

## Introduction

For systems using microprocessors or computers there are usually numerous power supplies. If a power supply fails the power manager circuits may, as a minimum, force a shutdown. For maintenance and troubleshooting it is very desirable to know which power supply failed and the type of failure condition (over-voltage or under-voltage). This reference design presents a solution that records the supply fault condition in non-volatile memory so the fault(s) can read back at a later time. This solution is fast, reliable, and cost effective because it is based on a Power Manager II, MachXO™ or LatticeXP2™ and non-volatile SPI Flash memory.

## Theory of Operation

This fault logger reference design uses a Lattice Power Manager II device to monitor the voltage levels in the system. The Power Manager II device is designed to monitor and control different power supplies within a system and has an on-board analog-to-digital converter. The user can set high and low voltage alarm points within the device and these can then be used to initiate different control actions of the user's choosing. For this design a voltage alarm will cause a fault status output to be driven high and the status of all the voltage monitor channels to be output on four data status lines. This design uses the ispPAC®-POWR1220AT8 Power Manager II device but could also be adapted to the ispPAC-POWR1014A device.

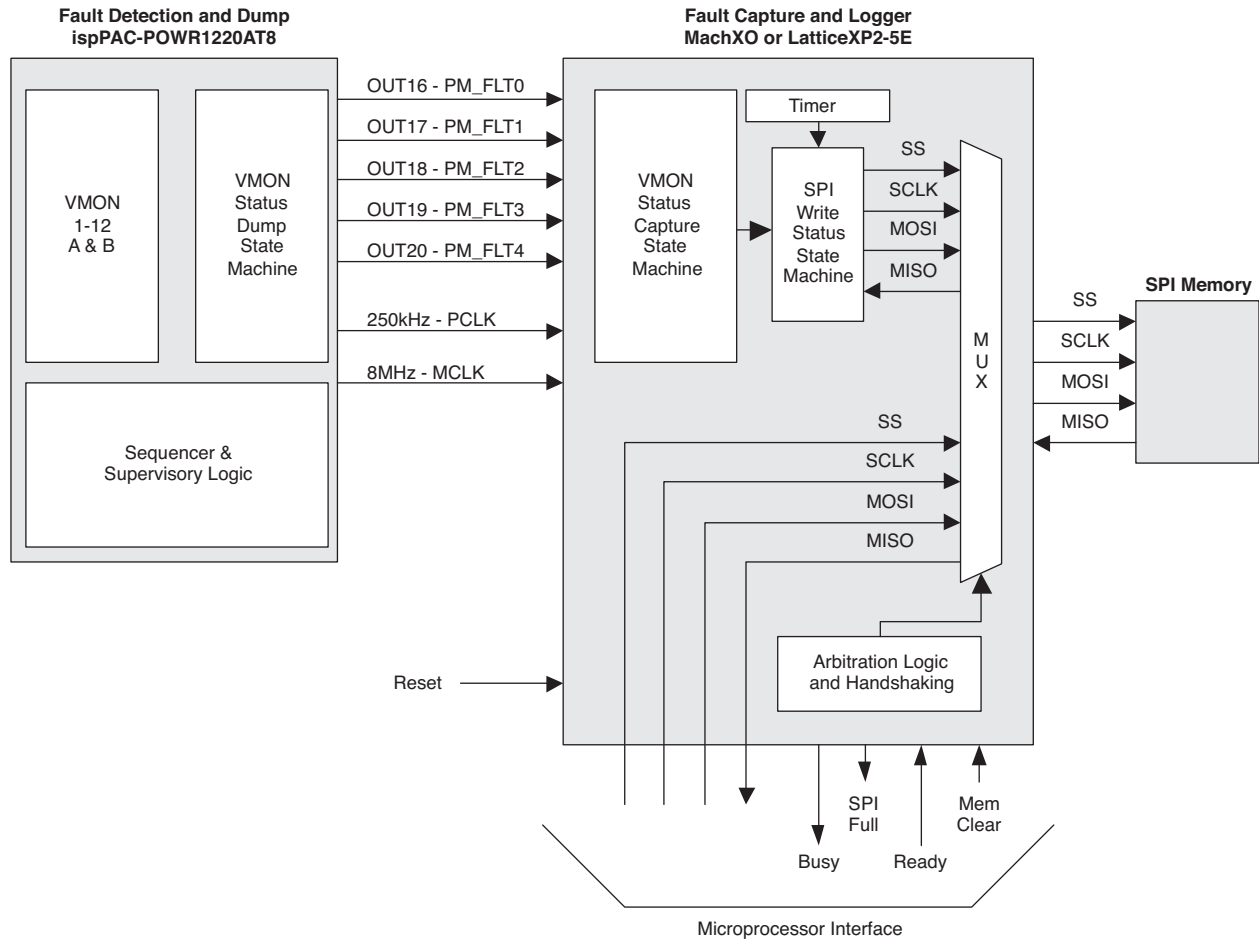
The fault status output and the four data status lines are connected to a MachXO or LatticeXP2 device which then captures the fault data, formats it, and writes the formatted data to a SPI Flash memory for later retrieval.

