

## FEATURES

- Complete supervisory and sequencing solution for up to 8 supplies
- 16 event deep black box nonvolatile fault recording
- 8 supply fault detectors enable supervision of supplies to
  - <0.5% accuracy at all voltages at 25°C
  - <1.0% accuracy across all voltages and temperatures
- 4 selectable input attenuators allow supervision of supplies to
  - 14.4 V on VH
  - 6 V on VP1 to VP3 (VPx)
- 4 dual-function inputs, VX1 to VX4 (VXx)
  - High impedance input to supply fault detector with thresholds between 0.573 V and 1.375 V
  - General-purpose logic input
- 8 programmable driver outputs, PDO1 to PDO8 (PDOx)
  - Open-collector with external pull-up
  - Push/pull output, driven to VDDCAP or VPx
  - Open-collector with weak pull-up to VDDCAP or VPx
  - Internally charge-pumped high drive for use with external NFET (PDO1 to PDO6 only)
- Sequencing engine (SE) implements state machine control of PDO outputs
  - State changes conditional on input events
  - Enables complex control of boards
  - Power-up and power-down sequence control
  - Fault event handling
  - Interrupt generation on warnings
  - Watchdog function can be integrated in SE
  - Program software control of sequencing through SMBus
- Device powered by the highest of VPx, VH for improved redundancy
- User EEPROM: 256 bytes
- Industry-standard 2-wire bus interface (SMBus)
- Guaranteed PDO low with VH, VPx = 1.2 V
- Available in 32-lead, 7 mm × 7 mm LQFP

## APPLICATIONS

- Central office systems
- Servers/routers
- Multivoltage system line cards
- DSP/FPGA supply sequencing
- In-circuit testing of margined supplies

## FUNCTIONAL BLOCK DIAGRAM

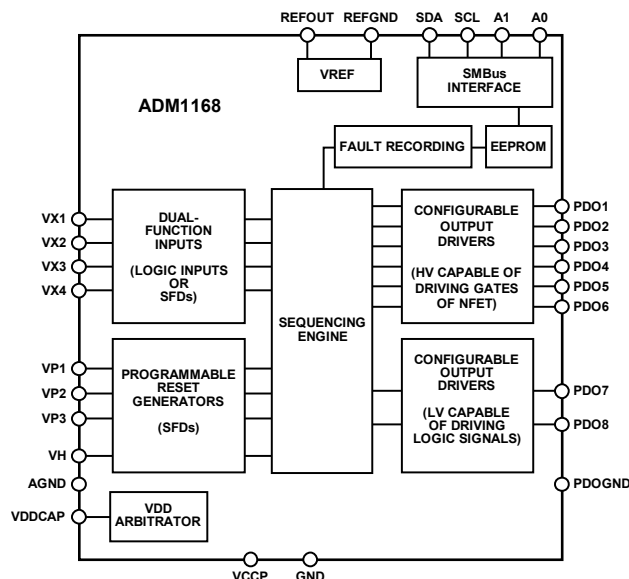


Figure 1.

## GENERAL DESCRIPTION

The [ADM1168](#) Super Sequencer® is a configurable supervisory/sequencing device that offers a single-chip solution for supply monitoring and sequencing in multiple supply systems.

The device also provides up to eight programmable inputs for monitoring undervoltage faults, overvoltage faults, or out-of-window faults on up to eight supplies. In addition, eight programmable outputs can be used as logic enables. Six of these programmable outputs can also provide up to a 12 V output for driving the gate of an NFET that can be placed in the path of a supply.

The logical core of the device is a sequencing engine. This state machine-based construction provides up to 63 different states. This design enables very flexible sequencing of the outputs based on the condition of the inputs.

A block of nonvolatile EEPROM is available that can be used to store user-defined information and may also be used to hold a number of fault records that are written by the sequencing engine defined by the user when a particular fault or sequence occurs.

The [ADM1168](#) is controlled via configuration data that can be programmed into an EEPROM. The whole configuration can be programmed using an intuitive GUI-based software package provided by Analog Devices, Inc.

For more information about the [ADM1168](#) register map, refer to the [AN-721 Application Note](#).

Rev. B

[Document Feedback](#)

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.  
Tel: 781.329.4700 ©2011–2015 Analog Devices, Inc. All rights reserved.  
Technical Support [www.analog.com](http://www.analog.com)

## TABLE OF CONTENTS

Features .....	1	Default Output Configuration.....	14
Applications.....	1	Sequencing Engine.....	15
Functional Block Diagram .....	1	Overview .....	15
General Description .....	1	Warnings.....	15
Revision History .....	2	SMBus Jump (Unconditional Jump).....	15
Detailed Block Diagram .....	3	Sequencing Engine Application Example .....	16
Specifications.....	4	Fault and Status Reporting.....	17
Absolute Maximum Ratings.....	6	Nonvolatile Black Box Fault Recording.....	17
Thermal Resistance .....	6	Black Box Writes with No External Supply .....	18
ESD Caution.....	6	Applications Diagram .....	19
Pin Configuration and Function Descriptions.....	7	Communicating with the ADM1168.....	20
Typical Performance Characteristics .....	8	Configuration Download at Power-Up.....	20
Powering the ADM1168 .....	10	Updating the Configuration .....	20
Slew Rate Consideration.....	10	Updating the Sequencing Engine.....	21
Inputs.....	11	Internal Registers.....	21
Supply Supervision.....	11	EEPROM .....	21
Programming the Supply Fault Detectors.....	11	Serial Bus Interface.....	22
Input Comparator Hysteresis.....	12	SMBus Protocols for RAM and EEPROM.....	24
Input Glitch Filtering .....	12	Write Operations .....	24
Supply Supervision with VXx Inputs.....	13	Read Operations.....	25
VXx Pins as Digital Inputs .....	13	Outline Dimensions .....	27
Outputs .....	14	Ordering Guide .....	27
Supply Sequencing Through Configurable Output Drivers.....	14		

## REVISION HISTORY

### 1/15—Rev. A to Rev. B

Added Slew Rate Consideration Section ..... 10

### 8/13—Rev. 0 to Rev. A

Changed REVID Value from 0x12 to 0x10 in Table 10..... 22

### 4/11—Revision 0: Initial Version

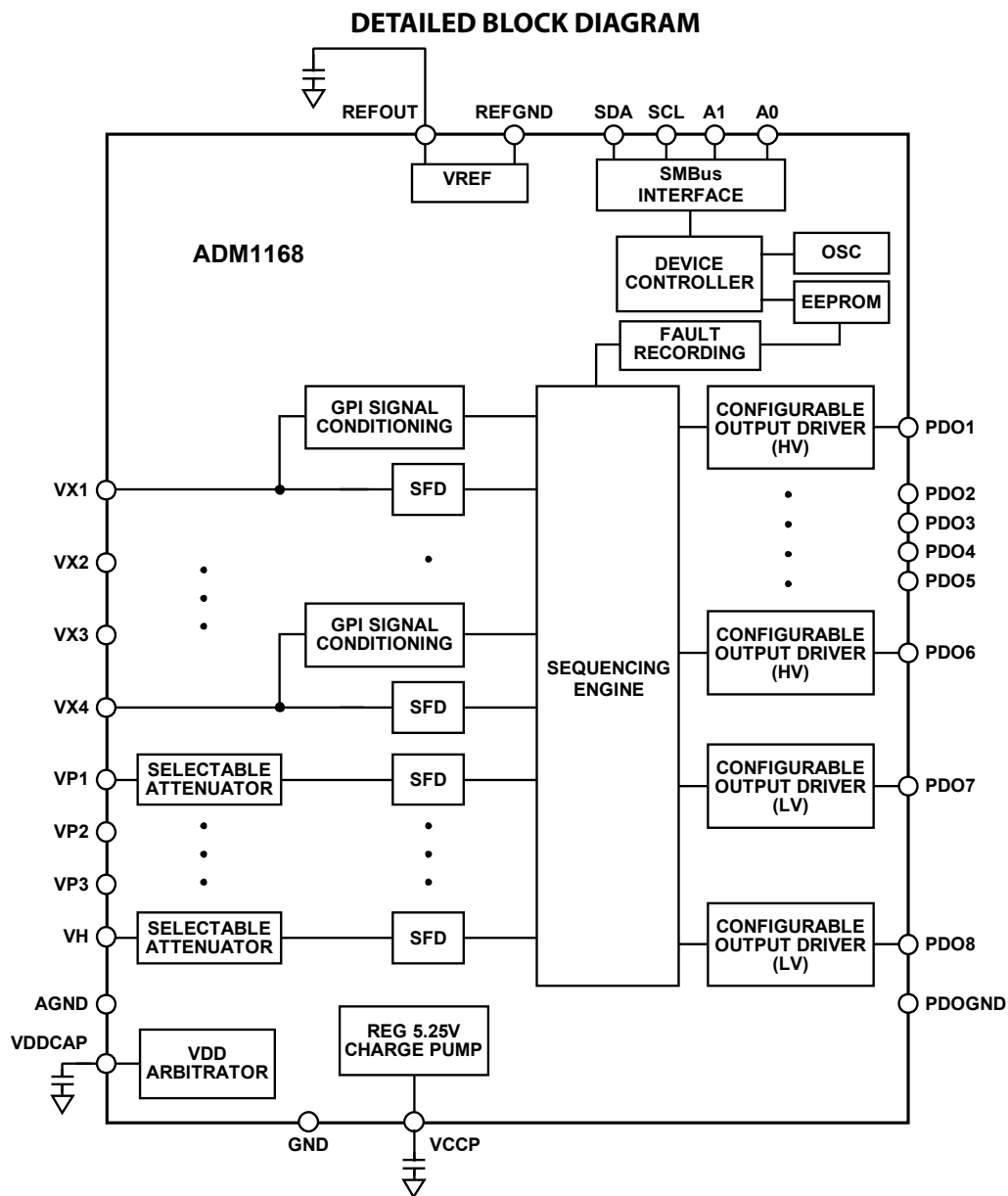


Figure 2.

09474-002

## SPECIFICATIONS

V<sub>H</sub> = 3.0 V to 14.4 V<sup>1</sup>, V<sub>Px</sub> = 3.0 V to 6.0 V<sup>1</sup>, T<sub>A</sub> = -40°C to +85°C, unless otherwise noted.

Table 1.

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
POWER SUPPLY ARBITRATION					
VH, VPx	3.0			V	Minimum supply required on one of the VH, VPx pins
VPx			6.0	V	Maximum VDDCAP = 5.1 V typical
VH			14.4	V	VDDCAP = 4.75 V
VDDCAP	2.7	4.75	5.4	V	Regulated LDO output
C <sub>VDDCAP</sub>	10			μF	Minimum recommended decoupling capacitance
POWER SUPPLY					
Supply Current, I <sub>VH</sub> , I <sub>VPx</sub>		4.2	6	mA	VDDCAP = 4.75 V, PDO1 to PDO8 off
Additional Currents					
All PDOx FET Drivers On		1		mA	VDDCAP = 4.75 V, PDO1 to PDO6 loaded with 1 μA each, PDO7 to PDO8 off
Current Available from VDDCAP			2	mA	Maximum additional load that can be drawn from all PDO pull-ups to VDDCAP
EEPROM Erase Current		10		mA	1 ms duration only, VDDCAP = 3 V
SUPPLY FAULT DETECTORS					
VH Pin					Midrange and high range
Input Impedance		52		kΩ	
Input Attenuator Error		±0.05		%	
Detection Ranges					
High Range	6		14.4	V	
Midrange	2.5		6	V	Low range and midrange
VPx Pins					
Input Impedance		52		kΩ	
Input Attenuator Error		±0.05		%	
Detection Ranges					
Midrange	2.5		6	V	No input attenuation error
Low Range	1.25		3	V	
Ultralow Range	0.573		1.375	V	
VXx Pins					No input attenuation error
Input Impedance	1			MΩ	
Detection Ranges					
Ultralow Range	0.573		1.375	V	
Absolute Accuracy			±1	%	
Threshold Resolution		8		Bits	Internal reference VREF error + DAC nonlinearity + comparator offset error
Digital Glitch Filter		0		μs	
		100		μs	Maximum programmable filter length
REFERENCE OUTPUT					
Reference Output Voltage	2.043	2.048	2.053	V	No load
Load Regulation		−0.25		mV	Sourcing current
		0.25		mV	Sinking current
Minimum Load Capacitance	1			μF	Capacitor required for decoupling, stability
PSRR		60		dB	DC

