TABLE OF CONTENTS

REVISION HISTORY

12/03—Data sheet changed from Rev. A to Rev. B

Changes to Specifications 5

Changes to Outputs section 33

Updated Outline Dimensions 50

5/03—Data sheet changed from Rev. 0 to Rev. A.

Changes to Features 1
Changes to Specifications 5
Changes to Figure 1 4
Changes to Absolute Maximum Ratings 7
Changes to Figures 2, 8, 15–16 8–10
Changes to Figure 17 11
Changes to Programmable Supply Fault Detectors section 11
Changes to Figure 18 12
Changes to Figure 19 13
Change to Table 9 15
Change to Table 14 16
Change to Table 19 17
Changes to Programmable Driver Outputs section 33
Change to Table 40 34
Changes to Figure 25–26 43
Changes to Figure 37 47
Changes to Table 58 49
Changes to Ordering Guide section 50

Revision 0: Initial Version

GENERAL DESCRIPTION

(continued from Page 1)

All of the inputs and outputs described previously are controlled by the programmable logic block array (PLBA). This is the logic core of the ADM1060. It is comprised of nine macrocells, one for each PDO. These macrocells are essentially just wide AND gates. Any/all of the inputs can be used as an input to these macrocells. The output of a macrocell can also be used as an input to any macrocell other than itself (an input to itself would result in a nonterminating loop). The PLBA outputs control the PDOs of the ADM1060 via delay blocks, where a delay of 0 ms to 500 ms can be programmed on the rising and/or the falling edge of the data. This results in a very flexible sequencing ability. Thus, for instance, PDO1 can be programmed so that it will not assert until the VP2, VP3, and VP4 supplies are in tolerance; VB1 and VH have been in tolerance for 200 ms; and PDO7 has already been asserted. A simple sequencing operation would be to daisy-chain each PLB output into the input of the next PLB such that PDO9 does not assert until PDO8 asserts, which in turn does not assert until PDO7 asserts, and so on.

All of the functional capability described here is programmable through the industry-standard 2-wire bus (SMBus) provided. Device settings can be written to EEPROM memory for automatic programming of the device on power-up. The EEPROM is organized in 512 bytes, half of which are used to program all of the functions on the ADM1060. The other 256 bytes of EEPROM are for general-purpose system use such as date codes and system ID. Read/write access to this is also via the 2-wire interface. In addition, each output state can be directly overdriven from the serial interface, allowing a further level of control, as in a system controlled soft power-down.

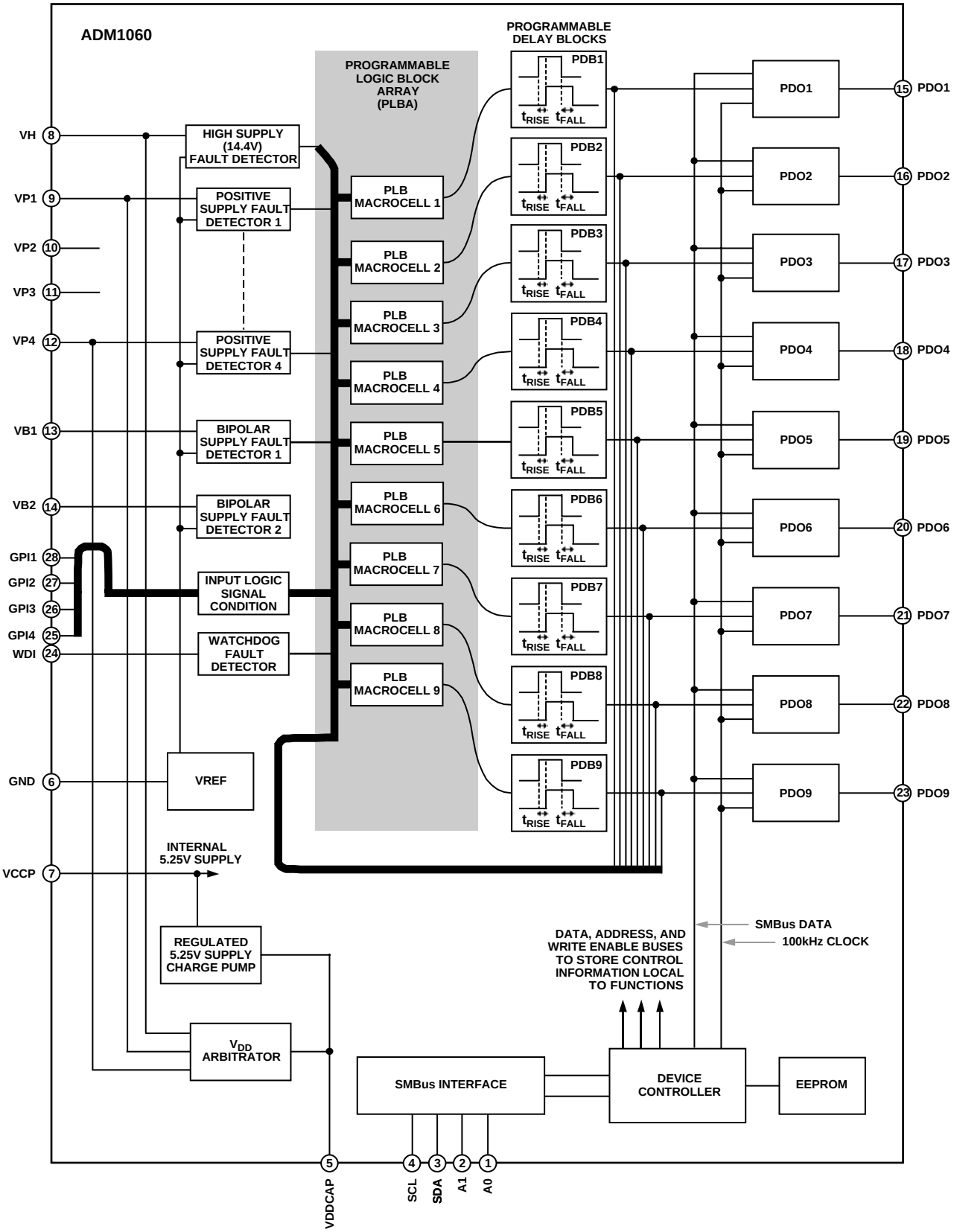

Figure 1. Functional Block Diagram

SPECIFICATIONS

(VH = 4.75 V to 14.4 V, VPn = 3.0 V to 6.0 V,[1] T_A = −40°C to +85°C, unless otherwise noted.)

Table 1.

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
POWER SUPPLY ARBITRATION					
VDDCAP	2.7			V	Any VPn ≥ 3.0 V
	2.7			V	VH ≥ 4.75 V
		4.75	5.1	V	Any VPn = 6.0 V
		4.75	5.1	V	VH = 14.4 V
POWER SUPPLY					
Supply Current, I_{DD}		3		mA	VDDCAP = 4.75 V, no PDO FET drivers on, no loaded PDO pull-ups to VDDCAP
			5	mA	VDDCAP = 4.75 V, all PDO FET drivers on (loaded with 1 µA), no PDO pull-ups to VDDCAP
Additional Current Available from VDDCAP[2]			1	mA	Max additional load that can be drawn from PDO pull-ups to VDDCAP
SUPPLY FAULT DETECTORS					
Input Impedance					
VH Input		52		kΩ	From VH to GND
VPn Inputs		52		kΩ	From VPn to GND
VBn Inputs		190		kΩ	From VBn to 2.25 V (internal reference)
		52		kΩ	From VBn to GND (positive mode)
		30		kΩ	From VBn to GND (negative mode)
Absolute Accuracy (VH, VPn, VBn Inputs)	−2.5		+2.5	%	
Calibrated Absolute Accuracy[3]					
VH, VPn Inputs	−1.0		+1.0	%	Factory preprogrammed to specific thresholds
VBn Inputs	−1.5		+1.5	%	Factory preprogrammed to specific thresholds
Glitch Filters (Digital)	0		100	µs	See Figure 19. Eight timeout options between 0 µs and 100 µs
PROGRAMMABLE DRIVER OUTPUTS					
High Voltage (Charge Pump) Mode (PDOs 1 to 4)					
Output Impedance, R_{OUT}		440		kΩ	
V_{OH}	11	12.5	14	V	I_{OH} = 0 µA
	10.5	12		V	I_{OH} = 1 µA
I_{OUTAVG}		20		µA	2 V < V_{OH} < 7 V
Standard (Digital Output) Mode (PDOs 1 to 9)					
V_{OH}	2.4			V	V_{PU} (pull-up to VDDCAP or VPn) > 2.7 V, I_{OH} = 1 mA
			4.5	V	V_{PU} to VPn = 6.0 V, I_{OH} = 0 mA
	VPU − 0.3			V	V_{PU} ≤ 2.7 V, I_{OH} = 1 mA
V_{OL}			0.4	V	I_{OL} = 2 mA
			1.2	V	I_{OL} = 10 mA
			2.0	V	I_{OL} = 15 mA
I_{SINK}[2]			20	mA	Total sink current (PDO1–PDO9)
R_{PULLUP}- Weak Pull-Up		20		kΩ	Internal pull-up
$I_{SOURCE\ (VPn)}$[2]			2	mA	Current load on any VPn pull-up (i.e., total source current available through any number of PDO pull-up switches configured on to any one)
Three-State Output Leakage Current			10	µA	V_{PDO} = 14.4 V