The sequence of events can be summarized as follows:

1. The ispPAC-POWR1220AT8 detects a fault on one or more of the VMON inputs and dumps the status of all the VMON inputs to the MachXO or LatticeXP2 device.
2. The dump of VMON status is implemented using supervisory logic equations and happens automatically using outputs and clock pins of the Power Manager II.
3. The MachXO or LatticeXP2 implements a receiver state machine to capture the VMON status.
4. The MachXO or LatticeXP2 adds a time stamp to the VMON status information, sends the commands and writes the data to a SPI Flash memory device.

A block diagram of the Fault Logging Reference Design is shown in Figure 1.

Figure 1. Fault Logging Design Block Diagram



## Design Details

Figure 3 shows the details of the VMON status dump from the Power Manager II to the MachXO or LatticeXP2 device and the transfer of data from the MachXO or LatticeXP2 device to the SPI Flash memory. The VMON status dump is generated within the Power Manager II device using an internal 250 KHz clock (fixed) which also drives the PCLK signal. The PCLK signal is used to drive the VMON Status Capture state machine inside the MachXO or LatticeXP2 device.

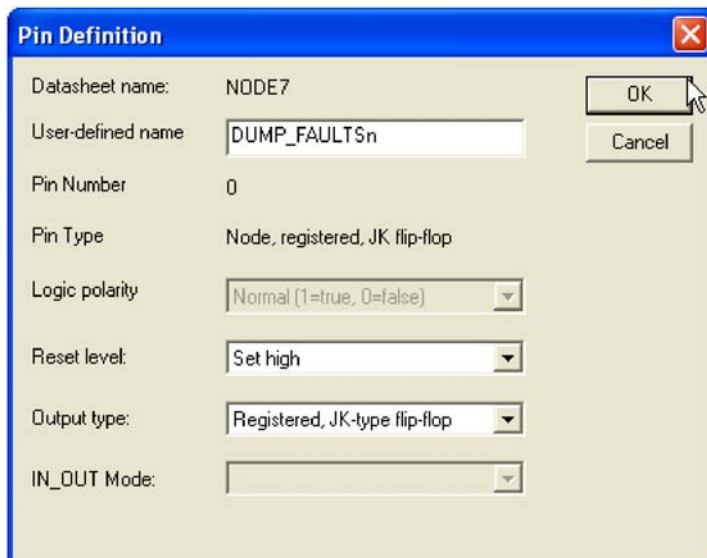
As shown in Figure 1, the fault logging process is divided between two devices; the Power Manager II detects and dumps the faults to the CPLD that captures the faults. The CPLD then writes them to a standard non-volatile SPI memory. The CPLD also provides arbitration logic, a timer, and a MUX interface to the SPI memory for microprocessor support. The details of the CPLD design are presented in RD1092 [Fault Logging Twelve Power Supplies using the MachXO](#). This document discusses the details of the Power Manager ispPAC-POWR1220AT8 design file RD1072\_Fault\_Logging.PAC, which is targeted for demonstration on the Power Manager Hercules Evaluation Board.