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
<b>PROGRAMMABLE DRIVER OUTPUTS</b>					
High Voltage (Charge Pump) Mode (PDO1 to PDO6)					
Output Impedance		500		k $\Omega$	
$V_{OH}$	11	12.5	14	V	$I_{OH} = 0 \mu A$
	10.5	12	13.5	V	$I_{OH} = 1 \mu A$
$I_{OUTAVG}$		20		$\mu A$	$2 V < V_{OH} < 7 V$
Standard (Digital Output) Mode (PDO1 to PDO8)					
$V_{OH}$	2.4			V	$V_{PU}$ (pull-up to VDDCAP or VPx) = 2.7 V, $I_{OH} = 0.5 mA$
			4.5	V	$V_{PU}$ to VPx = 6.0 V, $I_{OH} = 0 mA$
	$V_{PU} - 0.3$			V	$V_{PU} \leq 2.7 V$ , $I_{OH} = 0.5 mA$
$V_{OL}$	0		0.50	V	$I_{OL} = 20 mA$
$I_{OL}^2$			20	mA	Maximum sink current per PDO pin
$I_{SINK}^2$			60	mA	Maximum total sink for all PDO pins
$R_{PULL-UP}$	16	20	29	k $\Omega$	Internal pull-up
$I_{SOURCE} (VPx)^2$			2	mA	Current load on any VPx pull-ups, that is, total source current available through any number of PDO pull-up switches configured onto any one VPx pin
Three-State Output Leakage Current			10	$\mu A$	$V_{PDO} = 14.4 V$
Oscillator Frequency	90	100	110	kHz	All on-chip time delays derived from this clock
<b>DIGITAL INPUTS (VXx, A0, A1)</b>					
Input High Voltage, $V_{IH}$	2.0			V	Maximum $V_{IN} = 5.5 V$
Input Low Voltage, $V_{IL}$			0.8	V	Maximum $V_{IN} = 5.5 V$
Input High Current, $I_{IH}$	-1			$\mu A$	$V_{IN} = 5.5 V$
Input Low Current, $I_{IL}$			1	$\mu A$	$V_{IN} = 0 V$
Input Capacitance		5		pF	
Programmable Pull-Down Current, $I_{PULL-DOWN}$		20		$\mu A$	$VDDCAP = 4.75 V$ , $T_A = 25^\circ C$ , if known logic state is required
<b>SERIAL BUS DIGITAL INPUTS (SDA, SCL)</b>					
Input High Voltage, $V_{IH}$	2.0			V	
Input Low Voltage, $V_{IL}$			0.8	V	
Output Low Voltage, $V_{OL}^2$			0.4	V	$I_{OUT} = -3.0 mA$
<b>SERIAL BUS TIMING</b>					
Clock Frequency, $f_{SCLK}$			400	kHz	See Figure 27
Bus Free Time, $t_{BUF}$	1.3			$\mu s$	
Start Setup Time, $t_{SU,STA}$	0.6			$\mu s$	
Stop Setup Time, $t_{SU,STO}$	0.6			$\mu s$	
Start Hold Time, $t_{HD,STA}$	0.6			$\mu s$	
SCL Low Time, $t_{LOW}$	1.3			$\mu s$	
SCL High Time, $t_{HIGH}$	0.6			$\mu s$	
SCL, SDA Rise Time, $t_R$			300	ns	
SCL, SDA Fall Time, $t_F$			300	ns	
Data Setup Time, $t_{SU,DAT}$	100			ns	
Data Hold Time, $t_{HD,DAT}$	250			ns	
Input Low Current, $I_{IL}$			1	$\mu A$	$V_{IN} = 0 V$
<b>SEQUENCING ENGINE TIMING</b>					
State Change Time		10		$\mu s$	

<sup>1</sup> At least one of the VH, VPx pins must be  $\geq 3.0 V$  to maintain the device supply on VDDCAP.

<sup>2</sup> Specification is not production tested but is supported by characterization data at initial product release.

## ABSOLUTE MAXIMUM RATINGS

Table 2.

Parameter	Rating
Voltage on VH Pin	16 V
Voltage on VPx Pins	7 V
Voltage on VXx Pins	−0.3 V to +6.5 V
Voltage on A0, A1 Pins	−0.3 V to +7 V
Voltage on REFOUT Pin	5 V
Voltage on VDDCAP, VCCP Pins	6.5 V
Voltage on PDOx Pins	16 V
Voltage on SDA, SCL Pins	7 V
Voltage on GND, AGND, PDOGND, REFGND Pins	−0.3 V to +0.3 V
Input Current at Any Pin	±5 mA
Package Input Current	±20 mA
Maximum Junction Temperature (T <sub>j</sub> max)	150°C
Storage Temperature Range	−65°C to +150°C
Lead Temperature	
Soldering Vapor Phase, 60 sec	215°C
ESD Rating, All Pins	2000 V

Stresses at or above those listed under Absolute Maximum Ratings may cause permanent damage to the product. This is a stress rating only; functional operation of the product at these or any other conditions above those indicated in the operational section of this specification is not implied. Operation beyond the maximum operating conditions for extended periods may affect product reliability.

## THERMAL RESISTANCE

$\theta_{JA}$  is specified for the worst-case conditions, that is, a device soldered in a circuit board for surface-mount packages.

Table 3. Thermal Resistance

Package Type	$\theta_{JA}$	Unit
32-Lead LQFP	54	°C/W

## ESD CAUTION



**ESD (electrostatic discharge) sensitive device.** Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

## PIN CONFIGURATION AND FUNCTION DESCRIPTIONS

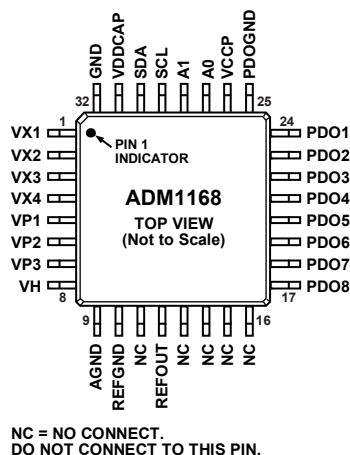


Figure 3. Pin Configuration

Table 4. Pin Function Descriptions

Pin No.	Mnemonic	Description
1 to 4	VX1 to VX4	High Impedance Inputs to Supply Fault Detectors. Fault thresholds can be set from 0.573 V to 1.375 V. Alternatively, these pins can be used as general-purpose digital inputs.
5 to 7	VP1 to VP3	Low Voltage Inputs to Supply Fault Detectors. Three input ranges can be set by altering the input attenuation on a potential divider connected to these pins, the output of which connects to a supply fault detector. These pins allow thresholds from 2.5 V to 6 V, from 1.25 V to 3 V, and from 0.573 V to 1.375 V.
8	VH	High Voltage Input to Supply Fault Detectors. Three input ranges can be set by altering the input attenuation on a potential divider connected to this pin, the output of which connects to a supply fault detector. This pin allows thresholds from 6 V to 14.4 V and from 2.5 V to 6 V.
9	AGND <sup>1</sup>	Ground Return for Input Attenuators.
10	REFGND <sup>1</sup>	Ground Return for On-Chip Reference Circuits.
11, 13 to 16	NC	No Connect. Do not connect to this pin.
12	REFOUT	Reference Output, 2.048 V. Note that a capacitor must always be connected between this pin and REFGND. A 10 $\mu$ F capacitor is recommended for this purpose.
17 to 24	PDO8 to PDO1	Programmable Output Drivers.
25	PDOGND <sup>1</sup>	Ground Return for Output Drivers.
26	VCCP	Central Charge-Pump Voltage of 5.25 V. A reservoir capacitor must be connected between this pin and GND. A 10 $\mu$ F capacitor is recommended for this purpose.
27	A0	Logic Input. This pin sets the seventh bit of the SMBus interface address.
28	A1	Logic Input. This pin sets the sixth bit of the SMBus interface address.
29	SCL	SMBus Clock Pin. Bidirectional open drain requires external resistive pull-up.
30	SDA	SMBus Data Pin. Bidirectional open drain requires external resistive pull-up.
31	VDDCAP	Device Supply Voltage. Linearly regulated from the highest of the VPx and VH pins to a typical of 4.75 V. A capacitor must be connected between this pin and GND. A 10 $\mu$ F capacitor is recommended for this purpose.
32	GND <sup>1</sup>	Supply Ground.

<sup>1</sup> In a typical application, all ground pins are connected together.

# TYPICAL PERFORMANCE CHARACTERISTICS

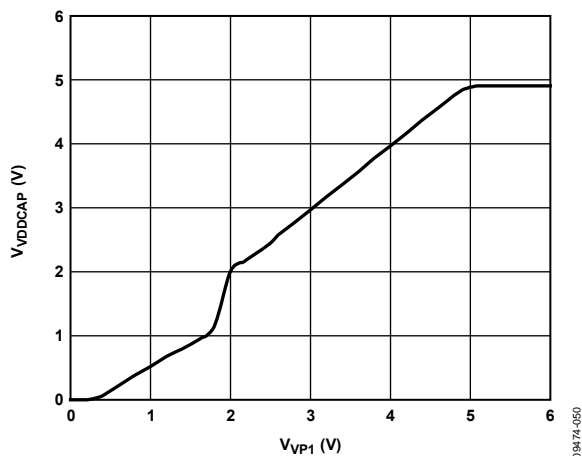


Figure 4.  $V_{DDCAP}$  vs.  $V_{VP1}$

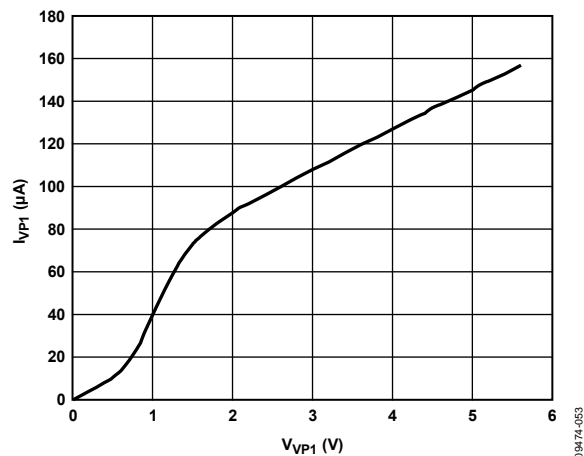


Figure 7.  $I_{VP1}$  vs.  $V_{VP1}$  (VP1 Not as Supply)

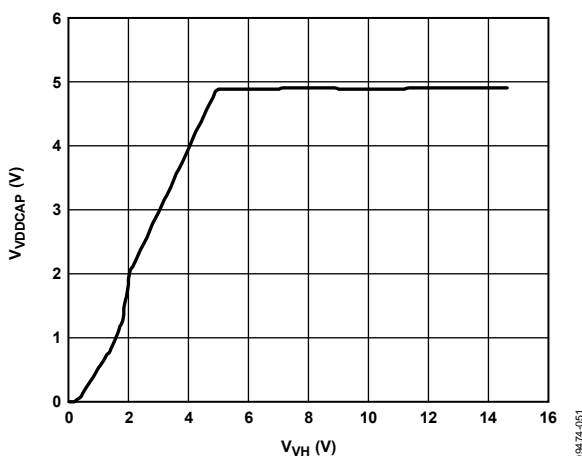


Figure 5.  $V_{DDCAP}$  vs.  $V_{VH}$

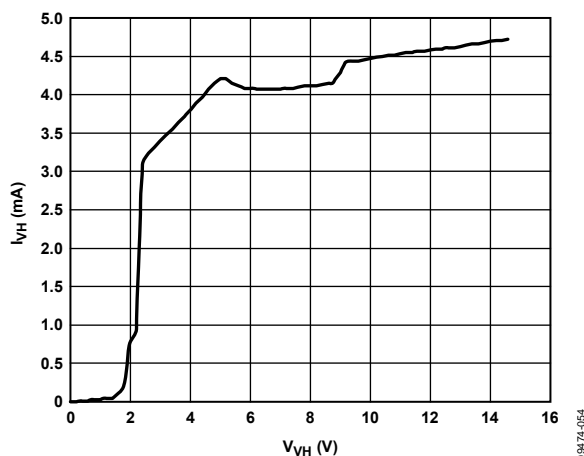


Figure 8.  $I_{VH}$  vs.  $V_{VH}$  (VH as Supply)

