

**Release Notes**  
**NetCCN™ for cPCI**  
**Firmware Revision v1.7.1**  
**Release Date: 2004 July 1**

---

***Features Changed***

- Added additional debouncing to sampled hotswap signals (such as board presence, board healthy, etc) during NetCCN initialization to help filter noise that may cause the NetCCN to initialize a cPCI board to an incorrect state.

***Known Issues***

- There is a known issue with Sun CP2140 not asserting its board healthy signal per the PICMG 2.1 hotswap specification when it is initially powering up. Specifically, the CP2140 will not assert its board healthy signal until after it is taken out of reset and has performed its POST. The NetCCN uses a hotswap state machine as defined in the PICMG 2.1 specification which requires the reset signal to remain asserted by the NetCCN after the assertion of the board select signal until the healthy signal is asserted by the cPCI board. Since the CP2140 does not conform to this requirement, the NetCCN keeps it powered off and in reset. There are several workarounds to this issue. One workaround is to set the healthy mask ('hlthymask' CLI command) for the CP2140's slot. This will allow the NetCCN to automatically power the board up. Another workaround is to disable the automatic hotswapping feature altogether on the NetCCN (using the 'hotswap disable' CLI command) and manually turn the CP2140 on and off as desired.
- There has been a single report that an event callback was received by a CCNlib application program shortly after a NetCCN reset where the FAULTNOW register contained a patently invalid value. However, the FAULTNOW register shortly thereafter corrected itself. No additional event callbacks were received that would correspond to any of the indicated fault conditions. Because of this, a suspected cause for the invalid FAULTNOW register value is a corrupt CCNtalk packet. However, to confirm and isolate this as the actual cause, CCNlib log messages in /var/adm/messages from a local CPU and from a remote CPU should be saved and forwarded to [support@ccpu.com](mailto:support@ccpu.com) if this problem is observed again. This information would be instrumental in determining the source of the corruption (e.g. corruption over CCNtalk serial link versus corrupt fault values within NetCCN software).