The heart of this design is contained in the supervisory logic equations in the LogiBuilder window of PAC-Designer® (see Listing 2). One set of equations is used to implement a three-bit binary up-counter and the second set of equations is used to dump all the VMON status values to the CPLD. When the counter is active the four-bit bus of output pins is updated with the VMON status values on the rising edge of the PLD clock (250 kHz). The rising edge of the fifth output (OUT20\_PM\_FLT4) is used to trigger the CPLD to capture the VMON status.

Equations 0 to 6 implement a three-bit binary up-counter. This counter is reset to a value of 3 and then counts up using the sequence; 4, 5, 6, 7, 0, and 1 at the rate of the PLD clock. The counter is enabled when NODE7 (named DUMP\_FAULTSn) is set low in the main sequence (see Listing 1). When the counter reaches a value of 2 it sets DUMP\_FAULTSn high which resets the counter and prevents multiple status dumps.

A key element to this design is the flag or internal signal DUMP\_FAULTSn. It is through this signal that the main sequence tells the equations to dump the VMON status. This node is named using the Pin Definition dialog (see Figure 2) from LogiBuilder's PINS window. In this design the main sequence monitors VMON9 (the slide potentiometer on the Power Manager Hercules Evaluation Board) and sets DUMP\_FAULTSn low if the voltage is in error.

**Figure 2. Naming NODE7 in the PINS Window**



Equations 7 to 11 decode the three-bit binary counter and combine the VMON status to provide the fault reporting to the fault logging device. Equations 7 to 10 are combinatorial and provide a four-bit data bus to the CPLD with the VMON status. Each OR branch of the equation decodes the counter value and combines a unique VMON status so that all 24 VMON status are transferred to the CPLD in six PLD clocks (24  $\mu$ s). The CPLD uses the negative edge of the PLD\_CLK to latch the four VMON status presented on the four-bit bus (see Figure 3).

Equation 11 simply maps the most significant bit of the counter (Cntr2) to the output OUT20\_PM\_FLT4. This output transitions from low to high to trigger the CPLD to capture the status dump. The status dump and binary counter are shown in Figure 3 with the PLD\_Clock.

The main sequence (see Listing 1) is a very simple monitoring loop. Steps 0 to 1 start-up and initialize the Power Good LED in the off state and reset the DUMP\_FAULTSn flag. Steps 2 to 6 monitor the status of VMON9 to control the Power Good LED and the DUMP\_FAULTSn flag. When VMON9 is between the trip points the Power Good LED is on. When VMON9 is outside the windowed trip points, the Power Good LED is off and the DUMP\_FAULTSn flag is activated.

In the Analog Input Settings dialog (from the main schematic window) VMON9 is configured in Window mode with the lower trip point of 2.0V and the upper trip point of 3.0V. On the Power Manager Hercules Evaluation Board VMON9 is connected to the slide potentiometer to simulate either an under or over voltage situation.

This design is easily modified to include any number of other VMON or input conditions to trigger the fault dump by adding to the Boolean logic in Step 4. Other sequences and controls could also be added before Step 2 based on the design requirements.

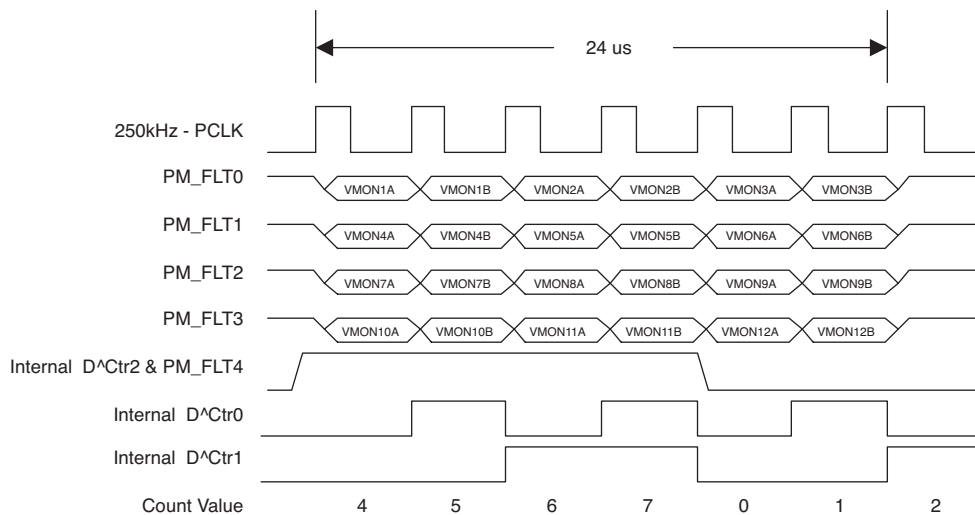
Normally, the Logical Signal Name for VMON9\_A would be renamed to indicate the signal source and the window mode of operation such as VM9\_SLIDE\_POT\_OK. However, this was not done in this design for two reasons. First is so that all the documentation uses the same signal name to clarify the concept. Secondly, to highlight the design flow used. Normal design flow suggests naming the inputs and outputs before starting on the sequence or supervisory logic equations. However, this particular design can be entered faster and with less chance for error by copying the equations from the Boolean\_EQ.txt file and pasting them into the equations dialog box. After the equations are entered the Logical Signal Names can be edited and the Sequence and Equations will automatically be updated.