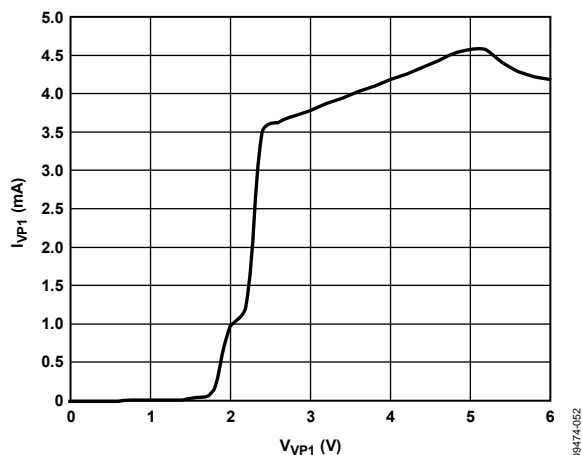


Figure 6.  $I_{VP1}$  vs.  $V_{VP1}$  (VP1 as Supply)

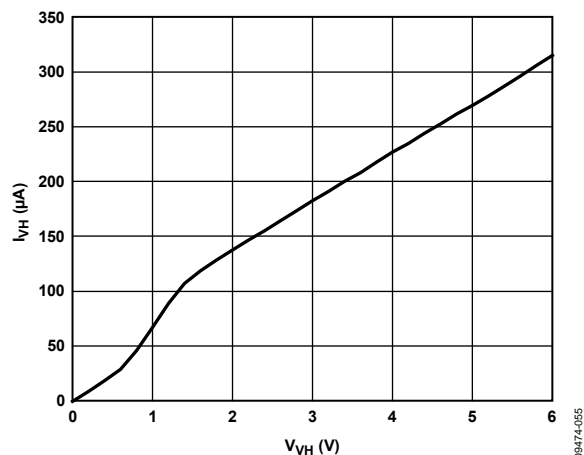
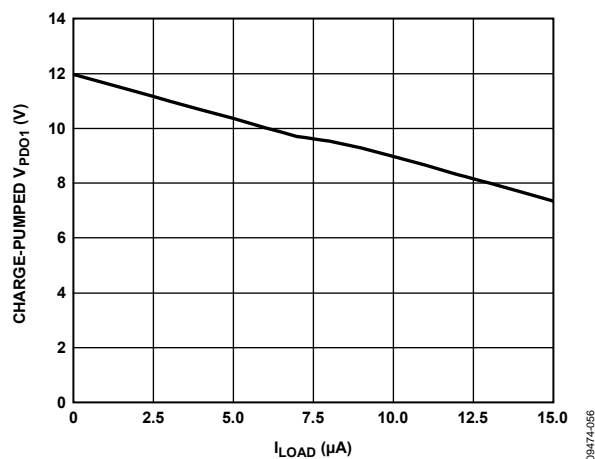
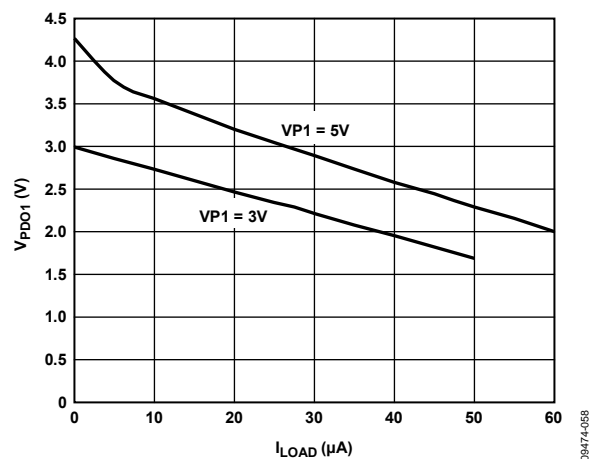
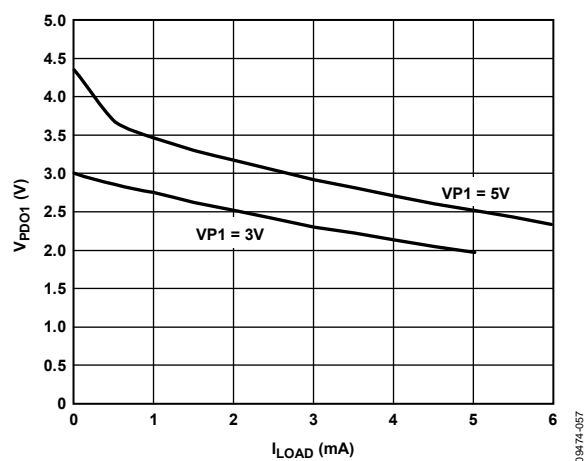
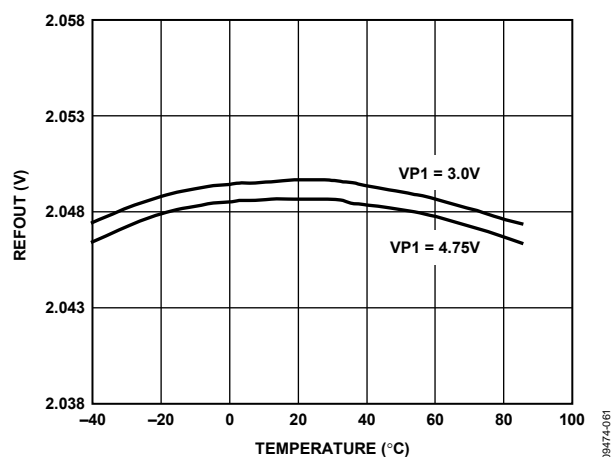


Figure 9.  $I_{VH}$  vs.  $V_{VH}$  (VH Not as Supply)



Figure 10. Charge-Pumped  $V_{PDO1}$  (FET Drive Mode) vs.  $I_{LOAD}$ Figure 12.  $V_{PDO1}$  (Weak Pull-Up to  $VPx$ ) vs.  $I_{LOAD}$ Figure 11.  $V_{PDO1}$  (Strong Pull-Up to  $VPx$ ) vs.  $I_{LOAD}$ Figure 13.  $REFOUT$  vs. Temperature

## POWERING THE ADM1168

The ADM1168 is powered from the highest voltage input on either the positive-only supply inputs (VPx) or the high voltage supply input (VH). This technique offers improved redundancy because the device is not dependent on any particular voltage rail to keep it operational. The same pins are used for supply fault detection (see the Supply Supervision section). A  $V_{DD}$  arbitrator on the device chooses which supply to use. The arbitrator can be considered an OR'ing of four low dropout regulators (LDOs) together. A supply comparator chooses the highest input to provide the on-chip supply. There is minimal switching loss with this architecture ( $\sim 0.2$  V), resulting in the ability to power the ADM1168 from a supply as low as 3.0 V. Note that the supply on the Vx pins cannot be used to power the device.

An external capacitor to GND is required to decouple the on-chip supply from noise. This capacitor should be connected to the VDDCAP pin, as shown in Figure 14. The capacitor has another use during brownouts (momentary loss of power). Under these conditions, when the input supply (VPx or VH) dips transiently below  $V_{DD}$ , the synchronous rectifier switch immediately turns off so that it does not pull  $V_{DD}$  down. The  $V_{DD}$  capacitor can then act as a reservoir to keep the device active until the next highest supply takes over the powering of the device. A 10  $\mu$ F capacitor is recommended for this reservoir/decoupling function.

The value of the VDDCAP capacitor may be increased if it is necessary to guarantee a complete fault record is written into EEPROM should all supplies fail. The value of the capacitor to use is discussed in the Black Box Writes with No External Supply section.

The VH input pin can accommodate supplies up to 14.4 V, which allows the ADM1168 to be powered using a 12 V backplane supply. In cases where this 12 V supply is hot swapped, it is recommended that the ADM1168 not be connected directly to the supply. Suitable precautions, such as the use of a hot swap controller or RC filter network, should be taken to protect the device from transients that may cause damage during hot swap events.

When two or more supplies are within 100 mV of each other, the supply that first takes control of  $V_{DD}$  keeps control. For example, if VP1 is connected to a 3.3 V supply,  $V_{DD}$  powers up to approximately 3.1 V through VP1. If VP2 is then connected to another 3.3 V supply, VP1 still powers the device, unless VP2 goes 100 mV higher than VP1.

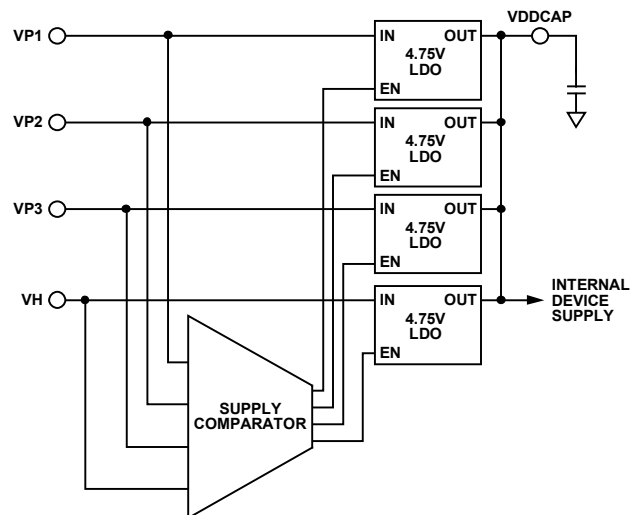


Figure 14.  $V_{DD}$  Arbitrator Operation

## SLEW RATE CONSIDERATION

When the ambient temperature of operation is less than approximately  $-20^{\circ}\text{C}$ , and in the event of a power loss where all supply inputs fail for less than a few hundreds of milliseconds (for example, due to a system supply brownout), it is recommended that the supply voltage recover with a ramp rate of at least 1.5 V/ms or less than 0.5 V/ms.

## INPUTS

### SUPPLY SUPERVISION

The [ADM1168](#) has eight programmable inputs. Four of these are dedicated supply fault detectors (SFDs). These dedicated inputs are called VH and VPx (VP1 to VP3) by default. The other four inputs are labeled VXx (VX1 to VX4) and have dual functionality. They can be used either as SFDs, with functionality similar to the VH and VPx, or as CMOS-/TTL-compatible logic inputs to the device. Therefore, the [ADM1168](#) can have up to eight analog inputs, a minimum of four analog inputs and four digital inputs, or a combination thereof. If an input is used as an analog input, it cannot be used as a digital input. Therefore, a configuration requiring eight analog inputs has no available digital inputs. Table 6 shows the details of each input.

### PROGRAMMING THE SUPPLY FAULT DETECTORS

The [ADM1168](#) can have up to eight SFDs on its eight input channels. These highly programmable reset generators enable the supervision of up to eight supply voltages. The supplies can be as low as 0.573 V and as high as 14.4 V. The inputs can be configured to detect an undervoltage fault (the input voltage drops below a preprogrammed value), an overvoltage fault (the input voltage rises above a preprogrammed value), or an out-of-window fault (the input voltage is outside a preprogrammed range). The thresholds can be programmed to an 8-bit resolution in registers provided in the [ADM1168](#). This translates to a voltage resolution that is dependent on the range selected.

The resolution is given by

$$\text{Step Size} = \text{Threshold Range}/255$$

Therefore, if the high range is selected on VH, the step size can be calculated as follows:

$$(14.4 \text{ V} - 6.0 \text{ V})/255 = 32.9 \text{ mV}$$

Table 5 lists the upper and lower limits of each available range, the bottom of each range ( $V_B$ ), and the range itself ( $V_R$ ).