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
DIGITAL INPUTS (GPI 1–4, WDI, A0, A1)[4]					
Input High Voltage, V_{IH}	2.0			V	
Input Low Voltage, V_{IL}			0.8	V	
Input High Current, I_{IH}	−1			µA	V_{IN} = 5.5 V
Input Low Current, I_{IL}			1	µA	V_{IN} = 0 V
Input Capacitance		10		pF	
Programmable Pull-Down Current, $I_{PULLDOWN}$		10		µA	If known logic state required
SERIAL BUS DIGITAL INPUTS (SDA, SCL)					
Input High Voltage, V_{IH}	2.0			V	
Input Low Voltage, V_{IL}			0.8	V	
Output Low Voltage, V_{OL}			0.4	V	I_{OUT} = −3.0 mA
PROGRAMMABLE DELAY BLOCK					
Timeout	0		500	ms	16 programmable options on both rising and falling edge
WATCHDOG TIMER INPUT					
Timeout	0		12.8	s	Eight programmable timeout options
EEPROM RELIABILITY					
Endurance[5, 6]	100			Kcycles	
Data Retention[7]	10			Years	
SERIAL BUS TIMING[8]					
Clock Frequency, f_{SCLK}			400	kHz	See Figure 27
Glitch Immunity, t_{SW}			50	ns	See Figure 27
Bus Free Time, t_{BUF}	4.7			µs	See Figure 27
Start Setup Time, $t_{SU;STA}$	4.7			µs	See Figure 27
Start Hold Time, $t_{HD;STA}$	4			µs	See Figure 27
SCL Low Time, t_{LOW}	4.7			µs	See Figure 27
SCL High Time, t_{HIGH}	4			µs	See Figure 27
SCL, SDA Rise Time, t_r			1000	ns	See Figure 27
SCL, SDA Fall Time, t_f			300	µs	See Figure 27
Data Setup Time, $t_{SU;DAT}$	250			ns	See Figure 27
Data Hold Time, $t_{HD;DAT}$	300			ns	See Figure 27

NOTES
[1]At least one VPn must be ≥3.0 V if used as supply. VH must be ≥4.5 V if used as supply.
[2]Specification is not production tested, but is supported by characterization data at initial product release.
[3]1% threshold accuracy is only achievable on parts preprogrammed by Analog Devices. Contact ADM1060.program@analog.com for further details.
[4]Logic inputs will accept input high voltages up to 5.5 V even when the device is operating at supply voltages below 5 V.
[5]Endurance is qualified to 100,000 cycles as per JEDEC Std. 22 method A117, and measured at −40°C, +25°C, and +85°C.
[6]For programming and erasing of EEPROM, a minimum V_{DD} = 3.0 V is required 0°C to +85°C and a minimum V_{DD} = 4.5 V is required −40°C to 0°C.
[7]Retention lifetime equivalent at junction temperature (T_J) = 55°C as per JEDEC Std. 22 method A117.
[8]Timing specifications are tested at logic levels of V_{IL} = 0.8 V for a falling edge and V_{IH} = 2.0 V for a rising edge.

ABSOLUTE MAXIMUM RATINGS

Table 2. Absolute Maximum Ratings

Parameter	Rating
Voltage on VH Pin, PDO Pins	17 V
Voltage on VP Pins	7 V
Voltage on VB Pins	–7 V to +7 V
Voltage on Any Other Input	–0.3 V to +6.5 V
Input Current at Any Pin	±5 mA
Package Input Current	±20 mA
Maximum Junction Temperature (T_J 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 above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

THERMAL CHARACTERISTICS

28-Lead TSSOP Package:
θ_{JA} = 98°C/W

TYPICAL PERFORMANCE CHARACTERISTICS

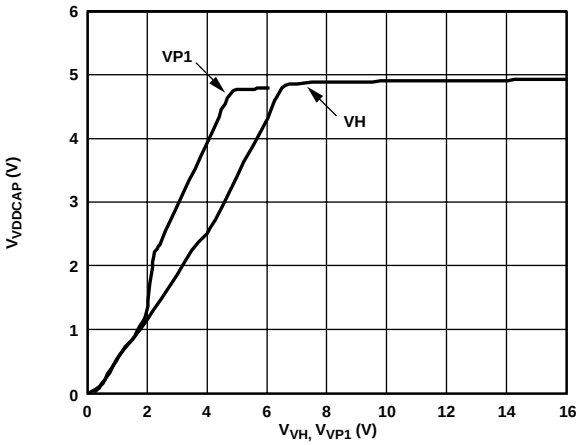

Figure 2. V_{VDDCAP} vs. V_{VH} and V_{VP1}

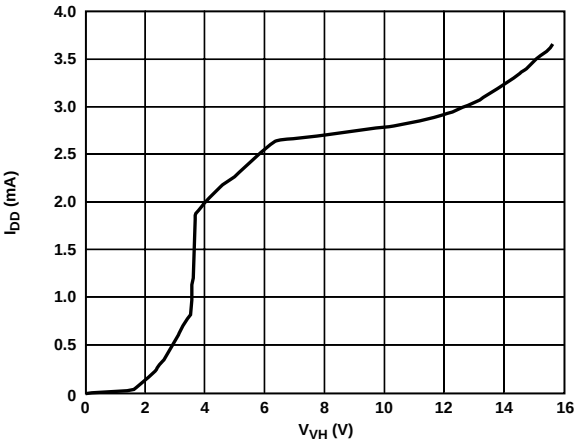

Figure 5. I_{DD} vs. V_{VH}

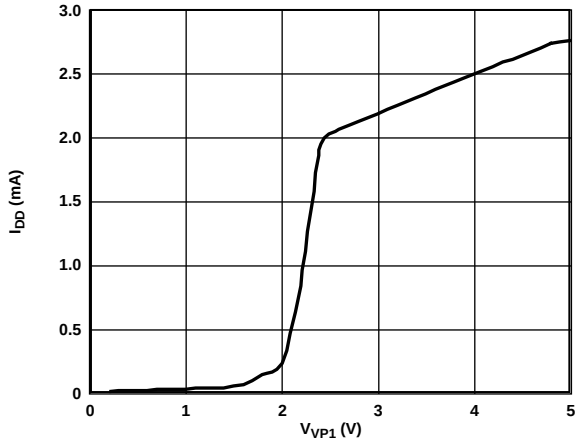

Figure 3. I_{DD} vs. V_{VP1} (Supply)

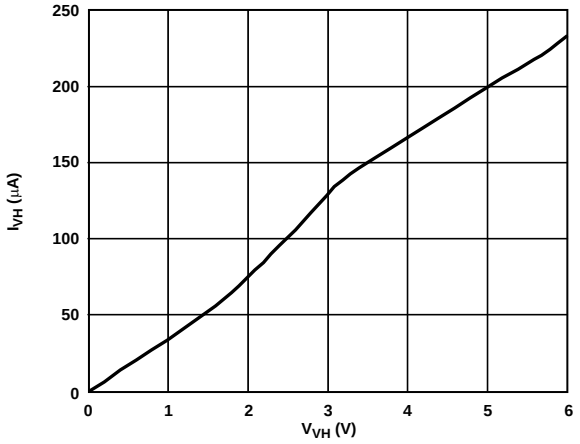

Figure 6. I_{VH} vs. V_{VH} (Not Supply)

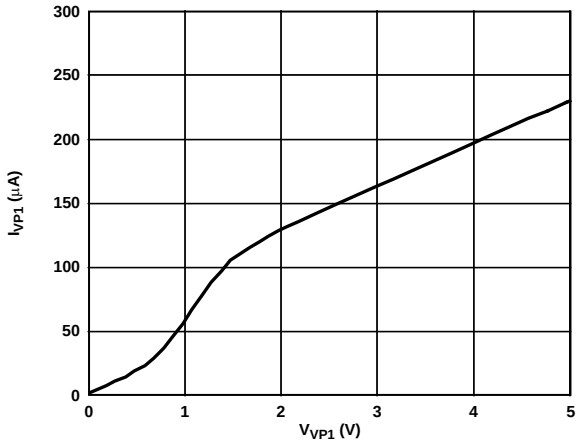

Figure 4. I_{VP1} vs. V_{VP1} (Not Supply)

