NAME

iss – low-level module for Tricord System's SCSI host bus adapter

SYNOPSIS

iss@*slot* **c8***bus***,0**

AVAILABILITY

x86

DESCRIPTION

The **iss** module provides low-level interface routines between the common disk/tape (see **cmdk**(7D) and **st**(7D)) I/O subsystem and Tricord System's Intelligent SCSI Subsystem (ISS) controllers. The **iss** module can be configured for hard disk, CD-ROM and streaming tape support for one or more ISS boards. Auto configuration code determines which ISS boards are present and what types of devices are attached to them.

The ISS family of controllers are proprietary multi-channel SCSI controllers. Data transfers to/from the ISS occur at system bus speeds as the controller resides directly on the system bus. The ISS comes in a 2 and 4 bus models and supports either single ended or differential devices. Caching and non-caching models of these controllers are available. Some models of the controller support 7 devices per bus while newer versions support wide SCSI and 15 devices per SCSI bus. Each ISS reserves SCSI id 0 on each bus leaving ids 1-(7 or 15 if wide) available for devices. A 4 channel ISS supports up to 28 (or 60 if wide) devices across the 4 busses. Multiple controllers can exist in the same machine. Up to 4 ISSs can be configured into the ES3000, and up to 6 ISS's can be configured in the ES4000, ES5000 models. A six ISS configuration machine supports up to 360 SCSI devices. This driver supports up to 6 ISS boards.

The controller firmware is driven either by an Intel 386 or 486 DX2 processor. The ISS can support standard physical disk drives, RAID 0, RAID 1, RAID 4, RAID 5 and RAID 1/0 (mirrored stripes) logical disk devices. These logical devices are configured by the DOS utility PowerRaid and are supported by this driver. Their logical nature is transparent to Solaris x86. SCSI tape and CDROM devices are also supported.

Implementation

This driver is implemented as a standard SCSI HBA driver. It is multiprocessor safe and is designed to operate most efficiently in a multiprocessor environment.

Multi-bus support is achieved by making each bus appear as a separate physical host bus adapter as far as Solaris x86 is concerned. Each possible bus of each possible ISS is configured via the **iss.conf** file. Groups of busses forming a physical ISS are controlled internally by the driver.

There are two primary code paths thru the driver: one for physical devices (disk, tape, cdrom) and another for logical disk devices raid **0,1,4,5,1/0**. Tables are built at initialization time by the driver that allow easy determination (based on the SCSI address) as to whether a device is physical or logical.

Device Naming Convention

This sub-section describes how the special device files in the directories **/dev/dsk** and **/dev/rdsk** are used on an ISS system. Since these are multichannel controllers, each bus of each controller has a different controller number. So for a system containing 2 4-channel ISS SCSI controllers, ISS 0 busses 0-3 may have controller 0-3 devices, respectively, and ISS 1 busses 0-3 may have controllers 4-7 devices, respectively.

A system can be installed using another vendor's SCSI controller as the boot controller, Adaptec for example. In this case the Adaptec controller devices assume the controller 0 names, and ISS busses have controllers from 1 onward.

If no devices reside on a given bus of an ISS then the controller number associated with that bus will be assigned to the next bus.

Logical devices are known to the operating system by the name of the physical devices that resides at the lowest bus and lowest id of all the physical devices that make up that logical device. For example: If a 2 drive mirror consisted of the two physical drives at bus0 id2 and bus3 id7 the device would be known by the OS as **c0t2d0s2**; **c3t7** would not exist. No device file would be built.

Logical Device Implementation

Inquiry and capacity commands are simulated by the driver for logical devices based on logical device information provided by the ISS that resides in ISS dual port memory. The fact that a device is logical or physical is kept hidden from any levels of software above this HBA driver.

Logical boot devices (mirrors, stripes, etc.) are allowed. If Solaris x86 is installed on a logical boot device then that logical boot device must include the physical disk device that resides on bus0 id 1 of the ISS in the lowest system bus slot. **[9-F]**.

Interrupts

Each physical ISS generates a unique IRQ. The IRQ is directly tied to the system bus slot that the board resides in. This is informational only. The user need not do anything configuration wise to accomplish this. The slot, **irq** association is as follows:

System Bus Slot	IRQ
9	23
10	22
11	21
12	20
13	19
14	18
15	17

CONFIGURATION

The driver attempts to initialize itself in accordance with the information found in the configuration file, **/kernel/drv/iss.conf.** No relevant user configurable items are in this file. Do *not* modify **/kernel/drv/iss.conf**.

SEE ALSO

driver.conf(4), **scsi**(4), **cmdk**(7D), **st**(7D)

BUGS

Once a disk device has been installed with the Solaris x86 OS, the ISS cannot be moved to a different slot or the system will not boot. The bootable disk drive is ISS slot dependent.