For convenience, RD1092, [Fault Logging Twelve Power Supplies using the MachXO](#), uses the 8 MHz MCLK signal from the Power Manager II to drive the SPI memory write operations. (The MCLK signal from the Power Manager II is a fixed rate clock.) A user design could use a faster clock signal if desired.

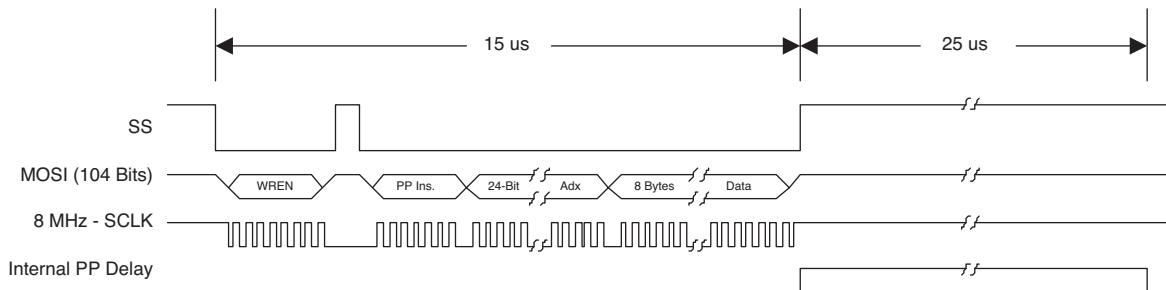
The entire operation takes less than 100 microseconds to store the fault record to the SPI Flash memory as shown in Figure 3.

**Figure 3. Waveform Diagrams of Fault Logging Design**

**Fault Logging Status Dump Waveforms**



**Fault Logging SPI Page Program Waveforms**



**Listing 1. LogiBuilder Sequence**

```
// Reset, turn off Power Good LED and reset Fault Dump flag.
Step 0    Wait for AGOOD
          OUT15_PWR_GOOD = 1, DUMP_FAULTSn = 1

// Optional additional startup delay.
Step 1    Wait for 15.36ms using timer 1

// Wait for VMON9 to be more than 2V and less than 3V
// VMON9 is the slide potentiometer on the Hercules Demo Board.
Step 2    Wait for VMON9_A

// Turn on the Power Good LED
Step 3    OUT15_PWR_GOOD = 0,

// Wait for VMON9 to be less than 2V or more than 3V
// a fault condition.
Step 4    Wait for NOT VMON9_A

// Turn off Power Good LED and start the Fault Dump.
Step 5    OUT15_PWR_GOOD = 1, DUMP_FAULTSn = 0,

// Return to monitoring VMON9.
Step 6    Go to step 2

// A shutdown sequence is not used in this design.
Step 7    Begin Shutdown Sequence
Step 8    Halt (end-of-program)
```