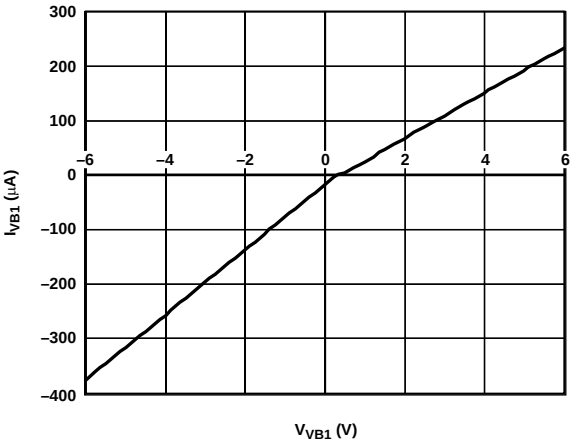

Figure 7. I_{VB1} vs. V_{VB1}

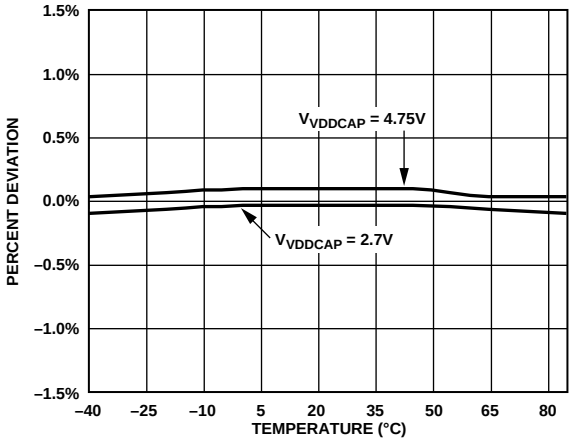

Figure 8. Percent Deviation in V_{THRESH} vs. Temperature

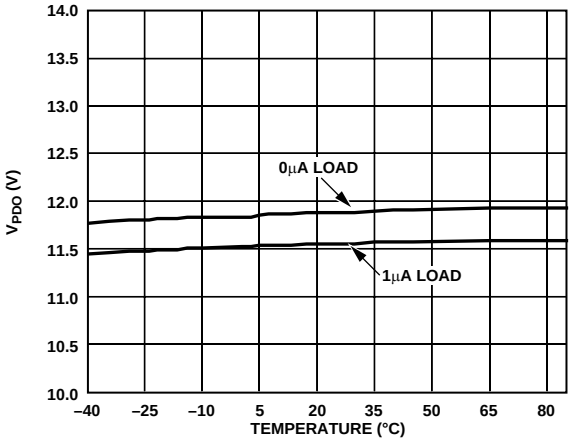

Figure 9. V_{PDO} (FET Drive Mode) vs. Temperature

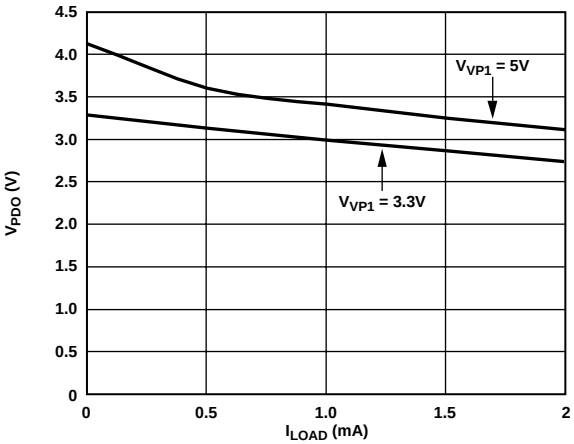

Figure 10. V_{PDO} (Strong Pull-Up to VP1) vs. Load Current

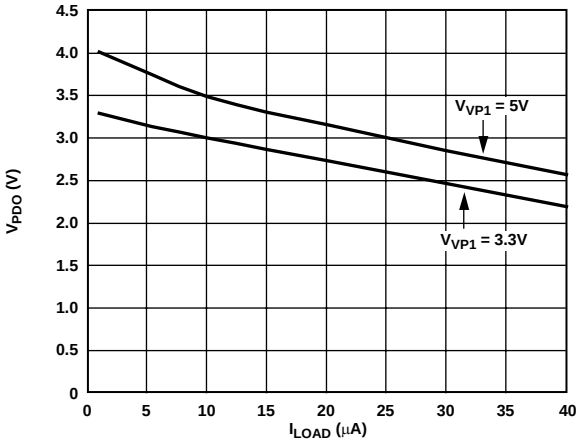

Figure 11. V_{PDO} (Weak Pull-Up to VP1) vs. Load Current

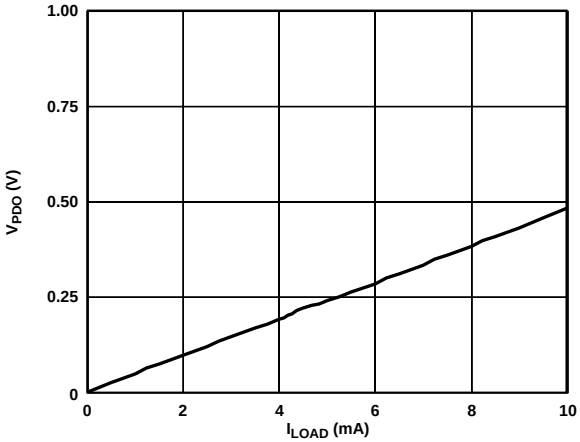

Figure 12. V_{PDO} (Strong Pull-Down) vs. Load Current

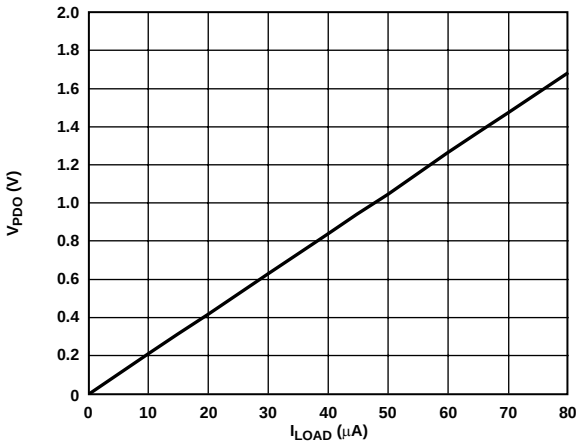

Figure 13. V_{PDO} (Weak Pull-Down) vs. Load Current

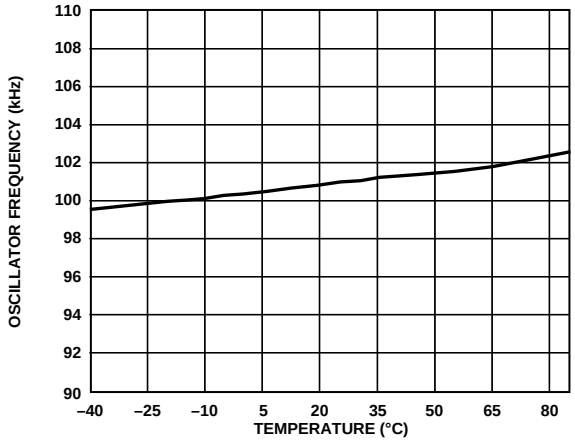

Figure 14. Oscillator Frequency vs. Temperature

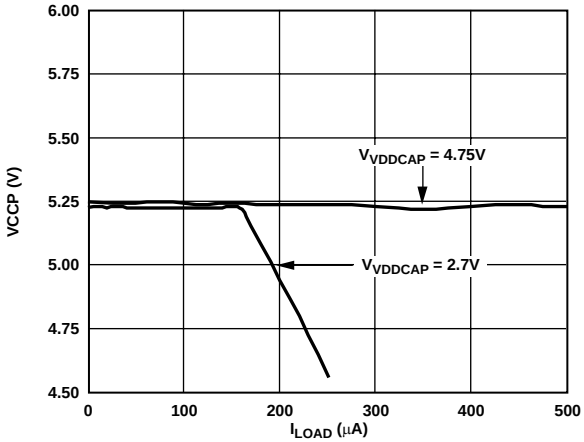

Figure 15. VCCP vs. Load Current

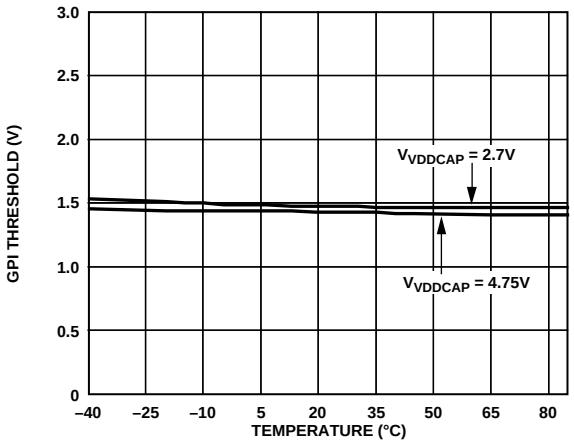

Figure 16. GPI Threshold vs. Temperature

INPUTS

POWERING THE ADM1060

The ADM1060 is powered from the highest voltage input on either the Positive Only supply inputs (VPn) or the High Voltage supply input (VH). The same pins are used for supply fault detection (discussed below). A V_{DD} arbitrator on the device chooses which supply to use. The arbitrator can be considered as diode OR'ing the positive supplies together (as shown in Figure 17).The diodes are supplemented with switches in a synchronous rectifier manner to minimize voltage loss. This loss can be reduced to ~0.2 V, resulting in the ability to power the ADM1060 from a supply as low as 3.0 V. Note that the supply on the VBn pins cannot be used to power the device, even if the input on these pins is positive. Also, the minimum supply of 3.0 V must appear on one of the VPn pins in order to correctly power up the ADM1060. A supply of no less than 4.5 V can be used on VH. This is because there is no synchronous rectifier circuit on the VH pin, resulting in a voltage drop of ~1.5 V across the diode of the V_{DD} arbitrator.