**Table 5. Voltage Range Limits**

Voltage Range (V)	$V_B$ (V)	$V_R$ (V)
0.573 to 1.375	0.573	0.802
1.25 to 3.00	1.25	1.75
2.5 to 6.0	2.5	3.5
6.0 to 14.4	6.0	8.4

The threshold value required is given by

$$V_T = (V_R \times N)/255 + V_B$$

where:

$V_T$  is the desired threshold voltage (undervoltage or overvoltage).

$V_R$  is the voltage range.

$N$  is the decimal value of the 8-bit code.

$V_B$  is the bottom of the range.

Reversing the equation, the code for a desired threshold is given by

$$N = 255 \times (V_T - V_B)/V_R$$

For example, if the user wants to set a 5 V overvoltage threshold on VP1, the code to be programmed in the PS1OVTH register (as described in the [AN-721 Application Note](#)) is given by

$$N = 255 \times (5 - 2.5)/3.5$$

Therefore,  $N = 182$  (1011 0110 or 0xB6).

## INPUT COMPARATOR HYSTERESIS

The UV and OV comparators shown in Figure 16 are always monitoring VPx. To avoid chatter (multiple transitions when the input is very close to the set threshold level), these comparators have digitally programmable hysteresis. The hysteresis can be programmed up to the values shown in Table 6.

The hysteresis is added after a supply voltage goes out of tolerance. Therefore, the user can program the amount above the undervoltage threshold to which the input must rise before an undervoltage fault is deasserted. Similarly, the user can program the amount below the overvoltage threshold to which an input must fall before an overvoltage fault is deasserted.

The hysteresis value is given by

$$V_{HYST} = V_R \times N_{THRESH}/255$$

where:

$V_{HYST}$  is the desired hysteresis voltage.

$N_{THRESH}$  is the decimal value of the 5-bit hysteresis code.

Note that  $N_{THRESH}$  has a maximum value of 31. The maximum hysteresis for the ranges is listed in Table 6.

## INPUT GLITCH FILTERING

The final stage of the SFDs is a glitch filter. This block provides time-domain filtering on the output of the SFD comparators, which allows the user to remove any spurious transitions such as supply bounce at turn-on. The glitch filter function is in addition to the digitally programmable hysteresis of the SFD comparators. The glitch filter timeout is programmable up to 100  $\mu$ s.

For example, when the glitch filter timeout is 100  $\mu$ s, any pulse appearing on the input of the glitch filter block that is less than 100  $\mu$ s in duration is prevented from appearing on the output of the glitch filter block. Any input pulse that is longer than 100  $\mu$ s appears on the output of the glitch filter block. The output is delayed with respect to the input by 100  $\mu$ s. The filtering process is shown in Figure 15.

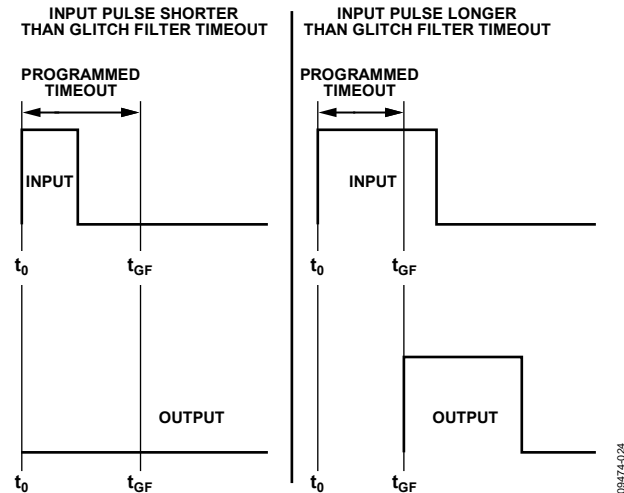


Figure 15. Input Glitch Filter Function

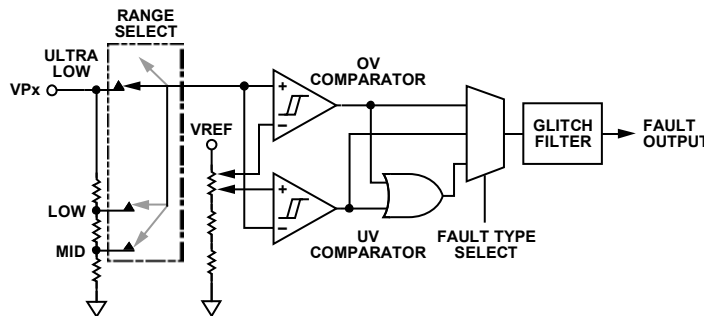


Figure 16. Supply Fault Detector Block

Table 6. Input Functions, Thresholds, and Ranges

Input	Function	Voltage Range (V)	Maximum Hysteresis	Voltage Resolution (mV)	Glitch Filter ( $\mu$ s)
VH	High voltage analog input	2.5 to 6.0	425 mV	13.7	0 to 100
		6.0 to 14.4	1.02 V	32.9	0 to 100
VPx	Positive analog input	0.573 to 1.375	97.5 mV	3.14	0 to 100
		1.25 to 3.00	212 mV	6.8	0 to 100
		2.5 to 6.0	425 mV	13.7	0 to 100
VXx	High-Z analog input	0.573 to 1.375	97.5 mV	3.14	0 to 100
	Digital input	0 to 5.0	Not applicable	Not applicable	0 to 100

## SUPPLY SUPERVISION WITH VXx INPUTS

The VXx inputs have two functions. They can be used as either supply fault detectors or as digital logic inputs. When selected as analog (SFD) inputs, the VXx pins have functionality that is very similar to the VH and VPx pins. The primary difference is that the VXx pins have only one input range: 0.573 V to 1.375 V. Therefore, these inputs can directly supervise only the very low supplies. However, the input impedance of the VXx pins is high, allowing an external resistor divide network to be connected to the pin. Thus, potentially any supply can be divided down into the input range of the VXx pin and supervised. This enables the ADM1168 to monitor other supplies, such as +24 V, +48 V, and -5 V.

An additional supply supervision function is available when the VXx pins are selected as digital inputs. In this case, the analog function is available as a second detector on each of the dedicated analog inputs, VPx and VH. The analog function of VX1 is mapped to VP1, VX2 is mapped to VP2, and so on. VX4 is mapped to VH. In this case, these SFDs can be viewed as secondary or warning SFDs.

The secondary SFDs are fixed to the same input range as the primary SFDs. They are used to indicate warning levels rather than failure levels. This allows faults and warnings to be generated on a single supply using only one pin. For example, if VP1 is set to output a fault when a 3.3 V supply drops to 3.0 V, VX1 can be set to output a warning at 3.1 V. Warning outputs are available for readback from the status registers. They are also ORed together and fed into the SE, allowing warnings to generate interrupts on the PDOs. Therefore, in this example, if the supply drops to 3.1 V, a warning is generated, and remedial action can be taken before the supply drops out of tolerance.

## VXx PINS AS DIGITAL INPUTS

As described in the Supply Supervision with VXx Inputs section, the VXx input pins on the ADM1168 have dual functionality. The second function is as a digital logic input to the device. Therefore, the ADM1168 can be configured for up to four digital inputs. These inputs are TTL-/CMOS-compatible inputs. Standard logic signals can be applied to the pins: RESET from reset generators, PWRGD signals, fault flags, manual resets, and more. These signals are available as inputs to the SE and, therefore, can be used to control the status of the PDOs. The inputs can be configured to detect either a change in level or an edge.

When configured for level detection, the output of the digital block is a buffered version of the input. When configured for edge detection, a pulse of programmable width is output from the digital block, when the logic transition is detected. The width is programmable from 0  $\mu$ s to 100  $\mu$ s.

The digital blocks feature the same glitch filter function that is available on the SFDs. This function enables the user to ignore spurious transitions on the inputs. For example, the filter can be used to debounce a manual reset switch.

When configured as digital inputs, each VXx pin has a weak (10  $\mu$ A) pull-down current source available for placing the input into a known condition, even if left floating. The current source, if selected, weakly pulls the input to GND.

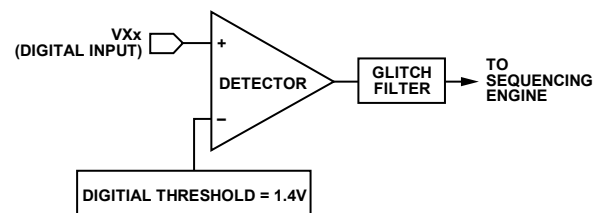


Figure 17. VXx Digital Input Function

09474-027

## OUTPUTS

### SUPPLY SEQUENCING THROUGH CONFIGURABLE OUTPUT DRIVERS

Supply sequencing is achieved with the [ADM1168](#) using the programmable driver outputs (PDOs) on the device as control signals for supplies. The output drivers can be used as logic enables or as FET drivers.

The sequence in which the PDOs are asserted (and, therefore, the supplies are turned on) is controlled by the sequencing engine (SE). The SE determines what action is taken with the PDOs, based on the condition of the [ADM1168](#) inputs. Therefore, the PDOs can be set up to assert when the SFDs are in tolerance, the correct input signals are received on the V<sub>X</sub> digital pins, and no warnings are received from any of the inputs of the device. The PDOs can be used for a variety of functions. The primary function is to provide enable signals for LDOs or dc-to-dc converters that generate supplies locally on a board. The PDOs can also be used to provide a PWRGD signal when all the SFDs are in tolerance or a RESET output if one of the SFDs goes out of specification (this can be used as a status signal for a DSP, FPGA, or other microcontroller).

The PDOs can be programmed to pull up to a number of different options. The outputs can be programmed as follows:

- Open drain (allowing the user to connect an external pull-up resistor)
- Open drain with weak pull-up to V<sub>DD</sub>
- Open drain with strong pull-up to V<sub>DD</sub>
- Open drain with weak pull-up to V<sub>Px</sub>
- Open drain with strong pull-up to V<sub>Px</sub>
- Strong pull-down to GND
- Internally charge-pumped high drive (12 V, PDO1 to PDO6 only)

The last option (available only on PDO1 to PDO6) allows the user to directly drive a voltage high enough to fully enhance an external NFET, which is used to isolate, for example, a card-side voltage from a backplane supply (a PDO can sustain greater than 10.5 V into a 1  $\mu$ A load). The pull-down switches can also be used to drive status LEDs directly.

The data driving each of the PDOs can come from one of three sources. The source can be enabled in the PDOxCFG configuration register (see the [AN-721 Application Note](#) for details).

The data sources are as follows:

- Output from the SE.
- Directly from the SMBus. A PDO can be configured so the SMBus has direct control over it. This enables software control of the PDOs. Therefore, a microcontroller can be used to initiate a software power-up/power-down sequence.
- On-chip clock. A 100 kHz clock is generated on the device. This clock can be made available on any of the PDOs. It can be used, for example, to clock an external device such as an LED.

### DEFAULT OUTPUT CONFIGURATION

All of the internal registers in an unprogrammed [ADM1168](#) device from the factory are set to 0. Because of this, the PDOx pins are pulled to GND by a weak (20 k $\Omega$ ), on-chip, pull-down resistor.

As the input supply to the [ADM1168](#) ramps up on V<sub>Px</sub> or V<sub>H</sub>, all PDOx pins behave as follows:

- Input supply = 0 V to 1.2 V. PDOs high impedance.
- Input supply = 1.2 V to 2.7 V. PDOs pulled to GND by a weak (20 k $\Omega$ ), on-chip, pull-down resistor.
- Supply > 2.7 V. Factory programmed devices continue to pull all PDOs to GND by a weak (20 k $\Omega$ ), on-chip, pull-down resistor. Programmed devices download current EEPROM configuration data, and the programmed setup is latched. The PDO then goes to the state demanded by the configuration, providing a known condition for the PDOs during power-up.