**Listing 2. Supervisory Logic Equations**

```
// Equations 3 - 8 provide a three-bit binary up-counter that
// is enabled by DUMP_FAULTSn.
EQ 0     Cntr0.D = NOT Cntr0
EQ 1     Cntr1.D = ( Cntr0 AND NOT Cntr1 ) OR ( NOT Cntr0 AND Cntr1 )
EQ 2     Cntr2.D = ( NOT Cntr2 AND Cntr0 AND Cntr1 )
          OR ( Cntr2 AND NOT ( Cntr0 AND Cntr1 ) )

// The binary counter is reset to a count of 3 when the
// DUMP_FAULTSn flag is true and counts 4,5,6,7,0,1,2 when
// the flag is low.
EQ 3     Cntr0.ap = DUMP_FAULTSn
EQ 4     Cntr1.ap = DUMP_FAULTSn
EQ 5     Cntr2.ar = DUMP_FAULTSn

// Equation 6 automatically sets DUMP_FAULTSn high to reset
// the binary counter; after the VMON status bits have been
// dumped (at the count of 2).
EQ 6     DUMP_FAULTSn.ap = NOT Cntr0 AND Cntr1 AND NOT Cntr2

// Equations 7 - 10 are combinatorial logic that decode the counter
// value to select the VMON status to output. The four bits of VMON
// status are updated on the rising edge of PLD_CLK.
```

```

// VMON1 - VMON3 status bits are decoded to fault bit zero.
EQ 7    OUT16_PM_FLT0 =
        ( VMON1_A AND NOT Cntr0 AND NOT Cntr1 AND      Cntr2 )
        OR ( VMON1_B AND      Cntr0 AND NOT Cntr1 AND      Cntr2 )
        OR ( VMON2_A AND NOT Cntr0 AND      Cntr1 AND      Cntr2 )
        OR ( VMON2_B AND      Cntr0 AND      Cntr1 AND NOT Cntr2 )
        OR ( VMON3_A AND NOT Cntr0 AND NOT Cntr1 AND NOT Cntr2 )
        OR ( VMON3_B AND      Cntr0 AND NOT Cntr1 AND NOT Cntr2 )

// VMON4 - VMON6 status bits are decoded to fault bit one.
EQ 8    OUT17_PM_FLT1 =
        ( VMON4_A AND NOT Cntr0 AND NOT Cntr1 AND      Cntr2 )
        OR ( VMON4_B AND      Cntr0 AND NOT Cntr1 AND      Cntr2 )
        OR ( VMON5_A AND NOT Cntr0 AND      Cntr1 AND      Cntr2 )
        OR ( VMON5_B AND      Cntr0 AND      Cntr1 AND NOT Cntr2 )
        OR ( VMON6_A AND NOT Cntr0 AND NOT Cntr1 AND NOT Cntr2 )
        OR ( VMON6_B AND      Cntr0 AND NOT Cntr1 AND NOT Cntr2 )

// VMON7 - VMON9 status bits are decoded to fault bit two.
EQ 9    OUT18_PM_FLT2 =
        ( VMON7_A AND NOT Cntr0 AND NOT Cntr1 AND      Cntr2 )
        OR ( VMON7_B AND      Cntr0 AND NOT Cntr1 AND      Cntr2 )
        OR ( VMON8_A AND NOT Cntr0 AND      Cntr1 AND      Cntr2 )
        OR ( VMON8_B AND      Cntr0 AND      Cntr1 AND NOT Cntr2 )
        OR ( VMON9_A AND NOT Cntr0 AND NOT Cntr1 AND NOT Cntr2 )
        OR ( VMON9_B AND      Cntr0 AND NOT Cntr1 AND NOT Cntr2 )

// VMON10 - VMON12 status bits are decoded to fault bit three.
EQ 10   OUT19_PM_FLT3 =
        ( VMON10_A AND NOT Cntr0 AND NOT Cntr1 AND      Cntr2 )
        OR ( VMON10_B AND      Cntr0 AND NOT Cntr1 AND      Cntr2 )
        OR ( VMON11_A AND NOT Cntr0 AND      Cntr1 AND      Cntr2 )
        OR ( VMON12_B AND      Cntr0 AND      Cntr1 AND NOT Cntr2 )
        OR ( VMON12_A AND NOT Cntr0 AND NOT Cntr1 AND NOT Cntr2 )
        OR ( VMON12_B AND      Cntr0 AND NOT Cntr1 AND NOT Cntr2 )

// Equation 11 is also combinatorial but only outputs the value of
// the most significant bit of the counter.  The rising edge of this
// output is used to trigger the CPLD to capture the VMON status and
// thus the faults.  Based on the simple logic; Cntr2 is optimized
// by the fitter to OUT20_PM_FLT4.  So the Cntr2 signal is missing
// from the simulation and fitter report.
EQ 11   OUT20_PM_FLT4 = Cntr2