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 17. The capacitor has another use during "brown outs" (momentary loss of power). Under these conditions, where the input supply, VPn, 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 chip active until the next highest supply takes over the powering of the device. A 1 µF capacitor is recommended for this function. A minimum capacitor value of 0.1 µF is required.

Note that in the case where there are two or more supplies within 100 mV of each other, the supply that takes control of V_{DD} first will keep control. For example, if VP1 is connected to a 3.3 V supply, V_{DD} will power up to approximately 3.1 V through VP1. If VP2 is then connected to another 3.3 V supply, VP1 will still power the device, unless VP2 goes 100 mV higher than VP1.

A second capacitor is required on the VCCP pin of the ADM1060. This capacitor is the reservoir capacitor for the central charge pump. Again, a 1 µF capacitor is recommended for this function. A minimum capacitor value of 0.1 µF is required.

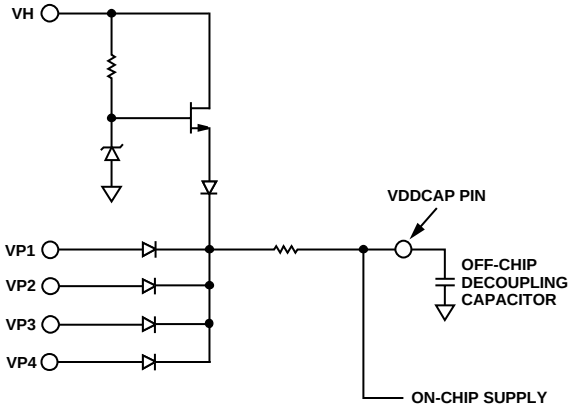

Figure 17. V_{DD} Arbitrator Operation

PROGRAMMABLE SUPPLY FAULT DETECTORS (SFDs)

The ADM1060 has seven programmable supply fault detectors (SFDs): one high voltage detector (+2 V to +14.4 V), two bipolar detectors (+1 V to +6 V, −2 V to −6 V) and four positive only voltage detectors (+0.6 V to +6 V). Inputs are applied to these detectors via the VH (high voltage supply input), VBn (bipolar supply input), and VPn (positive only input) pins, respectively. The SFDs detect a fault condition on any of these input supplies. A fault is defined as undervoltage (where the supply drops below a preprogrammed level), overvoltage (where the supply rises above a preprogrammed level), or out-of-window (where the supply deviates outside either the programmed overvoltage or undervoltage threshold). Only one fault type can be selected at a time.

An undervoltage (UV) fault is detected by comparing the input supply to a programmed reference (the undervoltage threshold). If the input voltage drops below the undervoltage threshold, the output of the comparator goes high, asserting a fault. The undervoltage threshold is programmed using an 8-bit DAC. On a given range, the UV threshold can be set with a resolution of

$$Step\ Size = Threshold\ Range/255$$

An overvoltage (OV) fault is detected in exactly the same way, using a second comparator and DAC to program the reference.

All thresholds are programmed using 8-bit registers, one register each for the seven UV thresholds and one each for the seven OV thresholds. The UV or OV threshold programmed by the user is given by

$$V_T = \frac{V_R \times N}{255} + V_B$$

where

Voltage Range	V_B (V)	V_R (V)
0.6 V to 1.8 V	0.604	1.204
1 V to 3 V	1.003	1.999
2 V to 6 V	2.005	3.997
4.8 V to 14.4 V	4.849	9.666
–2 V to –6 V	–1.994	–3.995

V_T is the desired threshold voltage (UV or OV)

V_R is the threshold voltage range

N is the decimal value of the 8 bit code

V_B is the bottom of threshold range

The code for a given threshold is therefore given by

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

For example, if the user wishes to set a 5 V OV threshold on VP1, the code to be programmed in the PS1OVTH register (discussed later) would be

$$N = 255 \times (5 - 2.005)/3.997$$

Thus, N = 191 (1011 1111 binary, or 0xBF)

The available threshold ranges and their resolutions are shown in Table 3. Note that the low end of the detection range is fixed at 33.33% of the top of the range. Note also that for a given SFD, the ranges overlap; for example, VH goes from 2 V to 6 V and then from 4.8 V to 14.4 V. This is to provide better threshold setting resolution as supplies decrease in value.

Table 3. Input Threshold Ranges and Resolution

Input Name	Voltage Ranges	Resolution
VH	4.8 V to 14.4 V	37.6 mV
	2 V to 6 V	15.6 mV
VBn	2 V to 6 V	15.6 mV (Pos. Mode)
	1 V to 3 V	7.8 mV (Pos. Mode)
	–6 V to –2 V	15.6 mV (Neg. Mode)
VPn	2 V to 6 V	15.6 mV
	1 V to 3 V	7.8 mV
	0.6 V to 1.8 V	4.7 mV

Figure 18 illustrates the function of the programmable SFD (for the case of a positive supply).

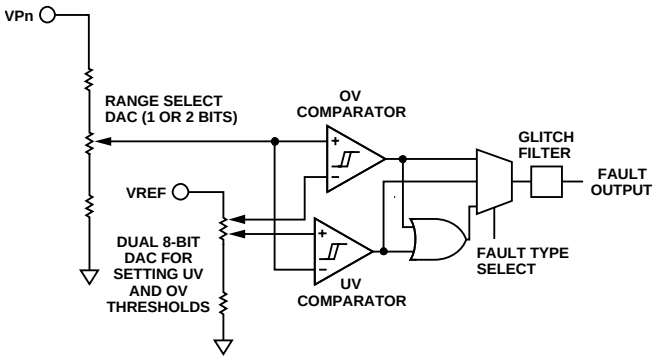

Figure 18. Positive Programmable Supply Fault Detector

SFD COMPARATOR HYSTERESIS

The OV and UV comparators shown in Figure 18 are always looking at VPn via a potential divider. In order to avoid chattering (multiple transitions when the input is very close to the set threshold level), these comparators have digitally programmable hysteresis. The UV and OV hysteresis can be programmed in two registers that are similar but separate to the UV or OV threshold registers. Only the five LSBs of these registers can be set. The hysteresis is added after the supply voltage goes out of tolerance. Thus, the user can determine how much above the UV threshold the input must rise again before a UV fault is deasserted. Similarly, the user can determine how much below the OV threshold the input must fall again before an OV fault is deasserted. The hysteresis figure is given by

$$V_H = V_R \times N_{THRESH}/255$$

where

V_H is the desired hysteresis voltage

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

Therefore, if the low range threshold detector was selected, the max hysteresis is defined as

$$(3\text{ V} - 1\text{ V}) \times 31/255 = 242\text{ mV, where } (2^5 - 1 = 31)$$

The hysteresis programming resolution is the same as the threshold detect ranges—that is, 37.5 mV on the high range, 15.6 mV on the midrange, 7.8 mV on the low range, and 4.7 mV on the ultralow range.

BIPOLAR SFDs

The two bipolar SFDs also allow the detection of faults on negative supplies. A polarity bit in the setup register for this SFD (Bit 7 in Register BSnSEL—see register map overleaf) determines if a positive or negative input should be applied to VBn. Only one range (–6 V to –2 V) is available when the SFDs are in negative mode. Note that the bipolar SFDs cannot be used to power the ADM1060, even if the voltage on VBn is positive.

SFD FAULT TYPES

Three types of faults can be asserted by the SFD: an OV fault, a UV fault, and an out-of-window fault (where the UV and OV faults are OR'ed together). The type of fault required is programmed using the fault type select bits (Bits 0, 1 in Register _SnSEL). If an application requires separate fault conditions to be detected on one supply (e.g., assert PDO1 if a UV fault occurs on a 3.3 V supply, assert PDO9 if an OV fault occurs on the same 3.3 V supply), that supply will need to be applied to more than one input pin.