The internal pull-down can be overdriven with an external pull-up of suitable value tied from the PDOx pin to the required pull-up voltage. The 20 k $\Omega$  resistor must be accounted for in calculating a suitable value. For example, if PDOx must be pulled up to 3.3 V, and 5 V is available as an external supply, the pull-up resistor value is given by

$$3.3 \text{ V} = 5 \text{ V} \times 20 \text{ k}\Omega / (R_{UP} + 20 \text{ k}\Omega)$$

Therefore,

$$R_{UP} = (100 \text{ k}\Omega - 66 \text{ k}\Omega) / 3.3 \text{ V} = 10 \text{ k}\Omega$$

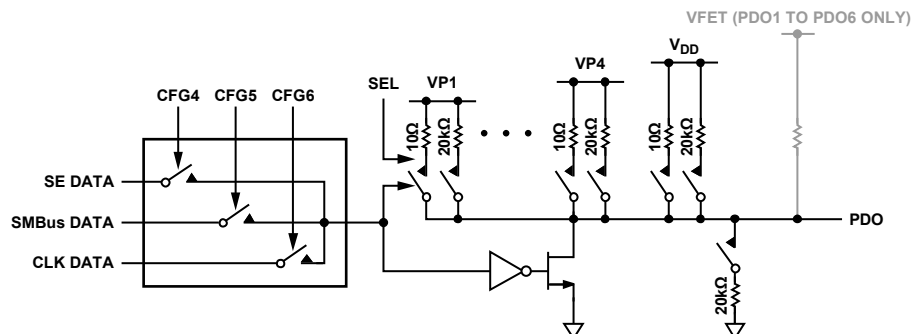


Figure 18. Programmable Driver Output

## SEQUENCING ENGINE

### OVERVIEW

The ADM1168 SE provides the user with powerful and flexible control of sequencing. The SE implements a state machine control of the PDO outputs with state changes conditional on input events. SE programs can enable complex control of boards such as power-up and power-down sequence control, fault event handling, and interrupt generation on warnings. A watchdog function that verifies the continued operation of a processor clock can be integrated into the SE program. The SE can also be controlled via the SMBus, giving software or firmware control of the board sequencing.

The SE state machine comprises 63 state cells. Each state has the following attributes:

- It monitors signals indicating the status of the eight input pins, VP1 to VP3, VH, and VX1 to VX4.
- It can be entered from any other state.
- Three exit routes move the state machine onto a next state: sequence detection, fault monitoring, and timeout.
- Delay timers for the sequence and timeout blocks can be programmed independently and changed with each state change. The range of timeouts is from 0 ms to 400 ms.
- The output condition of the eight PDO pins is defined and fixed within a state.
- It can transition from one state to the next in less than 20  $\mu$ s, which is the time needed to download a state definition from EEPROM to the SE.
- It can trigger a write of the black box fault and status registers into the black box section of EEPROM.

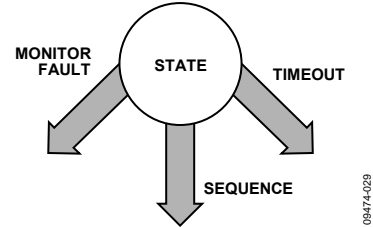


Figure 19. State Cell

The ADM1168 offers up to 63 state definitions. The signals monitored to indicate the status of the input pins are the outputs of the SFDs.

### WARNINGS

The SE also monitors warnings. These warnings can be generated when the ADC readings violate their limit register value or when the secondary voltage monitors on VPx and VH are triggered. The warnings are OR'ed together and are available as a single warning input to each of the three blocks that enable exiting a state.

### SMBus JUMP (UNCONDITIONAL JUMP)

The SE can be forced to advance to the next state unconditionally. This enables the user to force the SE to advance. Examples of the use of this feature include moving to a margining state or debugging a sequence. The SMBus jump or go-to command can be seen as another input to sequence and timeout blocks to provide an exit from each state.

Table 7. Sample Sequence State Entries

State	Sequence	Timeout	Monitor
IDLE1	If VX1 is low, go to State IDLE2.		
IDLE2	If VP1 is okay, go to State EN3V3.		
EN3V3	If VP2 is okay, go to State EN2V5.	If VP2 is not okay after 10 ms, go to State DIS3V3.	If VP1 is not okay, go to State IDLE1.
DIS3V3	If VX1 is high, go to State IDLE1.		
EN2V5	If VP3 is okay, go to State PWRGD.	If VP3 is not okay after 20 ms, go to State DIS2V5.	If VP1 or VP2 is not okay, go to State FSEL2.
DIS2V5	If VX1 is high, go to State IDLE1.		
FSEL1	If VP3 is not okay, go to State DIS2V5.		If VP1 or VP2 is not okay, go to State FSEL2.
FSEL2	If VP2 is not okay, go to State DIS3V3.		If VP1 is not okay, go to State IDLE1.
PWRGD	If VX1 is high, go to State DIS2V5.		If VP1, VP2, or VP3 is not okay, go to State FSEL1.

## SEQUENCING ENGINE APPLICATION EXAMPLE

The application in this section demonstrates the operation of the SE. Figure 21 shows how the simple building block of a single SE state can be used to build a power-up sequence for a three-supply system. Table 8 lists the PDO outputs for each state in the same SE implementation. In this system, a good 5 V supply on the VP1 pin and the VX1 pin held low are the triggers required to start a power-up sequence. The sequence next turns on the 3.3 V supply, then the 2.5 V supply (assuming successful turn on of the 3.3 V supply). When all three supplies have turned on correctly, the PWRGD state is entered, where the SE remains until a fault occurs on one of the three supplies or until it is instructed to go through a power-down sequence by VX1 going high.

Faults are dealt with throughout the power-up sequence on a case-by-case basis. The following three sections (the Sequence Detector section, the Monitoring Fault Detector section, and the Timeout Detector section) describe the individual blocks and use the sample application shown in Figure 21 to demonstrate the actions of the state machine.

### Sequence Detector

The sequence detector block is used to detect when a step in a sequence has been completed. It looks for one of the SE inputs to change state and is most often used as the gate for successful progress through a power-up or power-down sequence. A timer block that is included in this detector can insert delays into a power-up or power-down sequence, if required. Timer delays can be set from 10  $\mu$ s to 400 ms. Figure 20 is a block diagram of the sequence detector.

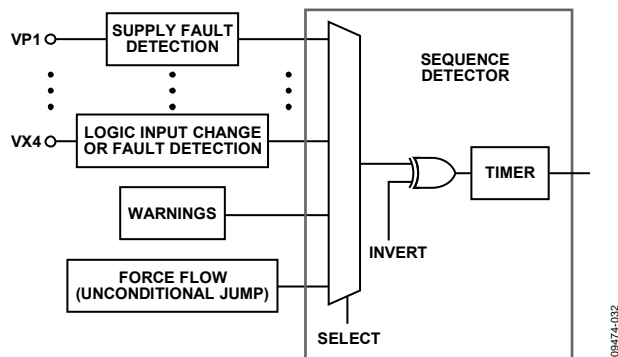


Figure 20. Sequence Detector Block Diagram

If a timer delay is specified, the input to the sequence detector must remain in the defined state for the duration of the timer delay. If the input changes state during the delay, the timer is reset.

The sequence detector can also help to identify monitoring faults. In the sample application shown in Figure 21, the FSEL1 and FSEL2 states first identify which of the VP1, VP2, or VP3 pins has faulted, and then they take appropriate action.

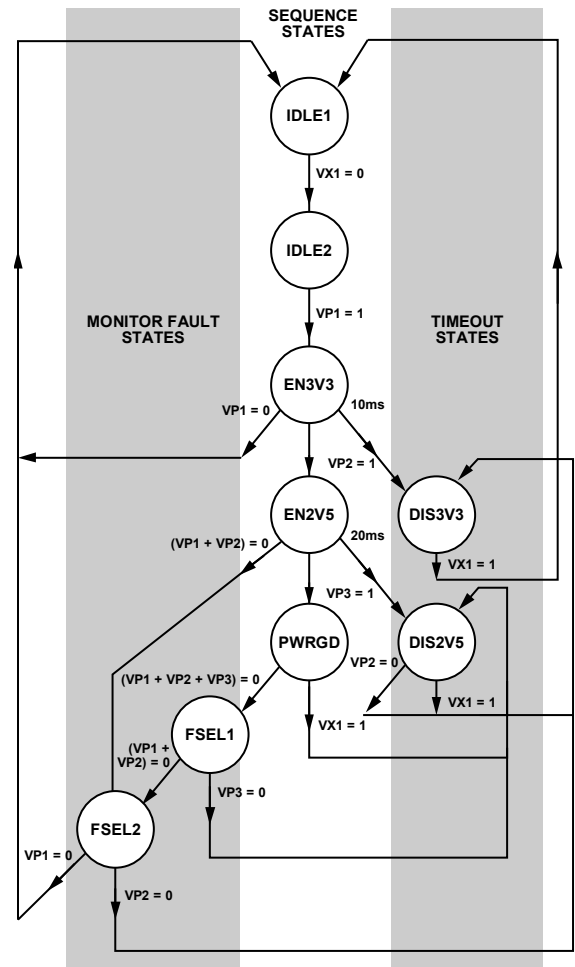


Figure 21. Sample Application Flow Diagram

Table 8. PDO Outputs for Each State

PDO Outputs	IDLE1	IDLE2	EN3V3	EN2V5	DIS3V3	DIS2V5	PWRGD	FSEL1	FSEL2
PDO1 = 3V3ON	0	0	1	1	0	1	1	1	1
PDO2 = 2V5ON	0	0	0	1	1	0	1	1	1
PDO3 = FAULT	0	0	0	0	1	1	0	1	1



### Monitoring Fault Detector

The monitoring fault detector block is used to detect a failure on an input. The logical function implementing this is a wide OR gate that can detect when an input deviates from its expected condition. The clearest demonstration of the use of this block is in the PWRGD state, where the monitor block indicates that a failure on one or more of the VPx, VXx, or VH inputs has occurred.

No programmable delay is available in this block because the triggering of a fault condition is likely to be caused by a supply falling out of tolerance. In this situation, the device needs to react as quickly as possible. Some latency occurs when moving out of this state, however, because it takes a finite amount of time (~20  $\mu$ s) for the state configuration to download from the EEPROM into the SE. Figure 22 is a block diagram of the monitoring fault detector.

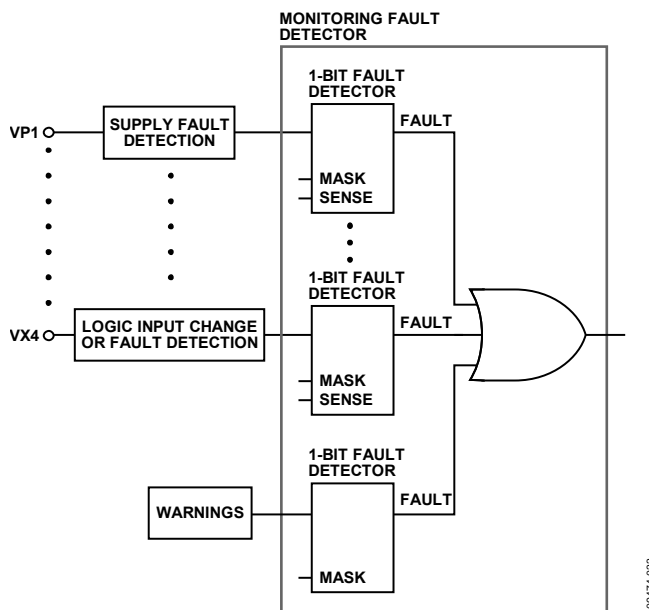


Figure 22. Monitoring Fault Detector Block Diagram

### Timeout Detector

The timeout detector allows the user to trap a failure to ensure proper progress through a power-up or power-down sequence.

In the sample application shown in Figure 21, the timeout next-state transition is from the EN3V3 and EN2V5 states. For the EN3V3 state, the signal 3V3ON is asserted on the PDO1 output pin upon entry to this state to turn on a 3.3 V supply.

This supply rail is connected to the VP2 pin, and the sequence detector looks for the VP2 pin to go above its undervoltage threshold, which is set in the supply fault detector (SFD) attached to that pin.

The power-up sequence progresses when this change is detected. If, however, the supply fails (perhaps due to a short circuit overloading this supply), the timeout block traps the problem. In this example, if the 3.3 V supply fails within 10 ms, the SE moves to the DIS3V3 state and turns off this supply by bringing PDO1 low. It also indicates that a fault has occurred by taking PDO3 high. Timeout delays of 100  $\mu$ s to 400 ms can be programmed.