```

## Simulation and Verification

The simulation of this design is done using a reduced value for Timer1 for faster and smaller simulation files. The timer value is changed from 15.3 ms to 104  $\mu$ s. Figure 4 shows the simulation result for an under-voltage fault and Figure 5 shows the simulation result for an over-voltage fault.

Figure 4. VMON9 Under-Voltage Simulation Fault Dump

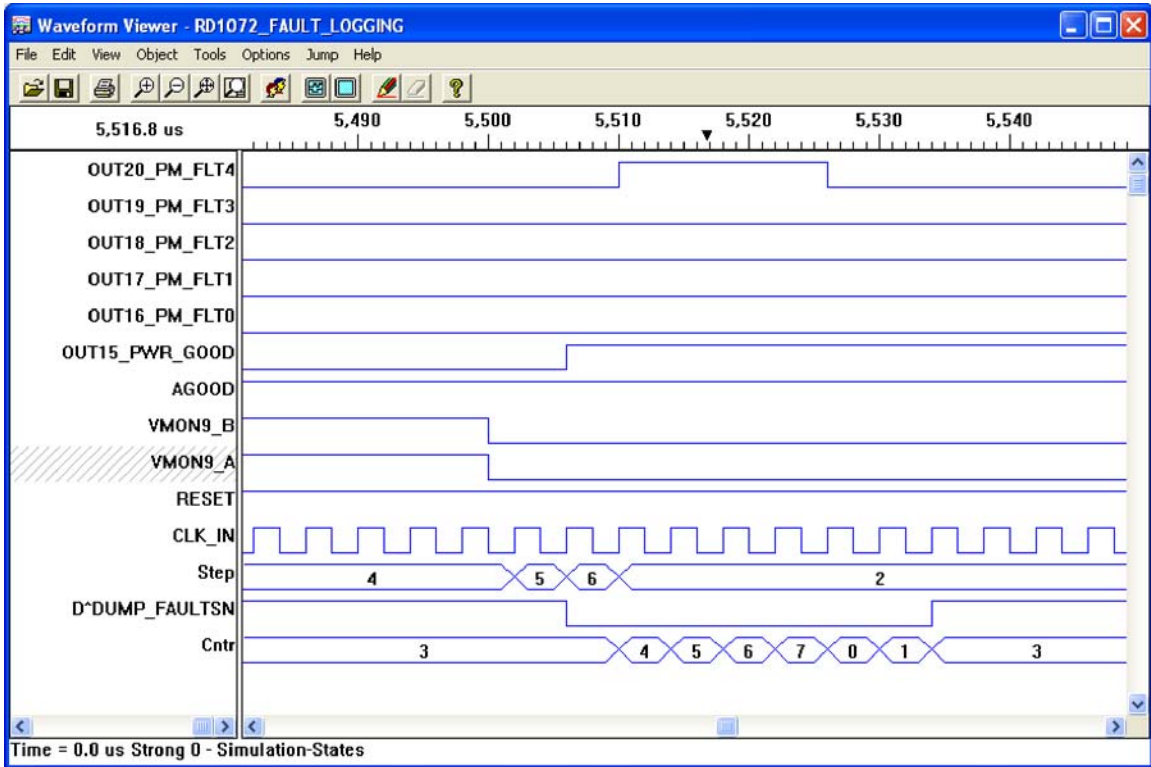
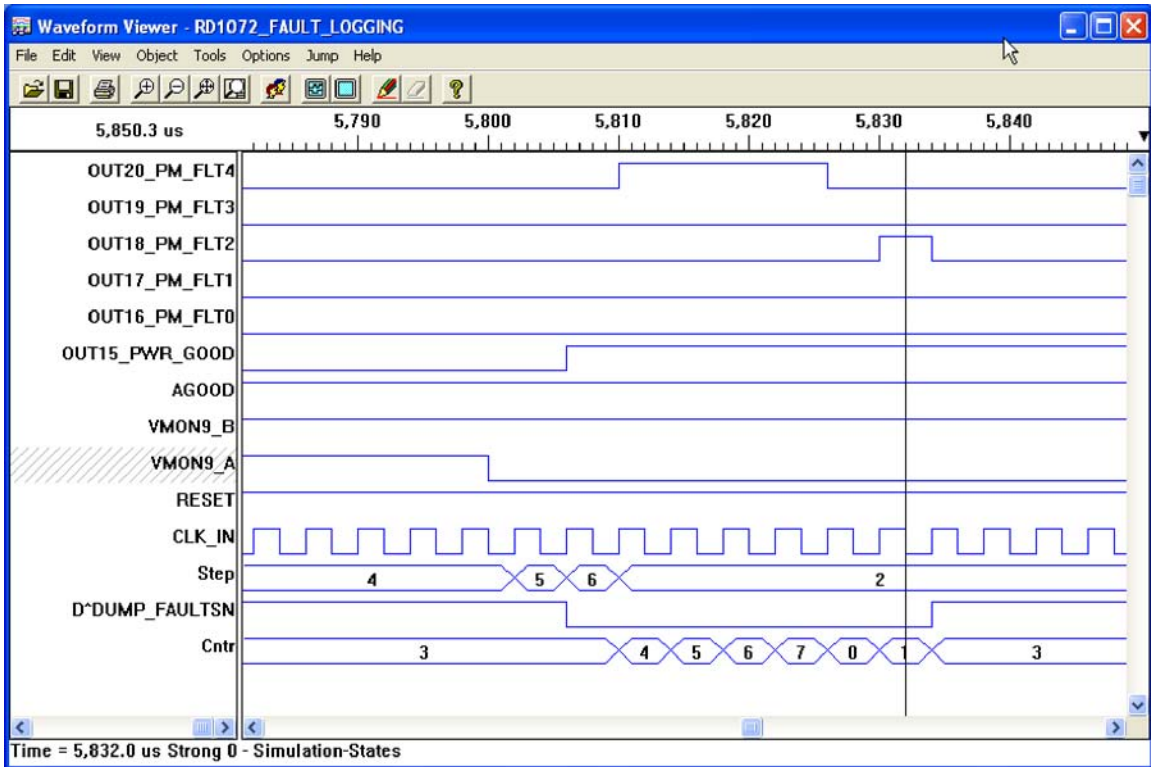


