Three-Wire Power Supply Fault Logging Using Lattice Programmable Logic

June 2010 Reference Design RD1062

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 non-volatile SPI Flash memory, a Power Manager II, and a Lattice Programmable Logic device (MachXO™, LatticeXP2™, or ispMACH® 4000).

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 three data status lines. This design uses the ispPAC®-POWR1014A Power Manager II device but could also be adapted to the ispPAC-POWR1220AT8 device.

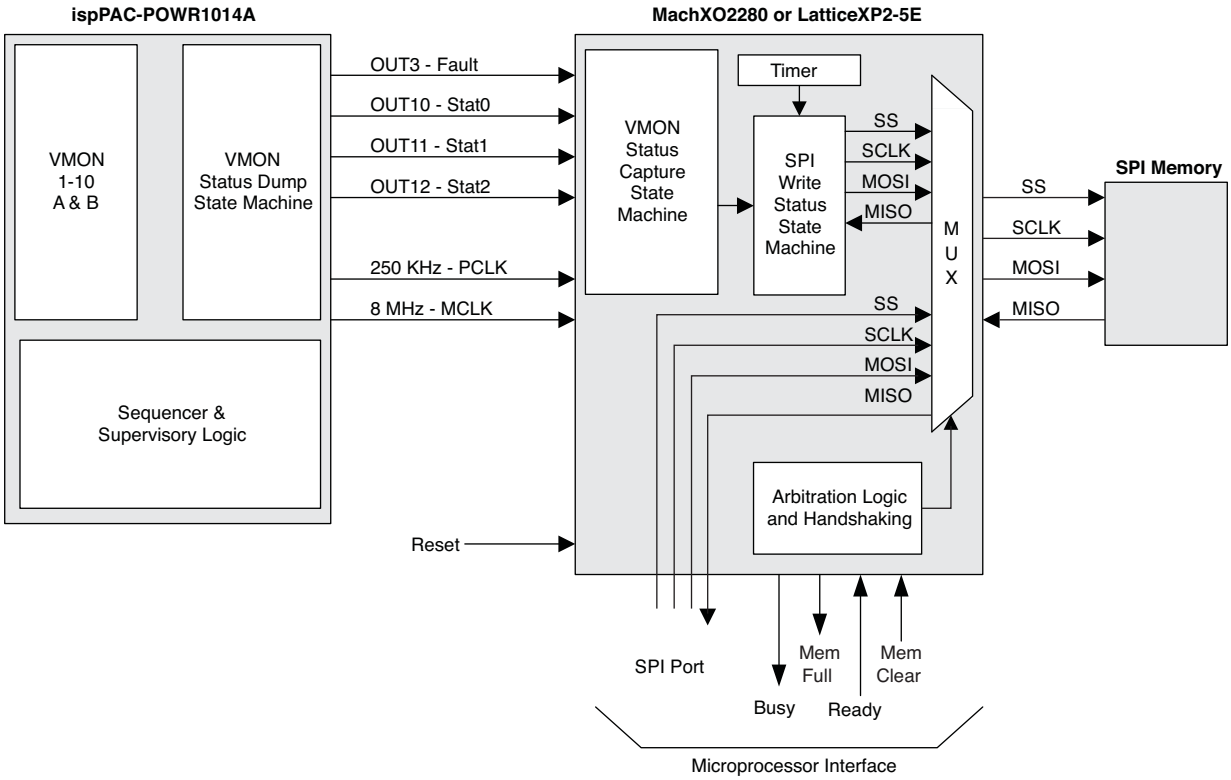
The fault status output and the three data status lines are connected to a Lattice Programmable Logic device which then captures the fault data, formats it, and writes the formatted data to a SPI Flash memory for later retrieval.

The sequence of events can be summarized as follows:

1. The ispPAC-POWR1014A detects a fault on one or more of the VMON inputs and dumps the status of all the VMON inputs to the Lattice Programmable Logic 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 Lattice Programmable Logic device implements a receiver state machine to capture the VMON status.
4. The Lattice Programmable Logic device 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



Signal Descriptions

Table 1 provides the signal descriptions for this reference design.