### FAULT AND STATUS REPORTING

The ADM1168 has a fault latch for recording faults. Two registers, FSTAT1 and FSTAT2, are set aside for this purpose. A single bit is assigned to each input of the device, and a fault on that input sets the relevant bit. The contents of the fault register can be read out over the SMBus to determine which input(s) faulted. The fault register can be enabled or disabled in each state. To latch data from one state, ensure that the fault latch is disabled in the following state. This ensures that only real faults are captured and not, for example, undervoltage conditions that may be present during a power-up or power-down sequence.

The ADM1168 also has a number of status registers. These include more detailed information, such as whether an undervoltage or overvoltage fault is present on a particular input. The status registers also include information on ADC limit faults.

There are two sets of these registers with different behaviors. The first set of status registers is not latched in any way and, therefore, can change at any time in response to changes on the inputs. These registers provide information such as the UV and OV state of the inputs, the digital state of the GPI VXx inputs, and the ADC warning limit status.

The second set of registers update each time the sequence engine changes state and are latched until the next state change. The second set of registers provides the same information as the first set, but in a more compact form. The reason for this is that these registers are used by the black box feature when writing status information for the previous state into EEPROM.

See the AN-721 Application Note for full details about the ADM1168 registers.

### NONVOLATILE BLACK BOX FAULT RECORDING

A section of EEPROM, from Address 0xF900 to Address 0xF9FF, is provided that by default can be used to store user-defined settings and information. Part of this section of EEPROM, Address 0xF980 to Address 0xF9FF, can instead be used to store up to 16 fault records.

Any sequencing engine state can be designated as a black box write state. Each time the sequence engine enters that state, a fault record is written into EEPROM. The fault record provides a snapshot of the entire ADM1168 state at the point in time when the last state was exited, just prior to entering the designated black box write state. A fault record contains the following information:

- A flag bit set to 0 after the fault record has been written.
- The state number of the previous state prior to the fault record write state.
- Did a sequence, timeout, or monitor condition cause the previous state to exit?
- UVSTATx and OVSTATx input comparator status.
- VXx GPISTAT status.
- LIMSTATx status.
- A checksum byte.

Each fault record contains eight bytes, with each byte taking typically about 250  $\mu$ s to write to EEPROM, for a total write time of about 2 ms. After the black box begins to write a fault record into EEPROM, the ADM1168 ensures that it is complete before attempting to write any additional fault records. This means that if consecutive sequencing engine states are designated as black box write states, then a time delay must be used in the first state to ensure that the fault record is written before moving to the next state.

When the ADM1168 powers on initially, it performs a search to find the first fault record that has not been written to. It does this by checking the flag bit in each fault record until it finds one where the flag bit is 1. The first fault record is stored at Address 0xF980, and at multiples of eight bytes after that, with the last record stored at Address 0xF9F8.

The fault recorder is only able to write in the EEPROM. It is not able to erase the EEPROM prior to writing the fault record. Therefore, to ensure correct operation, it is important that the fault record EEPROM is erased prior to use. When all the EEPROM locations for the fault records are used, no more fault records are written. This ensures that the first fault in any cascading fault is stored and not overwritten and lost.

To avoid the fault recorder filling up and fault records being lost, an application can periodically poll the ADM1168 to determine if there are fault records to be read. Alternatively, one of the PDOx outputs can be used to generate an interrupt for a processor in the fault record write state to signal the need to come and read one or more fault records.

After reading fault records during normal operation, two things must be done before the fault recorder will be able to reuse the EEPROM locations. First, the EEPROM section must be erased. The fault recorder must then be reset so that it performs its search again for the first unused location of EEPROM that is available to store a fault record.

## BLACK BOX WRITES WITH NO EXTERNAL SUPPLY

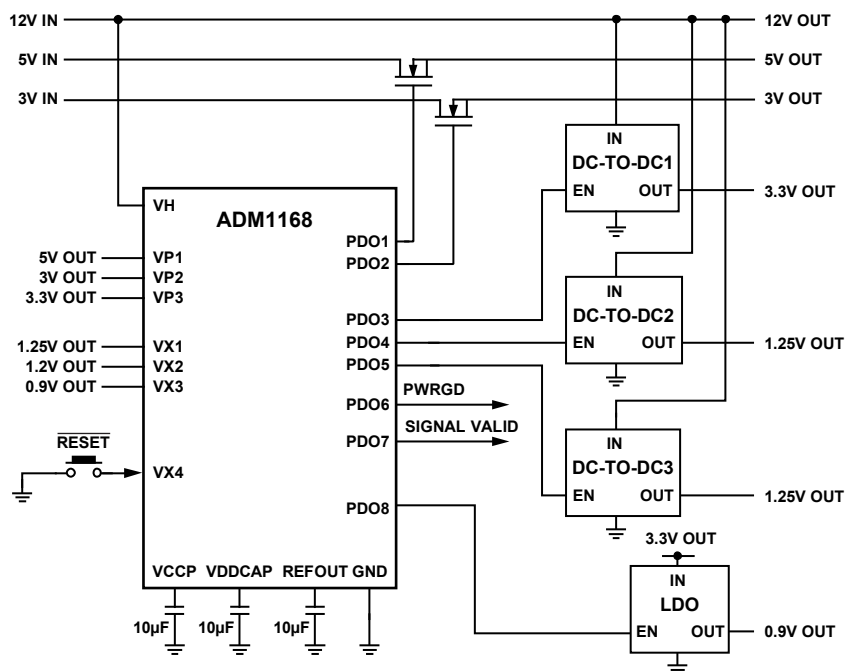
In cases where all the input supplies fail, for example, if the card has been removed from a powered backplane, the state machine can be programmed to trigger a write into the black box EEPROM. The decoupling capacitors on the rail that power the ADM1168 and other loads on the board form an energy reservoir. Depending on the other loads on the board and their behavior as the supply rails drop, there may be sufficient energy in the decoupling capacitors to allow the ADM1168 to write a complete fault record (eight bytes of data).

Typically, it takes 2 ms to write to the eight bytes of a fault record. If the ADM1168 is powered using a 12 V supply on the VH pin, then a UV threshold at 6 V can be set and used as the state machine trigger to start writing a fault record to EEPROM. The higher the threshold is, the earlier the black box write begins, and the more energy available in the decoupling capacitors to ensure it completes successfully.

Provided the VH supply, or another supply connected to a VPx pin, remains above 3.0 V during the time to write, the entire fault record is always written to the EEPROM. In many cases, there should be sufficient decoupling capacitors on a board to power the ADM1168 as it writes into the EEPROM.

In cases where the decoupling capacitors are not able to supply sufficient energy after the board is removed to ensure a complete fault record is written, the value of the capacitor on VDDCAP may be increased. In the worst case, assuming that no energy is supplied to the ADM1168 by the external decoupling capacitors, but that VDDCAP has 4.75 V on it, then a 47  $\mu$ F is sufficient to guarantee that a single complete black box record can be written to the EEPROM.

## APPLICATIONS DIAGRAM



09474-068

## COMMUNICATING WITH THE ADM1168

### CONFIGURATION DOWNLOAD AT POWER-UP

The configuration of the ADM1168 (undervoltage/overvoltage thresholds, glitch filter timeouts, and PDO configurations) is dictated by the contents of the RAM. The RAM comprises digital latches that are local to each of the functions on the device. The latches are double-buffered and have two identical latches, Latch A and Latch B. Therefore, when an update to a function occurs, the contents of Latch A are updated first, and then the contents of Latch B are updated with identical data. The advantages of this architecture are explained in detail in the Updating the Configuration section.

The two latches are volatile memory and lose their contents at power-down. Therefore, the configuration in the RAM must be restored at power-up by downloading the contents of the EEPROM (nonvolatile memory) to the local latches. This download occurs in steps, as follows:

1. With no power applied to the device, the PDOs are all high impedance.
2. When 1.2 V appears on any of the inputs connected to the VDD arbitrator (VH or VPx), the PDOs are all weakly pulled to GND with a 20 kΩ resistor.
3. When the supply rises above the undervoltage lockout of the device (UVLO is 2.5 V), the EEPROM starts to download to the RAM.
4. The EEPROM downloads its contents to all Latch As.
5. When the contents of the EEPROM are completely downloaded to the Latch As, the device controller signals all Latch As to download to all Latch Bs simultaneously, completing the configuration download.
6. At 0.5 ms after the configuration download completes, the first state definition is downloaded from EEPROM into the SE.

Note that any attempt to communicate with the device prior to the completion of the download causes the ADM1168 to issue a no acknowledge (NACK).

### UPDATING THE CONFIGURATION

After power-up, with all the configuration settings loaded from the EEPROM into the RAM registers, the user may need to alter the configuration of functions on the ADM1168, such as changing the undervoltage or overvoltage limit of an SFD, changing the fault output of an SFD, or adjusting the rise time delay of one of the PDOs.

The ADM1168 provides several options that allow the user to update the configuration over the SMBus interface. The following three options are controlled in the UPDCFG register.

#### Option 1

Update the configuration in real time. The user writes to the RAM across the SMBus, and the configuration is updated immediately.

#### Option 2

Update the Latch As without updating the Latch Bs. With this method, the configuration of the ADM1168 remains unchanged and continues to operate in the original setup until the instruction is given to update the Latch Bs.

#### Option 3

Change the EEPROM register contents without changing the RAM contents, and then download the revised EEPROM contents to the RAM registers. With this method, the configuration of the ADM1168 remains unchanged and continues to operate in the original setup until the instruction is given to update the RAM.

The instruction to download from the EEPROM in the Option 3 section is also a useful way to restore the original EEPROM contents if revisions to the configuration are unsatisfactory. For example, if the user needs to alter an overvoltage threshold, the RAM register can be updated, as described in the Option 1 section. However, if the user is not satisfied with the change and wants to revert to the original programmed value, the device controller can issue a command to download the EEPROM contents to the RAM again, as described in the Option 3 section, restoring the ADM1168 to its original configuration.

The topology of the ADM1168 makes this type of operation possible. The local, volatile registers (RAM) are all double-buffered latches. Setting Bit 0 of the UPDCFG register to leaves the double buffered latches open at all times, allowing the registers to be updated continuously as they are written to. If Bit 0 is set to 0 when a RAM write occurs across the SMBus, only the first side of the double-buffered latch is written to. The user must then write a 1 to Bit 1 of the UPDCFG register. This generates a pulse to update all the second latches at once. EEPROM writes occur in a similar way.

The final bit in this register can enable or disable EEPROM page erasure. If this bit is set high, the contents of an EEPROM page can all be set to 1. If this bit is set low, the contents of a page cannot be erased, even if the command code for page erasure is programmed across the SMBus. The bit map for the UPDCFG register is shown in the AN-721 Application Note. A flow diagram for download at power-up and subsequent configuration updates is shown in Figure 24.

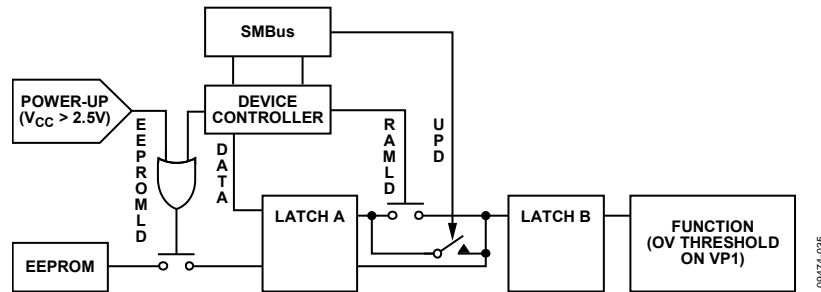


Figure 24. Configuration Update Flow Diagram

## UPDATING THE SEQUENCING ENGINE

SE functions are not updated in the same way as regular configuration latches. The SE has its own dedicated 512-byte EEPROM for storing state definitions, providing 63 individual states, each with a 64-bit word (one state is reserved). At power-up, the first state is loaded from the SE EEPROM into the engine itself. When the conditions of this state are met, the next state is loaded from the EEPROM into the engine and so on. The loading of each new state takes approximately 10  $\mu$ s.

