

md1200/md1220/md3200

Notes about the Dell Storage Array md1200/md1220/md3200

- md1200 disk shelf

md1200 disk shelf

I had a lot of problems getting the fans to spin down. To start with I had to buy a special cable, use these serial settings: RS232, 38400-8-n-1, and update the firmware, replace the EMMs after a failed _boot command.

The most useful commands have been:

```
_ver
_boot > File Transfer > xmodem > send
set_temp <#> <c>
_temp_rd
set_speed <%> | _shutup <%>
```

<https://forums.servethehome.com/index.php?threads/fun-with-an-md1200-md1220-sc200-sc220.27487/>

```
BlueDress.106.000 >devils
devils
asset_tag Set or Display the Asset Tag: asset_tag {setvalue}
asd_offset Set or Display the Auto-Shutdown Offset value: asd_offset {setvalue}
broadcast Send Broadcast SES Message: broadcast
chassistype Display and or Set Chassis Type: chassistype <0 = Blue Devil !0 = Red Devil>
clear_eel Clear Event Error Log: clear_eel
clear_temp Remove override of Temperature: clear_temp <sensor>
dbs Database Read : dbs <page>
devils Print the Help Screen
drive_led Write drive led: <logicaldrive> <data>
eepromdump EEPROM Dump: eepromdump <port><addr><size in K(1,2,4,8,..512)>
eepromfill EEPROM Fill: eepromfill <port><addr><size in K(1,2,4,8,..512)><pattern>
eepromwrite EEPROM Write: eepromwrite <port><addr><size in K(1,2,4,8,..512)><offset><length><pattern>
fanlog Fan fault count for each power supply fan [8 per unit]
fpgadisable Put FPGA in Slave Mode: fpgadisable>
fpgaenable Put FPGA in Master Mode: fpgaenable>
fpga_rd FPGA Access: fpga_rd <Register> <#bytes>
fpga_wr FPGA Access: fpga_wr <Register> <data>[<Register> <data> ...]
fru_display Display FRU Status: fru_display
```

fru_clear Clear Fru: fru_clear [0-SIM0, 1-SIM1, 2-PBP, 3-PS0, 4-PS1 5-SBP]
 fru_download Download Fru: fru_download [0-SIM0, 1-SIM1, 2-PBP, 3-PS0, 4-PS1 5-SBP]
 fru_read Read Fru: fru_read [0-SIM0, 1-SIM1, 2-PBP, 3-PS0, 4-PS1 5-SBP]]
 get_time get encl time: get_time
 gpio_rd Read a GPIO: gpio_rd <number>
 heart_beat SIM Heartbeat Control: heart_beat [0=off !0=on] <timeout>
 isim_debug Change or view isim stats: <data> <0 - Disable; 1 - Enable>
 l4_test L4 integration manufacturing diag: l4_test
 lm75 LM75 Read Access: lm75 <Slave Address>
 lm75_rd LM75 Read Access: lm75_rd <Slave Address> <Register>
 lm75_wr LM75 Write Access: lm75_wr <Slave Address> <Register> <1 byte>
 log_ipmi Log an IPMI Event: log_ipmi <Code> <Type> <Sensor> <EV0> <EV1> <EV2> <EV3>
 max6654 Display MAX6654 Registers: max6654 <i2c port> <slave addr>
 noise Write audible alarm: <data>
 nvramread8 Read NVram 32bit area: nvramread <address> <length>
 nvramread Read NVram 32bit area: nvramread <32bit address> <length>
 nvramwrite8 Write NVram 32bit area: nvramwrite <address> <data> [<32bit address> <data> ...]
 nvramwrite Write NVram 32bit area: nvramwrite <32bit address> <data> [<32bit address> <data> ...]
 page_a Display drive SAS Address: page_a
 ps_status Get P/S Module Status: ps_status <l(eft)/r(ight)>
 ps_cap Get P/S Module capability: ps_cap <l(eft)/r(ight)>
 ps_clear Clear P/S Module Status: ps_clr <l(eft)/r(ight)>
 ps_page Get P/S Module Status: ps_status <l(eft)/r(ight)>
 ppid Set or Display PPID: ppid {fruNumber} {setvalue}
 prompt Prompt on/off
 rd_8 8-bit Read: rd_8 <address> <# of 8 bit words>
 rd_16 16-bit Read: rd_16 <address> <# of 16 bit words>
 rd_32 32-bit Read: rd_32 <address> <# of 32 bit words>
 reset_peer Reset other SIM using GPIO <1-reset peer>
 reset Reset ARM using Watch Dog timer
 rev SIM Firmware and Diagnostc Revision
 sas_address Display SAS Address from Phys: (option for magic addr <1>)
 sbb_status Set SBB status: sbb_set <default-print status, 0-set good, 1-set failed>
 scratchpad Display Location of Memory Test Area: scratchpad
 service_tag Set or Display the Service Tag: service_tag {setvalue}
 ses_page Display SES Page: ses_page <page> <buffer size>
 set_speed Sets Fan Speeds: set_speed <0-100%> 20 default
 set_temp set encl temp: set_temp <sensor> <temp> (-55 to 125 degrees C)
 set_thres Set P/S Module Fan Speed Threshold: set_thres <l(eft)/r(ight)> <speed code 0..15>
 shelf_led Write shelf led: <data> <0 - Disable; 1 - Enable>

```

twi_dis    TWI device discovery: twi_dis
twi_rd     TWI device byte read: twi_rd <port ID> <address> <# of bytes - 0xff max>
twi_stats  Dump twi statistics: twi_stats [clear]
twi_wr     TWI device byte write: twi_wr <port ID> <address> {0xff bytes max}
twi_wr_rd  TWI device wr/rd: twi_wr_rd <port ID> <address> <#read bytes> <write data>
wr_8      8-bit Write: wr_8 <address> <data> [<address> <data> ...]
wr_16     16-bit Write: wr_16 <address> <data> [<address> <data> ...]
wr_32     32-bit Write: wr_32 <address> <data> [<address> <data> ...]

