

Kernel driver mlxreg-fan

Provides FAN control for the next Mellanox systems:

- QMB700, equipped with 40x200GbE InfiniBand ports;
- MSN3700, equipped with 32x200GbE or 16x400GbE Ethernet ports;
- MSN3410, equipped with 6x400GbE plus 48x50GbE Ethernet ports;
- MSN3800, equipped with 64x1000GbE Ethernet ports;

Author: Vadim Pasternak <vadimp@mellanox.com>

These are the Top of the Rack systems, equipped with Mellanox switch board with Mellanox Quantum or Spectrume-2 devices. FAN controller is implemented by the programmable device logic.

The default registers offsets set within the programmable device is as following:

pwml	0xe3
fan1 (tacho1)	0xe4
fan2 (tacho2)	0xe5
fan3 (tacho3)	0xe6
fan4 (tacho4)	0xe7
fan5 (tacho5)	0xe8
fan6 (tacho6)	0xe9
fan7 (tacho7)	0xea
fan8 (tacho8)	0xeb
fan9 (tacho9)	0xec
fan10 (tacho10)	0xed
fan11 (tacho11)	0xee
fan12 (tacho12)	0xef

This setup can be re-programmed with other registers.

Description

The driver implements a simple interface for driving a fan connected to a PWM output and tachometer inputs. This driver obtains PWM and tachometers registers location according to the system configuration and creates FAN/PWM hwmon objects and a cooling device. PWM and tachometers are sensed through the on-board programmable device, which exports its register map. This device could be attached to any bus type, for which register mapping is supported. Single instance is created with one PWM control, up to 12 tachometers and one cooling device. It could be as many instances as programmable device supports. The driver exposes the fan to the user space through the hwmon's and thermal's sysfs interfaces.

/sys files in hwmon subsystem

fan[1-12]_fault	RO	files for tachometers TACH1-TACH12 fault indication
fan[1-12]_input	RO	files for tachometers TACH1-TACH12 input (in RPM)
pwml	RW	file for fan[1-12] target duty cycle (0..255)

/sys files in thermal subsystem

cur_state	RW	file for current cooling state of the cooling device (0..max_state)
max_state	RO	file for maximum cooling state of the cooling device