To alter a state, the required changes must be made directly to the EEPROM. RAM for each state does not exist. The relevant alterations must be made to the 64-bit word, which is then uploaded directly to the EEPROM.

## INTERNAL REGISTERS

The [ADM1168](#) contains a large number of data registers. The principal registers are the address pointer register and the configuration registers.

### Address Pointer Register

The address pointer register contains the address that selects one of the other internal registers. When writing to the [ADM1168](#), the first byte of data is always a register address that is written to the address pointer register.

### Configuration Registers

The configuration registers provide control and configuration for various operating parameters of the [ADM1168](#). See the [AN-721 Application Note](#).

## EEPROM

The [ADM1168](#) has two 512-byte cells of nonvolatile, electrically erasable, programmable, read-only memory (EEPROM), from Address 0xF800 to Register Address 0xFBFF. The EEPROM is used for permanent storage of data that is not lost when the [ADM1168](#) is powered down. One EEPROM cell, 0xF800 to 0xF9FF, contains the configuration data, user information and, if enabled, any fault records of the device; the other section, 0xFA00 to 0xFBFF, contains the state definitions for the SE. Although referred to as read-only memory, the EEPROM can be written to, as well as read from, using the serial bus in exactly the same way as the other registers.

The major differences between the EEPROM and other registers are as follows:

- An EEPROM location must be blank before it can be written to. If it contains data, the data must first be erased.
- Writing to the EEPROM is slower than writing to the RAM.
- Writing to the EEPROM should be restricted because it has a limited write/cycle life of typically 10,000 write operations due to the usual EEPROM wear-out mechanisms.

The first EEPROM is split into 16 (0 to 15) pages of 32 bytes each. Page 0 to Page 4, from Address 0xF800 to Address 0xF89F, hold the configuration data for the applications on the [ADM1168](#) (such as the SFDs and PDOs). These EEPROM addresses are the same as the RAM register addresses, prefixed by F8. Page 5 to Page 7, from Address 0xF8A0 to Address 0xF8FF, are reserved.

Page 8 to Page 11 are available for customer use to store any information that may be required by the customer in their application. Customers can store information on Page 12 to Page 15, or these pages can store the fault records written by the sequencing engine if users have decided to enable writing of the fault records for different states.

Data can be downloaded from the EEPROM to the RAM in one of the following ways:

- At power-up, when Page 0 to Page 4 are downloaded.
- By setting Bit 0 of the UDOWNLD register (0xD8), which performs a user download of Page 0 to Page 4.

When the sequence engine is enabled, it is not possible to access the section of EEPROM from Address 0xFA00 to Address 0xFBFF. The sequence engine must be halted before it is possible to read or write to this range. Attempting to read or write to this range if the sequence engine is not halted generates a no acknowledge, or NACK.

Read/write access to the configuration and user EEPROM ranges from Address 0xF800 to Address 0xF89F and Address 0xF900 to Address 0xF9FF depends on whether the black box fault recorder is enabled. If the fault recorder is enabled and one or more states have been set as fault record trigger states, then it is not possible to access any EEPROM location in this range without first halting the black box. Attempts to read or write to this EEPROM range while the fault recorder is operating are acknowledged by the device but do not return any useful data or modify the EEPROM in any way.

If none of the states are set as fault record trigger states, then the black box is considered disabled, and read/write access is allowed without having to halt the black box fault recorder.

## SERIAL BUS INTERFACE

The ADM1168 is controlled via the serial system management bus (SMBus) and is connected to this bus as a slave device under the control of a master device. It takes approximately 1 ms after power-up for the ADM1168 to download from its EEPROM. Therefore, access to the ADM1168 is restricted until the download is complete.

### Identifying the ADM1168 on the SMBus

The ADM1168 has a 7-bit serial bus slave address (see Table 9). The device is powered up with a default serial bus address. The five MSBs of the address are set to 10001; the two LSBs are determined by the logical states of Pin A1 and Pin A0. This allows the connection of four ADM1168s to one SMBus.

**Table 9. Serial Bus Slave Address**

A1 Pin	A0 Pin	Hex Address	7-Bit Address <sup>1</sup>
Low	Low	0x88	1000100x
Low	High	0x8A	1000101x
High	Low	0x8C	1000110x
High	High	0x8E	1000111x

<sup>1</sup>x = read/write bit. The address is shown only as the first seven MSBs.

The device also has several identification registers (read-only) that can be read across the SMBus. Table 10 lists these registers with their values and functions.

**Table 10. Identification Register Values and Functions**

Name	Address	Value	Function
MANID	0xF4	0x41	Manufacturer ID for Analog Devices
REVID	0xF5	0x10	Silicon revision
MARK1	0xF6	0x00	Software brand
MARK2	0xF7	0x00	Software brand

### General SMBus Timing

Figure 25, Figure 26, and Figure 27 are timing diagrams for general read and write operations using the SMBus. The SMBus specification defines specific conditions for different types of read and write operations, which are discussed in the Write Operations and Read Operations sections.

The general SMBus protocol operates in the following three steps.

#### Step 1

The master initiates data transfer by establishing a start condition, defined as a high-to-low transition on the serial data line SDA, while the serial clock line SCL remains high. This indicates that a data stream follows. All slave peripherals connected to the serial bus respond to the start condition and shift in the next eight bits, consisting of a 7-bit slave address (MSB first) plus an R/W bit. This bit determines the direction of the data transfer, that is, whether data is written to or read from the slave device (0 = write, 1 = read).

The peripheral whose address corresponds to the transmitted address responds by pulling the data line low during the low period before the ninth clock pulse, known as the acknowledge bit, and by holding it low during the high period of this clock pulse.

All other devices on the bus remain idle while the selected device waits for data to be read from or written to it. If the R/W bit is a 0, the master writes to the slave device. If the R/W bit is a 1, the master reads from the slave device.

#### Step 2

Data is sent over the serial bus in sequences of nine clock pulses, eight bits of data followed by an acknowledge bit from the slave device. Data transitions on the data line must occur during the low period of the clock signal and remain stable during the high period because a low-to-high transition when the clock is high may be interpreted as a stop signal. If the operation is a write operation, the first data byte after the slave address is a command byte. This command byte tells the slave device what to expect next. It may be an instruction telling the slave device to expect a block write, or it may be a register address that tells the slave where subsequent data is to be written. Because data can flow in only one direction, as defined by the R/W bit, sending a command to a slave device during a read operation is not possible. Before a read operation, it may be necessary to perform a write operation to tell the slave what sort of read operation to expect and the address from which data is to be read.

#### Step 3

When all data bytes have been read or written, stop conditions are established. In write mode, the master pulls the data line high during the 10<sup>th</sup> clock pulse to assert a stop condition. In read mode, the master device releases the SDA line during the low period before the ninth clock pulse, but the slave device does not pull it low. This is known as a no acknowledge. The master then takes the data line low during the low period before the 10<sup>th</sup> clock pulse and then high during the 10<sup>th</sup> clock pulse to assert a stop condition.

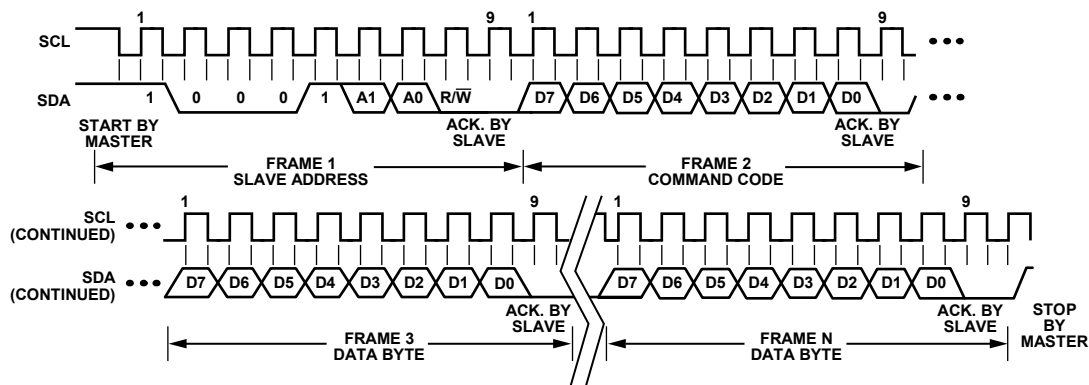


Figure 25. General SMBus Write Timing Diagram

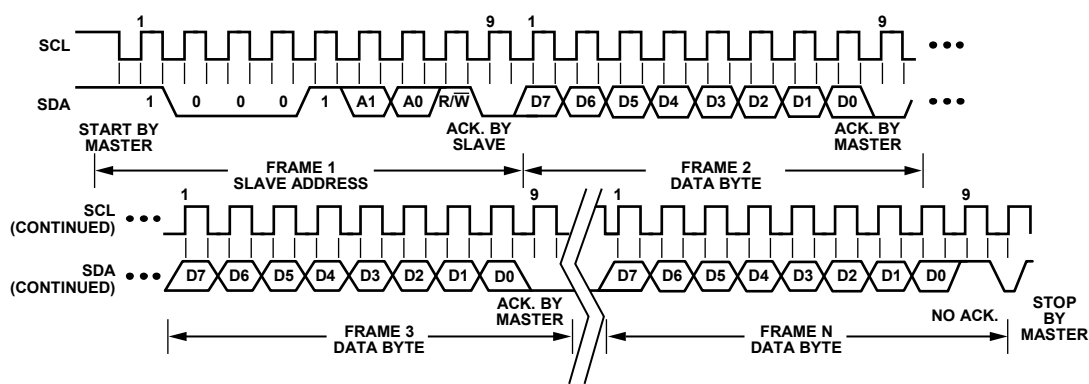


Figure 26. General SMBus Read Timing Diagram

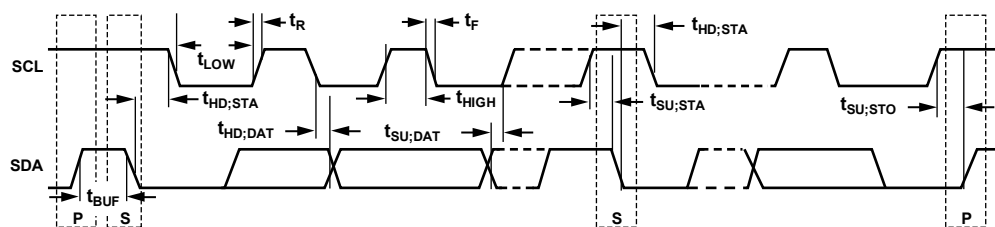


Figure 27. Serial Bus Timing Diagram



## SMBus PROTOCOLS FOR RAM AND EEPROM

The ADM1168 contains volatile registers (RAM) and nonvolatile registers (EEPROM). User RAM occupies Address 0x00 to Address 0xDF; and the EEPROM occupies Address 0xF800 to Address 0xFBFF.

Data can be written to and read from both the RAM and the EEPROM as single data bytes. Data can be written only to unprogrammed EEPROM locations. To write new data to a programmed location, the location contents must first be erased. EEPROM erasure cannot be done at the byte level. The EEPROM is arranged as 32 pages of 32 bytes each, and an entire page must be erased.

Page erasure is enabled by setting Bit 2 in the UPDCFG register (Address 0x90) to 1. If this bit is not set, page erasure cannot occur, even if the command byte (0xFE) is programmed across the SMBus.

## WRITE OPERATIONS

The SMBus specification defines several protocols for different types of read and write operations. The following abbreviations are used in Figure 28 to Figure 36:

- S = Start
- P = Stop
- R = Read
- W = Write
- A = Acknowledge
- $\overline{A}$  = No acknowledge

The ADM1168 uses the following SMBus write protocols.

### Send Byte

In a send byte operation, the master device sends a single command byte to a slave device, as follows:

1. The master device asserts a start condition on SDA.
2. The master sends the 7-bit slave address followed by the write bit (low).
3. The addressed slave device asserts an acknowledge (ACK) on SDA.
4. The master sends a command code.
5. The slave asserts ACK on SDA.
6. The master asserts a stop condition on SDA and the transaction ends.