Figure 5. VMON9 Over-Voltage Fault Dump Simulation



## Implementation

### Performance and Resource Utilization

Device	Macrocells	Product Terms	VMONs	In	Out	HVOUT	Timers
ispPAC-POWR1220AT8 <sup>1</sup>	17	80	1 <sup>2</sup>	0	5	0	1

1. Resource utilization characteristics are generated using PAC-Designer 5.2 software. When using this design in a different device, utilization characteristics may vary.
2. All 12 VMONs status are captured in this design but, none are dedicated to this design. All are free to be used at the designer's needs and discretion. VMON9 is used as an example in this design and to be compatible with the Power Manager Hercules Evaluation Board.

## References

- [ispPAC-POWR1220AT8 Data Sheet](#)
- [MachXO Family Data Sheet](#)
- [LatticeXP2 Family Data Sheet](#)

## Technical Support Assistance

Hotline: 1-800-LATTICE (North America)  
+1-503-268-8001 (Outside North America)  
e-mail: [techsupport@latticesemi.com](mailto:techsupport@latticesemi.com)  
Internet: [www.latticesemi.com](http://www.latticesemi.com)

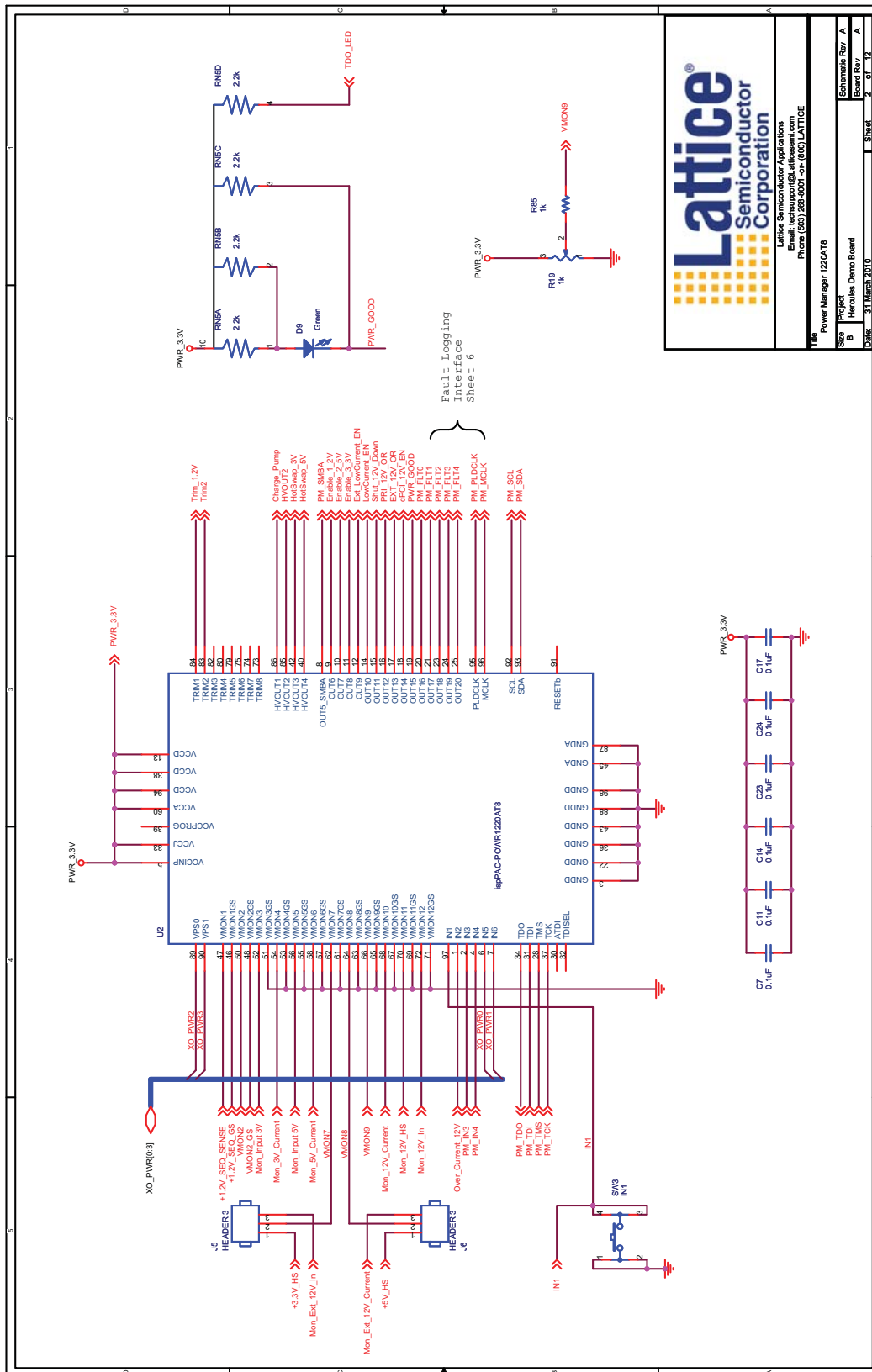
## Revision History

Date	Version	Change Summary
April 2010	01.0	Initial release.



# Appendix A. Schematic

Figure 6. Power Manager ispPAC-POWR1220AT8



**Lattice Semiconductor Corporation**

Lattice Semiconductor Corporation  
 Email: [techsupport@lattice.com](mailto:techsupport@lattice.com)  
 Phone: (503) 288-8001 ext. (800) LATTICE

Power Manager 1220AT8

Project	Heracles Demo Board	Sheet	2 of 12
Size	B	Schematic Rev	A
Board Rev	A		
Date:	31 March 2010		