Table 1. Fault Logger Signal Descriptions

Name	Direction	Active State	Description
reset	Input	Low	Async reset signal
Power Manager Interface (ispPAC-POWR1014A)			
fault	Input	High	Wakeup status capture state machine
stat[2:0]	Input	N/A	VMON status data
clk_1	Input	N/A	8 MHz clock input
clk_2	Input	N/A	250 kHz clock input
SPI Interface (SPI Memory)			
ss	Output	High	SPI chip select
sclk	Output	Selectable	SPI clock
miso	Input	High	Serial data from SPI device
mosi	Output	High	Serial control/address/data into SPI
Microprocessor Interface			
sys_spi_ss	Input	High	SPI chip select from system resource (microcontroller)
sys_spi_sclk	Input	Selectable	SPI clock from system resource (microcontroller)
sys_spi_miso	Output	High	Serial data from SPI to system resource (microcontroller)
sys_spi_mosi	Input	High	Serial control/address/data into SPI (microcontroller)
busy	Output	High	Flag indicates fault capture and logging to SPI in process
mem_full	Output	High	Flag indicates the SPI memory is full

Table 1. Fault Logger Signal Descriptions (Continued)

Name	Direction	Active State	Description
ready	Input	High	Microprocessor asserts when reading or erasing SPI memory
mem_clear	Input	High	Microprocessor asserts to reset Fault Page Pointer when SPI memory is erased

Design Details

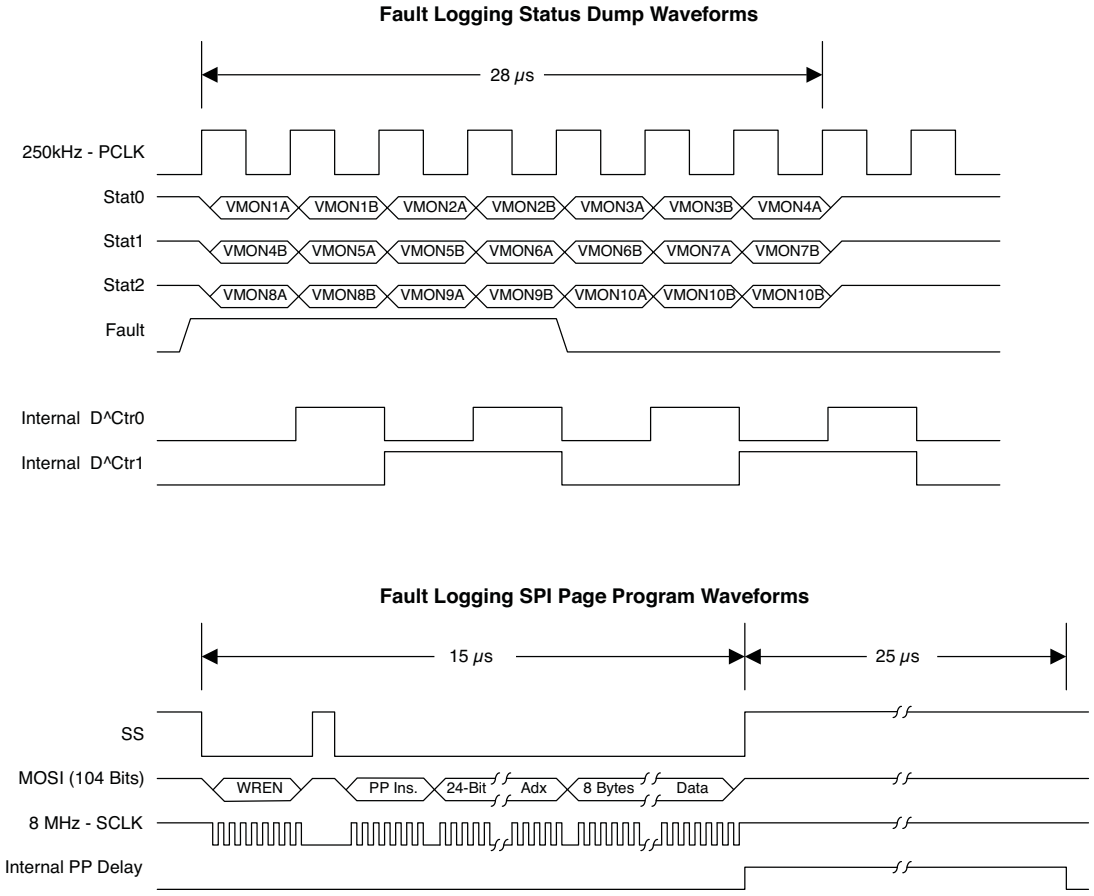
One key aspect of the Fault Logging Reference Design is the SPI-MUX and arbitration logic. This allows the Fault Logging Reference Design to share the SPI Flash Memory with any embedded or stand-alone microprocessor or microcontroller. The arbitration logic prevents the two SPI masters from communicating to the SPI Flash memory at the same time.

Figure 2 shows the details of the VMON status dump from the Lattice Programmable Logic device and the transfer of data from the 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 Lattice Programmable Logic device. The Supervisory Equations feature within the PAC-Designer® software was used to create the VMON status dump from the Power Manager II device. For an example of the Supervisory Equations that will produce the bit order shown in Figure 2, please see the References section of this document.

For convenience, this reference design 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 2.