GLITCH FILTERING ON THE SFDs

The final stage of the SFD is a glitch filter. This block provides time domain filtering on the output of the SFD. This allows the user to remove any spurious transitions (such as supply bounce at turn-on). This deglitching function is in addition to the programmable hysteresis of the SFDs. The glitch filter timeout is programmable up to 100 µs. If a pulse shorter than the programmed timeout appears on the input, this pulse is masked and the signal change will appear on the output. If an input pulse longer than the programmed timeout appears on the input, this pulse will appear on the output. The output will be delayed (with respect to the input) by the length of the programmed timeout.

Figure 19 shows the implementation of glitch filtering.

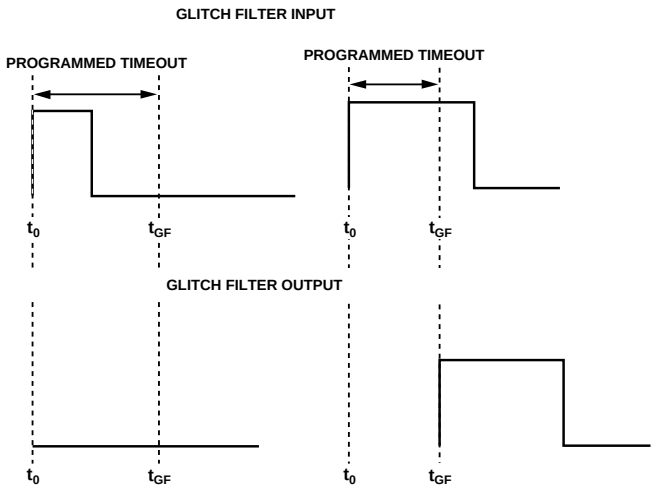

Figure 19. Glitch Filtering on the SFDs

PROGRAMMING THE SFDs ON THE SMBus

The details of using the SMBus are described later, but the register names associated with the supply fault detector blocks, the bit map of those registers, and the function of each of the bits is described in the following tables. The tables show how to set up UV threshold, UV hysteresis, OV threshold, OV hysteresis, glitch filtering, and fault type for each of the SFDs on the ADM1060.

SFD REGISTER NAMES

Table 4. List of Registers for the Supply Fault Detectors

Hex Address	Table	Name	Default Power-On Value	Description
A0	Table 5	BS1OVTH	0xFF	Overvoltage Threshold for Bipolar Voltage SFD1 (BS1SFD)
A1	Table 6	BS1OVHYST	0x00	Digital Hysteresis on OV Threshold for BS1SFD
A2	Table 7	BS1UVTH	0x00	Undervoltage Threshold for BS1SFD
A3	Table 8	BS1UVHYST	0x00	Digital Hysteresis on UV Threshold for BS1SFD
A4	Table 9	BS1SEL	0x00	Glitch Filter, Range, and Fault Type Select for BS1SFD
A8	Table 5	BS2OVTH	0xFF	Overvoltage Threshold for Bipolar Voltage SFD2 (BS2SFD)
A9	Table 6	BS2OVHYST	0x00	Digital Hysteresis on OV Threshold for BS2SFD
AA	Table 7	BS2UVTH	0x00	Undervoltage Threshold for BS2SFD
AB	Table 8	BS2UVHYST	0x00	Digital Hysteresis on UV Threshold for BS2SFD
AC	Table 9	BS2SEL	0x00	Glitch Filter, Range, and Fault Type Select for BS2SFD
B0	Table 10	HSOVTH	0xFF	Overvoltage Threshold for High Voltage SFD (HVSFD)
B1	Table 11	HSOVHYST	0x00	Digital Hysteresis on OV Threshold for HVSFD
B2	Table 12	HSUVTH	0x00	Undervoltage Threshold for HVSFD
B3	Table 13	HSUVHYST	0x00	Digital Hysteresis on UV Threshold for HVSFD
B4	Table 14	HSSEL	0x00	Glitch Filter, Range, and Fault Type Select for HVSFD
B8	Table 15	PS1OVTH	0xFF	Overvoltage Threshold for Positive Voltage SFD1 (PS1SFD)
B9	Table 16	PS1OVHYST	0x00	Digital Hysteresis on OV Threshold for PS1SFD
BA	Table 17	PS1UVTH	0x00	Undervoltage Threshold for PS1SFD
BB	Table 18	PS1UVHYST	0x00	Digital Hysteresis on UV Threshold for PS1SFD
BC	Table 19	PS1SEL	0x00	Glitch Filter, Range, and Fault Type Select for PS1SFD
C0	Table 15	PS2OVTH	0xFF	Overvoltage Threshold for Positive Voltage SFD2 (PS2SFD)
C1	Table 16	PS2OVHYST	0x00	Digital Hysteresis on OV Threshold for PS2SFD
C2	Table 17	PS2UVTH	0x00	Undervoltage Threshold for PS2SFD
C3	Table 18	PS2UVHYST	0x00	Digital Hysteresis on UV Threshold for PS2SFD
C4	Table 19	PS2SEL	0x00	Glitch Filter, Range, and Fault Type Select for PS2SFD
C8	Table 15	PS3OVTH	0xFF	Overvoltage Threshold for Positive Voltage SFD3 (PS3SFD)
C9	Table 16	PS3OVHYST	0x00	Digital Hysteresis on OV Threshold for PS3SFD
CA	Table 17	PS3UVTH	0x00	Undervoltage Threshold for PS3SFD
CB	Table 18	PS3UVHYST	0x00	Digital Hysteresis on UV Threshold for PS3SFD
CC	Table 19	PS3SEL	0x00	Glitch Filter, Range, and Fault Type Select for PS3SFD
D0	Table 15	PS4OVTH	0xFF	Overvoltage Threshold for Positive Voltage SFD4 (PS4SFD)
D1	Table 16	PS4OVHYST	0x00	Digital Hysteresis on OV Threshold for PS4SFD
D2	Table 17	PS4UVTH	0x00	Undervoltage Threshold for PS4SFD
D3	Table 18	PS4UVHYST	0x00	Digital Hysteresis on UV Threshold for PS4SFD
D4	Table 19	PS4SEL	0x00	Glitch Filter, Range, and Fault Type Select for PS4SFD

SFD Register Bit Maps

BIPOLAR SUPPLY FAIL DETECT (BSn SFD) REGISTERS

Table 5. Register 0xA0, 0xA8 BSnOVTH (Power-On Default 0xFF)

Bit	Name	R/W	Description
7–0	OV7–OV0	R/W	8-Bit Digital Value for OV Threshold on BSn SFD

Table 6. Register 0xA1, 0xA9 BSnOVHYST (Power-On Default 0x00)

Bit	Name	R/W	Description
7–5	Reserved	N/A	Cannot Be Used
4–0	HY4–HY0	R/W	5-Bit Digital Value for Hysteresis on OV Threshold of BSn SFD

Table 7. Register 0xA2, 0xAA BSnUVTH (Power-On Default 0x00)

Bit	Name	R/W	Description
7–0	UV7–UV0	R/W	8-Bit Digital Value for UV Threshold on BSn SFD

Table 8. Register 0xA3, 0xAB BSnUVHYST (Power-On Default 0x00)

Bit	Name	R/W	Description
7–5	Reserved	N/A	Cannot Be Used
4–0	HY4–HY0	R/W	5-Bit Digital Value for Hysteresis on UV Threshold of BSn SFD

Table 9. Register 0xA4, 0xAC BSnSEL (Power-On Default 0x00)

Bit	Name	R/W	Description			
7	POL	R/W	Polarity of Bipolar SFDn	**POL**	**Sign of Detection Range**	
				0	Positive	
				1	Negative	
6–4	GF2–GF0	R/W	**GF2**	**GF1**	**GF0**	**Glitch Filter Delay (µs)**
			0	0	0	0
			0	0	1	5
			0	1	0	10
			0	1	1	20
			1	0	0	30
			1	0	1	50
			1	1	0	75
			1	1	1	100
3	Reserved	N/A	Cannot Be Used			
2	RSEL	R/W	**Note:** When POL is set to 1 (SFD is in negative mode), RSEL is unused since there is only one range in this mode.			
			RSEL1	**Bottom of Range**	**Top of Range**	**Step Size (mV)**
			0	1 V	3 V	7.8
			1	2 V	6 V	15.6
1–0	FS1–FS0	R/W	**FS1**	**FS0**	**Fault Select Type**	
			0	0	Overvoltage	
			0	1	Undervoltage	
			1	0	Out-of-Window	
			1	1	Not Allowed	

HIGH VOLTAGE SUPPLY FAULT DETECT (HV SFD) REGISTERS

Table 10. Register 0xB0 HSOVTH (Power-On Default 0xFF)

Bit	Name	R/W	Description
7–0	OV7–OV0	R/W	8-Bit Digital Value for OV Threshold on HV SFD

Table 11. Register 0xB1 HSOVHYST (Power-On Default 0x00)

Bit	Name	R/W	Description
7–5	Reserved	N/A	Cannot Be Used
4–0	HY4–HY0	R/W	5-Bit Digital Value for Hysteresis on OV Threshold of HV SFD