```

BlueDress.106.000 >_devils

```

_devils
_boot      Download the boot image (FW image 1 and 2 erased)
_clrphyerr Clear PHY error counter(s) <phy Num (invalid PHY Num=all)>
_date      Date: date
_debugpage2 Dispalys Page 2 data when Host read it : 0 = OFF !0 = Active
_devils    Print the Extended help
_dwd_reset Disable WatchDog reset: dwd_reset
_download  Down load code using Xmodem: Region[0-9] Offset Erase[y/n]
_download_fpga Down FPGA load code using Xmodem: _download_fpga
_drive_pres Return Drive presense: _drive_pres
_ecc       Display ECC counts: ecc {clear}
_ema_poll  Turns on or off the analysis polling. Disables SES & LED update
_erase     Erase F/W Region: erase [0..10] NOTE: This can mess up your code!
_event     Event log test
_fan_ctrl_thrd Write fan control parameters in C: fan_ctrl_thrd <M> <H> <SIM offset> <hysteresis>
_flashdump Dump Flash: flashdump <offset> <length>
_fpgaread  Read FPGA Register: fpgaread <register> <length>
_fpgawrite Write FPGA Register: fpgawrite <register> <data> [<register> <data>...]
_fail_ts   Fail temperature sensor: fail_ts <sensor>
_flashpeer Flash Peer: <0=ACTIVE ; 1=BOOT>
_gpio_setting Read a GPIO Setting: gpio_settings <0..7>
_history   Show CLI History: history
_hotswap   Enable/disable Hotswap
_icid_clear Clears the ICID value to default.
_isim_msg  Send message to ISIM thread
_ledconfig Show LED Configuration Settings
_ledmode   Set Mode for led: ledmode <number><direction><state>
_ledread   Read LED GPIO value: ledread <number>
_ledset    Set LED GPIO Value: ledset <number><state>
_lm75_trip LM75 Interrrupt Control: lm75_trip <sensor><state>

```

_loadcpld Load xsvf file from flash into CPLD
 _map Display SES Sensor Data: map
 _phy_info Display SAS phy information
 _ps_pmb_test Test P/S Module PMBUS commands: ps_pmb_test <l(eft)/r(ight)>
 _psfup P/S Module Firmware update: psfup <l(eft)/r(ight)>
 _queue Dump the Message Queue usage: queue
 _quick Quick regression: quick
 _rdcam Read CAM Address contents <address (0-1024)>, <display count>
 _rdtxphy Display current SASTX2G Phy settings <phy Num (invalid PHY Num=all)>
 _rdrxphy Display current SASRX2G Phy settings <phy Num (invalid PHY Num=all)>
 _rdphyerr Display PHY error counters <phy Num (invalid PHY Num=all)>
 _rdled LED Control Register Read: rdled <offset> <# of 32 bit words>
 _runtime Down load code using Xmodem to the non-active region: runtime
 _set_cid Set company ID of ELI: cid_set [0-From Exp, 1-Custom <32bit data>]
 _shutdown Invoke a thermal shut down sequence: _shutdown <why>
 _shutup Slow down fans: _shutup <0-100%> 20 default
 _slotled Control Slot LEDs: slotled <slot><pattern> (slot > MaxDrive = 'all')
 _splitforce Control Split Override: splitforce <0-switched; 1-ForceJoin; 3-ForceSplit>
 _ssc_control Turn SSC on or off <enable 1, disable 0>
 _stack Show Stack Usage: stack
 _temp_rd Read current temperature values
 _temp_thrd Write temp thresholds: temp_thrd <HS> <HC> <HW> <LW> <LC> <LS>
 _test_stub test_stub
 _thread Display Threadx Information: thread
 _timer Display Timer information: timer
 _trace Display Trace Log [- for Tail] [+ for Head]: trace [-]count
 _traceclear Clear trace log: traceclear
 _twi_hang Hang the TWI bus for testing purposes only
 _ver Version Information: ver
 _wdt WatchDog test: wdt <who>
 _who Who's home (installed): who
 _wrphy Write SAS2G1-3 Phy setting <phy number,Dword>
 _wrphyall Write all SAS2G1-3 Phy setting <uses PHY setting table>
 _zone_mask Display Zone Mask for all PHYs