Figure 2. Waveform Diagrams of Fault Logging Design



To simplify the design and minimize the SPI write time, each page within the SPI Flash is dedicated to a single event log. The first byte of each page is used as a Fault Flag. If the page is erased the Fault Flag will have a value of 0xFF. If the page has any fault data recorded then the Fault Flag will have a value of 0xC3. This allows both the fault logger and an external fault reader (the microprocessor) to know which pages have valid data. For this design eight bytes are used to log a fault and the organization of the bytes within a page is shown in Tables 2 and 3.

Table 2. SPI Page Memory Map Byte Arrangement

Byte Address	Byte Description	Comments
0x00	Fault Flag	0xFF = No fault, page empty: 0xC3 = Fault log, page written.
0x01	VMON1 - VMON4	See Table 2 for details of fault order.
0x02	VMON5 – VMON8	See Table 2 for details of fault order.
0x03	VMON9 – VMON12	See Table 2 for details of fault order.
0x04	Seconds_0	Least significant byte – long integer.
0x05	Seconds_1	
0x06	Seconds_2	
0x07	Seconds_3	Most significant byte – long integer.

Table 3. VMON Fault Memory Map Bit Arrangement

Byte Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x01	VMON4_B	VMON4_A	VMON3_B	VMON3_A	VMON2_B	VMON2_A	VMON1_B	VMON1_A
0x02	VMON8_B	VMON8_A	VMON7_B	VMON7_A	VMON6_B	VMON6_A	VMON5_B	VMON5_A
0x03	VMON12_B	VMON12_A	VMON11_B	VMON11_A	VMON10_B	VMON10_A	VMON9_B	VMON9_A

In this reference design the memory map bytes 0x04 through 0x07 contain a 32-bit binary representation of the timer value in seconds. The timer can count up to a value of 2^{32} -1 seconds which is equivalent to over 136 years. The timer will be reset when the device is powered down or a reset is issued.

Start-up and Reset Conditions

When the design is powered up or a reset is issued, the reference design will begin by reading the SPI Flash memory to determine if any faults are stored in the memory at the current time. The presence of a fault in a page of the SPI Flash is indicated by byte address 0x00 as shown in Table 2. If a fault is present in a page the design will increment the page address counter to the next page address and re-check for a fault in the new page. This will continue until an empty SPI page is found and then the next fault will be logged into this location. This allows any existing faults to remain logged into the SPI Flash memory until the microprocessor issues a SPI Flash memory erase command. At this time the microprocessor will also set the MEM CLR status bit so the reference design will know that a memory clear has occurred. When the MEM CLR status bit is received the reference design will reset the address counter to begin at page 00. Because of this design feature, the faults will always be logged into the SPI Flash memory in sequential order.

Implementation

Table 4. Performance and Resource Utilization

Device	Language	Speed Grade	Utilization (LUTs)	f_{MAX} (MHz)[4]	I/Os[5]	Architecture Resources
MachXO[1]	Verilog	-5	250	>70	19	N/A
	VHDL	-5	251	>70	19	N/A
LatticeXP2[2]	Verilog	-5	300	>70	19	N/A
	VHDL	-5	302	>70	19	N/A
ispMACH 4000[3]	Verilog	-3.5 (ns)	261	>70	19	N/A
	VHDL	-3.5 (ns)	261	>70	19	N/A

1. Performance and utilization characteristics are generated using LCMXO2280C-5FT256C, with Lattice ispLEVER 8.1 software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
2. Performance and utilization characteristics are generated using LFXP2-5E-5FT256C, with Lattice ispLEVER 8.1 software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
3. Performance and utilization characteristics are generated using LC4512V-35T176C, with Lattice ispLEVER Classic 1.3 software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
4. This reference design uses the 8 MHz MCLK signal (labeled clk_1 in the design) from the Power Manager II device to drive the SPI memory write operations. The MCLK signal is not adjustable. A user could connect a faster clock signal if desired. In this case, the design will support a 70 MHz or faster clock for this purpose.
5. The number of I/Os is for the Fault Logging Reference Design with a SPI port interface which is available for a microprocessor or microcontroller. The external port for the microprocessor uses four I/Os for the SPI port plus another four I/Os for the arbitration logic. The external SPI port for the memory uses only four I/Os.

References

- ispPAC-POWR1014/A Data Sheet
- MachXO Family Data Sheet
- LatticeXP2 Family Data Sheet
- ispMACH 4000V/B/C/Z Family Data Sheet
- Supervisory Equations.txt file – See the Docs directory in this Reference Design zip file.

Technical Support Assistance

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

Revision History

Date	Version	Change Summary
October 2009	01.0	Initial release.
February 2010	01.1	Added VHDL support.
		Added support for LatticeXP2 device family.
June 2010	01.2	Changed document title from “Power Supply Fault Logging” to “Three-Wire Power Supply Fault Logging Using Lattice Programmable Logic.”
		Added support for ispMACH 4000 device family.