Table 12. Register 0xB2 HSUVTH (Power-On Default 0x00)

Bit	Name	R/W	Description
7–0	UV7–UV0	R/W	8-Bit Digital Value for UV Threshold on HV SFD

Table 13. Register 0xB3 HSUVHYST (Power-On Default 0x00)

Bit	Name	R/W	Description
7–5	Reserved	N/A	Cannot Be Used
4–0	HY4–HY0	R/W	5-Bit Digital Value for Hysteresis on UV Threshold of HV SFD

Table 14. Register 0xB4 HSSEL (Power-On Default 0x00)

Bit	Name	R/W	Description			
7	Reserved	N/A	Cannot Be Used			
6–4	GF2–GF0	R/W	**GF2**	**GF1**	**GF0**	**Glitch Filter Delay (µs)**
			0	0	0	0
			0	0	1	5
			0	1	0	10
			0	1	1	20
			1	0	0	30
			1	0	1	50
			1	1	0	75
			1	1	1	100
3	Reserved	N/A	Cannot Be Used			
2	RSEL	W	**RSEL**	**Bottom of Range**	**Top of Range**	**Step Size (mV)**
			0	2 V	6 V	15.6
			1	4.8 V	14.4 V	37.6
1–0	FS1–FS0	W	**FS1**	**FS0**	**Fault Select Type**	
			0	0	Overvoltage	
			0	1	Undervoltage	
			1	0	Out-of-Window	
			1	1	Not Allowed	

POSITIVE VOLTAGE SUPPLY FAULT DETECT (PSn SFD) REGISTERS

Table 15. Register 0xB8, 0xC0, 0xC8, 0xD0 PSnOVTH (Power-On Default 0xFF)

Bit	Name	R/W	Description
7–0	OV7–OV0	R/W	8-Bit Digital Value for OV Threshold on PSn SFD.

Table 16. Register 0xB9, 0xC1, 0xC9, 0xD1 PSnOVHYST (Power-On Default 0x00)

Bit	Name	R/W	Description
7–5	Reserved	N/A	Cannot Be Used
4–0	HY4–HY0	R/W	5-Bit Digital Value for Hysteresis on OV Threshold of PSn SFD

Table 17. Register 0xBA, 0xC2, 0xCA, 0xD2 PSnUVTH (Power-On Default 0x00)

Bit	Name	W	Description
7–0	UV7–UV0	R/W	8-Bit Digital Value for UV Threshold on PSn SFD

Table 18. Register 0xBB, 0xC3, 0xCB, 0xD3 PSnUVHYST (Power-On Default 0x00)

Bit	Name	W	Description
7–5	Reserved	N/A	Cannot Be Used
4–0	HY4–HY0	R/W	5-Bit Digital Value for Hysteresis on UV Threshold of PSn SFD

Table 19. Register 0xBC, 0xC4, 0xCC, 0xD4 PSnSEL (Power-On Default 0x00)

Bit	Name	R/W	Description				
7	Reserved	N/A	Cannot Be Used				
6–4	GF2–GF0	R/W	**GF2**	**GF1**	**GF0**	**Glitch Filter Delay (µs)**	
			0	0	0	0	
			0	0	1	5	
			0	1	0	10	
			0	1	1	20	
			1	0	0	30	
			1	0	1	50	
			1	1	0	75	
			1	1	1	100	
3–2	RSEL1–RESL0	R/W	**RSEL1**	**RSEL0**	**Bottom of Range**	**Top of Range**	**Step Size (mV)**
			0	0	2 V	6 V	15.6
			0	1	1 V	3 V	7.8
			1	X	0.6 V	1.8 V	4.7
1–0	FS1–FS0	R/W	**FS1**	**FS0**	**Fault Select Type**		
			0	0	Overvoltage		
			0	1	Undervoltage		
			1	0	Out-of-Window		
			1	1	Not Allowed		

WATCHDOG FAULT DETECTOR

The ADM1060 has a watchdog fault detector. This can be used to monitor a processor clock to ensure normal operation. The detector monitors the WDI pin, expecting a low-to-high or high-to-low transition within a preprogrammed period. The watchdog timeout period can be programmed from 200 ms to a maximum of 12.8 sec.

If no transition is detected, two signals are asserted. One is a latched high signal, indicating a fault has occurred. The other signal is a low-high-low pulse that can be used as a RESET signal for a processor core. The width of this pulse can be programmed from 10 µs to a maximum of 10 ms. These two watchdog signals can be selected as inputs to each of the PLBs (see the PLBA section). They can also be inverted, if required; for example, if a high-low-high pulse were required by a processor to reset. Thus, a fault on the watchdog can be used to generate a pulsed or latched output on any or all of the nine PDOs.

The latched signal can be cleared low by reading LATF1, then LATF2 across the SMBus interface (see the Fault Registers section). The RAM register list and the bit map for the watchdog fault detector are shown below.

Table 20. Watchdog Fault Detector Registers

Hex Address	Table	Name	Default Power-On Value	Description
9C	Table 21	WDCFG	0x00	Program Length Watchdog Timeout and Length of Pulsed Output

Table 21. WDCFG Register 0x9C (Power-On Default 0x00)

Bit	Name	R/W	Description				
7–5	Reserved	R/W	Unused				
4–3	PULS1–PULS0	R/W	Length of Pulse Output once the Watchdog Detector has Timed Out	**PULS1**	**PULS0**	**Pulse Length Selected (µs)**	
				0	0	10	
				0	1	100	
				1	0	1,000	
				1	1	10,000	
2–0	PER2–PER0	R/W	Watchdog Timeout Period	**PER2**	**PER1**	**PER0**	**Watchdog Timeout Selected (ms)**
				0	0	0	Disabled
				0	0	1	200
				0	1	0	400
				0	1	1	800
				1	0	0	1,600
				1	0	1	3,200
				1	1	0	6,400
				1	1	1	12,800

GENERAL-PURPOSE INPUTS (GPIs)

The ADM1060 has four general-purpose logic inputs (GPIs). These are TTL/CMOS logic level compatible. Standard logic signals can be applied to the pins: RESET from reset generators, PWRGOOD signals, fault flags, manual resets, and so on. These signals can be gated with the other inputs supervised by the ADM1060 and used to control the status of the PDOs. The inputs can be simply buffered, or a logic transition can be detected and a pulse output generated. The width of this pulse is programmable from 10 µs to a maximum of 10 ms. The configuration of the GPIs is shown in the register and bit maps below.

The GPIs also feature a glitch filter similar to that provided on the SFDs. This enables the user to ignore spurious transitions on the GPIs. For example, the glitch filter can be used to debounce a manual reset switch. The length of the glitch filter can also be programmed.

LOGIC STATE OF THE GPIs AND OTHER LOGIC INPUTS

Each of the GPIs can have a weak (10 µA) pull-down current source. The current sources can be connected to the inputs by progamming the relevant bit in the PDEN register. This enables the user to control the condition of these inputs, pulling them to GND even when they are unused or left floating.

Note that the same pull-down function is provided for the SMBus address pins, A0 and A1, and for the WDI pin. A register is used to program which of the inputs is connected to the current sources.

Table 22. General-Purpose Inputs (GPIn) Registers

Hex Address	Name	Default Power-On Value	Description
98	GPI4CFG	0x00	GPI4 configuration setup of the glitch filter delay, pulse width, level/edge detection, etc.
99	GPI3CFG	0x00	GPI3 configuration setup of the glitch filter delay, pulse width, level/edge detection, etc.
9A	GPI2CFG	0x00	GPI2 configuration setup of the glitch filter delay, pulse width, level/edge detection, etc.
9B	GPI1CFG	0x00	GPI1 configuration setup of the glitch filter delay, pulse width, level/edge detection, etc.

Table 23. GPInCFG Registers Bit Map (Power-On Default 0x00)

Bit	Name	R/W	Description				
7	Reserved	N/A	Cannot Be Used				
6	INVIN	R/W	If High, Invert Input				
5	INTYP	R/W	Determines whether a Level or an Edge is Detected on the Pin. If an edge is detected, a positive pulse of programmable length is output.	**INTYP**	**Detect**		
				0	Detect Level		
				1	Detect Edge		
4–3	PULS1–0	R/W	Length of Pulse Output Once an Edge Has Been Detected on Input	**PULS1**	**PULS0**	**Pulse Length Selected (µs)**	
				0	0	10	
				0	1	100	
				1	0	1,000	
				1	1	10,000	
2–0	GF2–GF0	R/W	Length of Time for which the Input Is Ignored	**GF2**	**GF1**	**GF0**	**Glitch Filter Delay (µs)**
				0	0	0	0
				0	0	1	5
				0	1	0	10
				0	1	1	20
				1	0	0	30
				1	0	1	50
				1	1	0	75
				1	1	1	100

Table 24. Registers for the Pull-Down Current Sources on Logic Inputs

Hex Address	Name	Default Power On Value	Description
91	PDEN	0x00	Setup of the Pull-Down Current Sources on All Logic Inputs. Pulls the selected input to GND.

