software Assisted Error Recovery Mechanism can be used to change the inbound/outbound AckID, enabling a match to the other side. This needs a host device to execute and specific procedures to ensure AckIDs are matched during hot swap.

In most cases, a 'port disable' state should be used to halt traffic to a live port,

the port before swapping.

Method a: Have an AckID value defined in software that will be used on both sides of the link. Zero is easiest, since newly powered devices will start at 0 by default.

Method b: Use a second bus (such as i²c) connected to the link partner to identify AckIDs, then program these values locally.

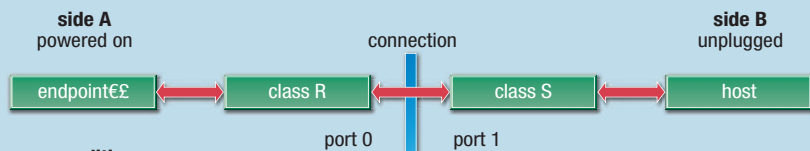
Method c: Send a link request control symbol to the link partner and get its inbound AckID value. Program that value into the switch so the outbound AckID will match the link partner's inbound AckID. The partner needs to send a link request to the switch and use the switch's inbound AckID value for its outbound AckID.

In the figures, Class R devices only have vendor specific per port reset of AckID and no Software Assist Error Recovery option. Class S devices have the sRIO standard Software Assist Error Recovery option and vendor specific per port reset of AckID. Unplugged is the board that will be hot swapped.

Author profiles:

Trevor Hiatt is applications manager, Raymond Ho is an application engineer, with IDT's communications division.

Figure 2: Hot swap between class S and class R switches



condition:

one host on the powered off side. the power will turn off or reset on side B. side A is powered all the time

solution:

- step 1: enable pre port reset option at side A before unplug.
- step 2: after power up, side B sends a reset control symbol to side A. it will reset both side AckID to 0