In the ADM1168, the send byte protocol is used for two purposes:

- To write a register address to the RAM for a subsequent single byte read from the same address, or for a block read or a block write starting at that address, as shown in Figure 28.

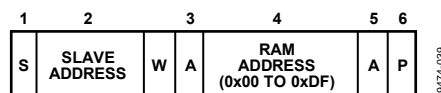


Figure 28. Setting a RAM Address for Subsequent Read

- To erase a page of EEPROM memory. EEPROM memory can be written to only if it is unprogrammed. Before writing to one or more EEPROM memory locations that are already programmed, the page(s) containing those locations must first be erased. EEPROM memory is erased by writing a command byte.

The master sends a command code telling the slave device to erase the page. The ADM1168 command code for a page erasure is 0xFE (1111 1110). Note that for a page erasure to take place, the page address must be given in the previous write word transaction (see the Write Byte/Word section). In addition, Bit 2 in the UPDCFG register (Address 0x90) must be set to 1. See Figure 29.

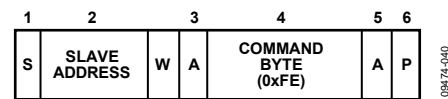


Figure 29. EEPROM Page Erasure

As soon as the ADM1168 receives the command byte, page erasure begins. The master device can send a stop command as soon as it sends the command byte. Page erasure takes approximately 20 ms. If the ADM1168 is accessed before erasure is complete, it responds with a no acknowledge (NACK).

### Write Byte/Word

In a write byte/word operation, the master device sends a command byte and one or two data bytes to the slave device, as follows:

1. The master device asserts a start condition on SDA.
2. The master sends the 7-bit slave address followed by the write bit (low).
3. The addressed slave device asserts ACK on SDA.
4. The master sends a command code.
5. The slave asserts ACK on SDA.
6. The master sends a data byte.
7. The slave asserts ACK on SDA.
8. The master sends a data byte or asserts a stop condition.
9. The slave asserts ACK on SDA.
10. The master asserts a stop condition on SDA to end the transaction.

In the ADM1168, the write byte/word protocol is used for three purposes:

- To write a single byte of data to the RAM. In this case, the command byte is RAM Address 0x00 to RAM Address 0xDF, and the only data byte is the actual data, as shown in Figure 30.

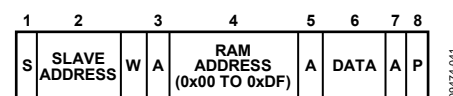


Figure 30. Single Byte Write to the RAM



- To set up a 2-byte EEPROM address for a subsequent read, write, block read, block write, or page erase. In this case, the command byte is the high byte of EEPROM Address 0xF8 to EEPROM Address 0xFB. The only data byte is the low byte of the EEPROM address, as shown in Figure 31.

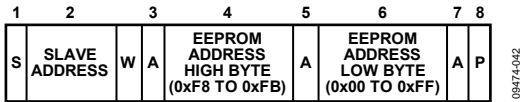


Figure 31. Setting an EEPROM Address

Because a page consists of 32 bytes, only the three MSBs of the address low byte are important for page erasure. The lower five bits of the EEPROM address low byte specify the addresses within a page and are ignored during an erase operation.

- To write a single byte of data to the EEPROM. In this case, the command byte is the high byte of EEPROM Address 0xF8 to EEPROM Address 0xFB. The first data byte is the low byte of the EEPROM address, and the second data byte is the actual data, as shown in Figure 32.

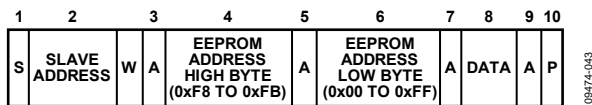


Figure 32. Single Byte Write to the EEPROM

### Block Write

In a block write operation, the master device writes a block of data to a slave device, as shown in Figure 34. The start address for a block write must have been set previously. In the ADM1168, a send byte operation sets a RAM address, and a write byte/word operation sets an EEPROM address as follows:

- The master device asserts a start condition on SDA.
- The master sends the 7-bit slave address followed by the write bit (low).
- The addressed slave device asserts ACK on SDA.
- The master sends a command code that tells the slave device to expect a block write. The ADM1168 command code for a block write is 0xFC (1111 1100).
- The slave asserts ACK on SDA.
- The master sends a data byte that tells the slave device how many data bytes are being sent. The SMBus specification allows a maximum of 32 data bytes in a block write.
- The slave asserts ACK on SDA.
- The master sends N data bytes.
- The slave asserts ACK on SDA after each data byte.
- The master asserts a stop condition on SDA to end the transaction.

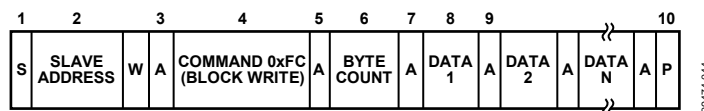


Figure 34. Block Write to the EEPROM or RAM

Unlike some EEPROM devices that limit block writes to within a page boundary, there is no limitation on the start address when performing a block write to EEPROM, except when

- There must be at least N locations from the start address to the highest EEPROM address (0xFBFF) to avoid writing to invalid addresses.
- An address crosses a page boundary. In this case, both pages must be erased before programming.

Note that the ADM1168 features a clock extend function for writes to the EEPROM. Programming an EEPROM byte takes approximately 250  $\mu$ s, which limits the SMBus clock for repeated or block write operations. The ADM1168 pulls SCL low and extends the clock pulse when it cannot accept any more data.

### READ OPERATIONS

The ADM1168 uses the following SMBus read protocols.

#### Receive Byte

In a receive byte operation, the master device receives a single byte from a slave device, as follows:

- The master device asserts a start condition on SDA.
- The master sends the 7-bit slave address followed by the read bit (high).
- The addressed slave device asserts ACK on SDA.
- The master receives a data byte.
- The master asserts NACK on SDA.
- The master asserts a stop condition on SDA, and the transaction ends.

In the ADM1168, the receive byte protocol is used to read a single byte of data from a RAM or EEPROM location whose address has previously been set by a send byte or write byte/word operation, as shown in Figure 33.

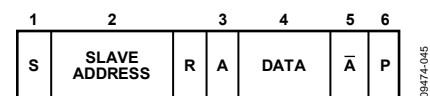


Figure 33. Single Byte Read from the EEPROM or RAM

Block Read

In a block read operation, the master device reads a block of data from a slave device. The start address for a block read must have been set previously. In the ADM1168, this is done by a send byte operation to set a RAM address, or a write byte/word operation to set an EEPROM address. The block read operation itself consists of a send byte operation that sends a block read command to the slave, immediately followed by a repeated start and a read operation that reads out multiple data bytes, as follows:

- 1. The master device asserts a start condition on SDA.
- 2. The master sends the 7-bit slave address followed by the write bit (low).
- 3. The addressed slave device asserts ACK on SDA.
- 4. The master sends a command code that tells the slave device to expect a block read. The ADM1168 command code for a block read is 0xFD (1111 1101).
- 5. The slave asserts ACK on SDA.
- 6. The master asserts a repeat start condition on SDA.
- 7. The master sends the 7-bit slave address followed by the read bit (high).
- 8. The slave asserts ACK on SDA.
- 9. The ADM1168 sends a byte-count data byte that tells the master how many data bytes to expect. The ADM1168 always returns 32 data bytes (0x20), which is the maximum allowed by the SMBus 1.1 specification.
- 10. The master asserts ACK on SDA.
- 11. The master receives 32 data bytes.
- 12. The master asserts ACK on SDA after each data byte.
- 13. The master asserts a stop condition on SDA to end the transaction. See Figure 35.

Error Correction

The ADM1168 provides the option of issuing a packet error checking (PEC) byte after a write to the RAM, a write to the EEPROM, a block write to the RAM/EEPROM, or a block read from the RAM/EEPROM. This option enables the user to verify that the data received by or sent from the ADM1168 is correct. The PEC byte is an optional byte sent after that last data byte has been written to or read from the ADM1168. The protocol is the same as a block read for Step 1 to Step 12 and then proceeds as follows:

- 13. The ADM1168 issues a PEC byte to the master. The master checks the PEC byte and issues another block read, if the PEC byte is incorrect.
- 14. A no acknowledge (NACK) is generated after the PEC byte to signal the end of the read.
- 15. The master asserts a stop condition on SDA to end the transaction.

Note that the PEC byte is calculated using CRC-8. The frame check sequence (FCS) conforms to CRC-8 by the polynomial

$$C(x) = x^8 + x^2 + x^1 + 1$$

See the SMBus 1.1 specification for details.

An example of a block read with the optional PEC byte is shown in Figure 36.

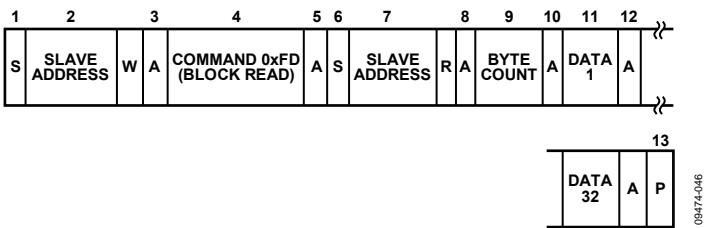


Figure 35. Block Read from the EEPROM or RAM

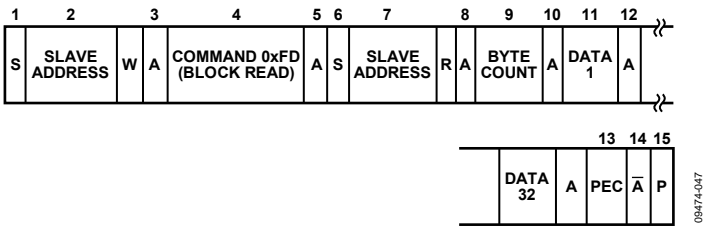
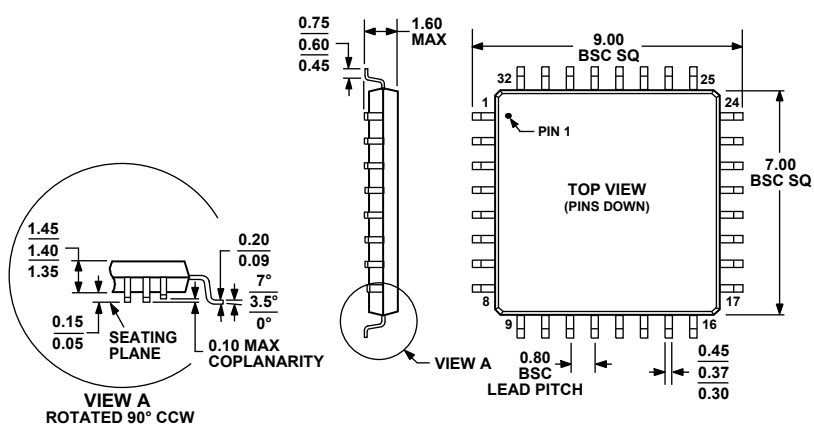


Figure 36. Block Read from the EEPROM or RAM with PEC

## OUTLINE DIMENSIONS



COMPLIANT TO JEDEC STANDARDS MS-026-BBA

Figure 37. 32-Lead Low Profile Quad Flat Package [LQFP]  
(ST-32-2)

Dimensions shown in millimeters

## ORDERING GUIDE

Model <sup>1</sup>	Temperature Range	Package Description	Package Option
ADM1168ASTZ	–40°C to +85°C	32-Lead Low Profile Quad Flat Package [LQFP]	ST-32-2
ADM1168ASTZ-RL7	–40°C to +85°C	32-Lead Low Profile Quad Flat Package [LQFP]	ST-32-2
EVAL-ADM1168LQEBZ		Evaluation Board	

<sup>1</sup> Z = RoHS Compliant Part.

# Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

[Analog Devices Inc.:](#)

[ADM1168ASTZ](#) [ADM1168ASTZ-RL7](#) [EVAL-ADM1168LQEBZ](#)