Table 25. PDEN Register 0x91 Bit Map (Power-On Default 0x00)

Bit	Name	R/W	Description
7	Reserved	N/A	Cannot Be Used
6	PDENA1	R/W	If high, address pin A1 is pulled to GND using a 10 µA pull-down current source.
5	PDENA0	R/W	If high, address pin A0 is pulled to GND using a 10 µA pull-down current source.
4	PDENWDI	R/W	If high, WDI is pulled to GND using a 10 µA pull-down current source.
3	PDENGPI4	R/W	If high, GPI4 is pulled to GND using a 10 µA pull-down current source.
2	PDENGPI3	R/W	If high, GPI3 is pulled to GND using a 10 µA pull-down current source.
1	PDENGPI2	R/W	If high, GPI2 is pulled to GND using a 10 µA pull-down current source.
0	PDENGPI1	R/W	If high, GPI1 is pulled to GND using a 10 µA pull-down current source.

PROGRAMMING

PROGRAMMABLE LOGIC BLOCK ARRAY

The ADM1060 contains a programmable logic block array (PLBA). This block is the logical core of the device. The PLBA (and the PDBs—see the Programmable Delay Block section) provides the sequencing function of the ADM1060. The assertion of the nine programmable driver outputs (PDO) is controlled by the PLBA. The PLBA is comprised of nine macrocells, one per PDO channel. The main components of the macrocells are two wide AND-OR gates, as shown in Figure 20. Each AND gate represents a function (A or B) that can be used independently to control the assertion of the PDO pin. There are 21 inputs to each of these AND gates:

- The logic outputs of all seven supply fault detectors
- The four GPI logic inputs
- The watchdog fault detector (latched and pulsed)
- The delayed output of any of the other macrocells (the output of a macrocell cannot be an input to itself, since this would result in a nonterminating loop).

All 21 inputs are hardwired to both function A and function B AND gates. The user can then select which of these inputs controls the output. This is done using two control signals, IMK (a masking bit, setting it ignores the relevant input) and POL (a polarity bit, setting it inverts the input before it is applied to the AND gate). The effect of setting these bits can be seen in Figure 20. The inverting gate shown is an XOR gate, resulting in the following truth table:

Table 26. Truth Table for PLB Input Inversion

POL	Input Signal	XOR Output
0	0	0
0	1	1
1	0	1
1	1	0

The last two entries in the truth table show that with the INVERT (POL) bit set, the XOR output is always the inverse of the input.

Similarly, the ignore gate shown is an OR gate, resulting in the following truth table:

Table 27. Truth Table for PLB Input Masking

IMK	Input Signal	OR Output
0	0	0
0	1	1
1	0	1
1	1	1

It can be seen here that once the IMK bit is set, the OR output is always 1, regardless of the input, thus ignoring it. Figure 21 is a detailed diagram of the 21 inputs and the registers required to program them. Those shown are just for function A of PLB1, but function B and all of the functions in the other eight PLBs are programmed exactly the same way. An enable register allows the user to use function A, function B, or both. The output of functions A and/or B is input to a programmable delay block (PDB) where a delay can be programmed on both the rising and falling edges of an input (see the Programmable Delay Block section). The output of this PDB block can be progammed to invert before any of the PDO pins is asserted.

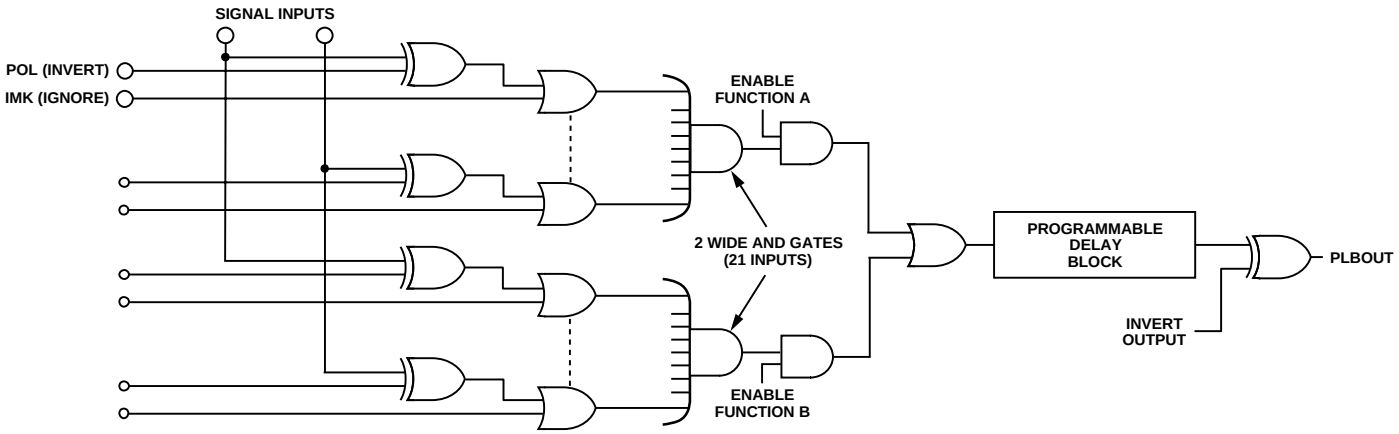

Figure 20. Simplified Programmable Logic Block Macrocell Schematic

LOGIC

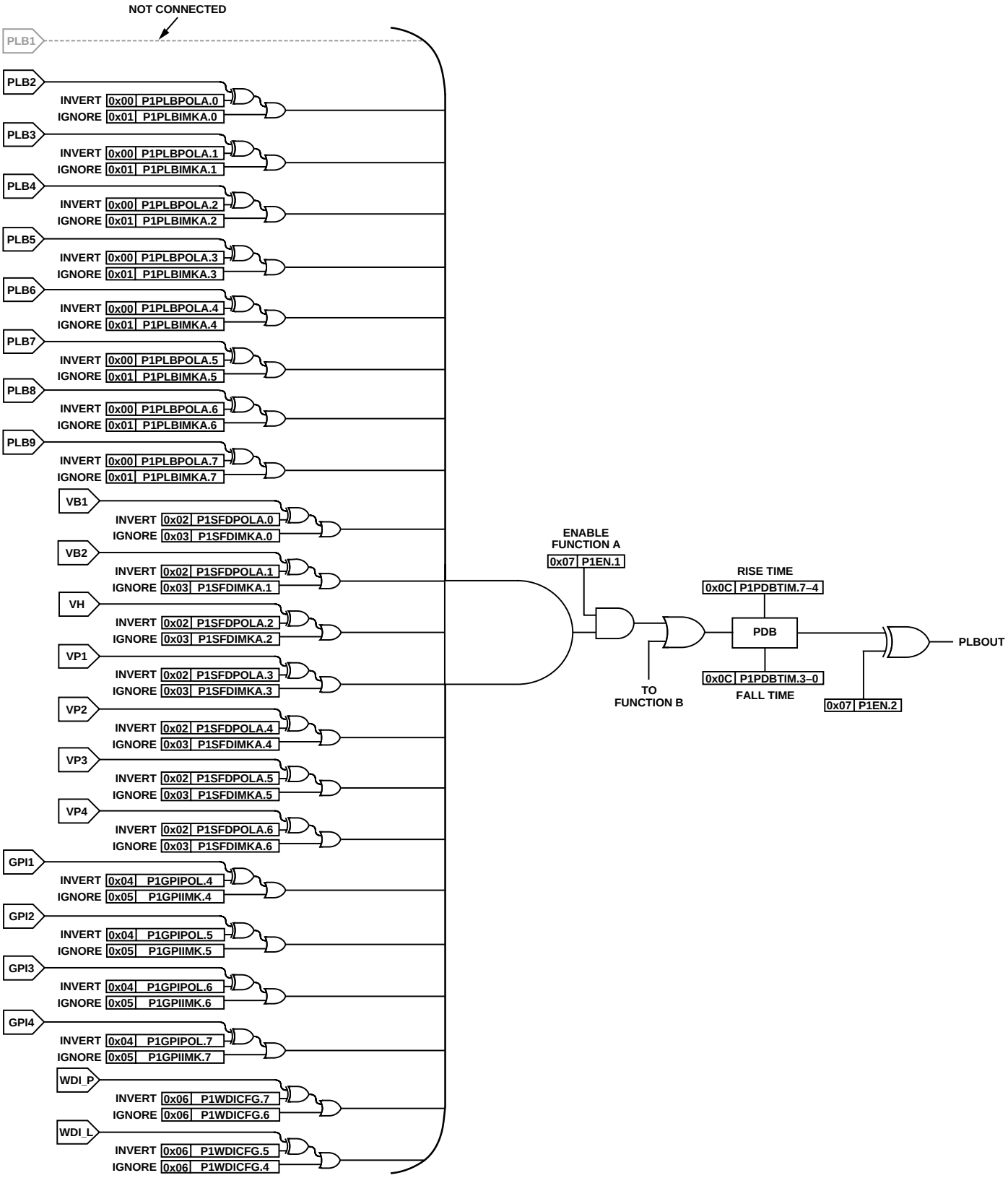

Figure 21. Detailed Diagram for Function A of PLB1

The control bits for these macrocells are stored locally in latches that are loaded at power-up. These latches can also be updated via the serial interface. The registers containing the macrocell control bits and the function of each bit are defined in the tables that follow.

Figure 21 highlights all 21 inputs to a given function and the register/bits that need to be set in order to condition the 21 inputs correctly. The diagram only shows function A of Programmable Logic Block 1 (PLB1), but all functions are programmed in the same way.

For example, if the user wishes to assert PLBOUT 200 ms after all of the supplies are in spec (PLBOUT may be used to drive the enable pin of an LDO), the supply fault detectors VBn, VH, and VPn are required to control the function. The function is programmed as follows:

1. The IGNORE bit of all the other inputs (GPIs, PDBs, WDI) in the relevant P1xxxIMK registers is set to 1. Thus, regardless of its status, the input to the function AND gate for these inputs will be 1.
2. Since the SFDs assert a 1 under a fault condition and a 0 when the supplies are in tolerance, the SFD outputs need to be inverted before being applied to the function. Thus the relevant bit in the P1SFDPOL register is set (see Table 38).
3. The function is enabled (Bit 1 of Register P1EN—see Table 36).
4. A rise time of 200 ms is programmed (register P1PDBTIM—see register map for details).

Table 28. Programmable Logic Block Array (PLBA) Registers

Hex Address	Table	Name	Default Power-On Value	Description
00	Table 29	P1PLBPOLA	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the A function of PLB1
01	Table 30	P1PLBIMKA	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the A function of PLB1
02	Table 31	P1SFDPOLA	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB1
03	Table 32	P1SFDIMKA	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB1
04	Table 33	P1GPIPOL	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the A function of PLB1
05	Table 34	P1GPIIMK	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the B function of PLB1
06	Table 35	P1WDICFG	0x00	Polarity sense and ignore mask bits for the pulsed and latched outputs of the watchdog detector when used as inputs to both A and B functions of PLB1
07	Table 36	PS1EN	0x00	Enable bits for A and B functions of PLB1, polarity bit for PLB1 output
08	Table 29	P1PLBPOLB	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the B function of PLB1
09	Table 30	P1PLBIMKB	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the B function of PLB1
0A	Table 31	P1SFDPOLB	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB1
0B	Table 32	P1SFDIMKB	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB1
10	Table 29	P2PLBPOLA	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the A function of PLB2
11	Table 30	P2PLBIMKA	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the A function of PLB2
12	Table 31	P2SFDPOLA	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB2
13	Table 32	P2SFDIMKA	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB2
14	Table 33	P2GPIPOL	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the A function of PLB2

Hex Address	Table	Name	Default Power-On Value	Description
15	Table 34	P2GPIIMK	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the B function of PLB2
16	Table 35	P2WDICFG	0x00	Polarity sense and ignore mask bits for the pulsed and latched outputs of the watchdog detector when used as inputs to both A and B functions of PLB2
17	Table 36	PS2EN	0x00	Enable bits for A and B functions of PLB2, polarity bit for PLB2 output
18	Table 29	P2PLBPOLB	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the B function of PLB2
19	Table 30	P2PLBIMKB	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the B function of PLB2
1A	Table 31	P2SFDPOLB	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB2
1B	Table 32	P2SFDIMKB	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB2
20	Table 29	P3PLBPOLA	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the A function of PLB3
21	Table 30	P3PLBIMKA	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the A function of PLB3
22	Table 31	P3SFDPOLA	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB3
23	Table 32	P3SFDIMKA	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB3
24	Table 33	P3GPIPOL	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the A function of PLB3
25	Table 34	P3GPIIMK	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the B function of PLB3
26	Table 35	P3WDICFG	0x00	Polarity sense and ignore mask bits for the pulsed and latched outputs of the watchdog detector when used as inputs to both A and B functions of PLB3
27	Table 36	PS3EN	0x00	Enable bits for A and B functions of PLB3, polarity bit for PLB3 output
28	Table 29	P3PLBPOLB	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the B function of PLB3
29	Table 30	P3PLBIMKB	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the B function of PLB3
2A	Table 31	P3SFDPOLB	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB3
2B	Table 32	P3SFDIMKB	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB3
30	Table 29	P4PLBPOLA	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the A function of PLB1
31	Table 30	P4PLBIMKA	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the A function of PLB1
32	Table 31	P4SFDPOLA	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB1
33	Table 32	P4SFDIMKA	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB1
34	Table 33	P4GPIPOL	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the A function of PLB1
35	Table 34	P4GPIIMK	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the B function of PLB1
36	Table 35	P4WDICFG	0x00	Polarity sense and ignore mask bits for the pulsed and latched outputs of the watchdog detector when used as inputs to both A and B functions of PLB4
37	Table 36	PS4EN	0x00	Enable bits for A and B functions of PLB4, polarity bit for PLB4 output
38	Table 29	P4PLBPOLB	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the B function of PLB4

Hex Address	Table	Name	Default Power-On Value	Description
39	Table 30	P4PLBIMKB	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the B function of PLB4
3A	Table 31	P4SFDPOLB	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB4
3B	Table 32	P4SFDIMKB	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB4
40	Table 29	P5PLBPOLA	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the A function of PLB5
41	Table 30	P5PLBIMKA	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the A function of PLB5
42	Table 31	P5SFDPOLA	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB5
43	Table 32	P5SFDIMKA	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB5
44	Table 33	P5GPIPOL	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the A function of PLB5
45	Table 34	P5GPIIMK	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the B function of PLB5
46	Table 35	P5WDICFG	0x00	Polarity sense and ignore mask bits for the pulsed and latched outputs of the watchdog detector when used as inputs to both A and B functions of PLB5
47	Table 36	PS5EN	0x00	Enable bits for A and B functions of PLB5, polarity bit for PLB5 output
48	Table 29	P5PLBPOLB	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the B function of PLB5
49	Table 30	P5PLBIMKB	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the B function of PLB5
4A	Table 31	P5SFDPOLB	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB5
4B	Table 32	P5SFDIMKB	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB5
50	Table 29	P6PLBPOLA	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the A function of PLB6
51	Table 30	P6PLBIMKA	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the A function of PLB6
52	Table 31	P6SFDPOLA	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB6
53	Table 32	P6SFDIMKA	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB6
54	Table 33	P6GPIPOL	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the A function of PLB6
55	Table 34	P6GPIIMK	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the B function of PLB6
56	Table 35	P6WDICFG	0x00	Polarity sense and ignore mask bits for the pulsed and latched outputs of the watchdog detector when used as inputs to both A and B functions of PLB6
57	Table 36	PS6EN	0x00	Enable bits for A and B functions of PLB6, polarity bit for PLB6 output
58	Table 29	P6PLBPOLB	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the B function of PLB6
59	Table 30	P6PLBIMKB	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the B function of PLB6
5A	Table 31	P6SFDPOLB	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB6
5B	Table 32	P6SFDIMKB	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB6
60	Table 29	P7PLBPOLA	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the A function of PLB7

Hex Address	Table	Name	Default Power-On Value	Description
61	Table 30	P7PLBIMKA	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the A function of PLB7
62	Table 31	P7SFDPOLA	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB7
63	Table 32	P7SFDIMKA	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB7
64	Table 33	P7GPIPOL	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the A function of PLB7
65	Table 34	P7GPIIMK	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the B function of PLB7
66	Table 35	P7WDICFG	0x00	Polarity sense and ignore mask bits for the pulsed and latched outputs of the watchdog detector when used as inputs to both A and B functions of PLB7
67	Table 36	PS7EN	0x00	Enable bits for A and B functions of PLB7, polarity bit for PLB7 output
68	Table 29	P7PLBPOLB	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the B function of PLB7
69	Table 30	P7PLBIMKB	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the B function of PLB7
6A	Table 31	P7SFDPOLB	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB7
6B	Table 32	P7SFDIMKB	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB7
70	Table 29	P8PLBPOLA	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the A function of PLB8
71	Table 30	P8PLBIMKA	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the A function of PLB8
72	Table 31	P8SFDPOLA	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB8
73	Table 32	P8SFDIMKA	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB8
74	Table 33	P8GPIPOL	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the A function of PLB8
75	Table 34	P8GPIIMK	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the B function of PLB8
76	Table 35	P8WDICFG	0x00	Polarity sense and ignore mask bits for the pulsed and latched outputs of the watchdog detector when used as inputs to both A and B functions of PLB8
77	Table 36	PS8EN	0x00	Enable bits for A and B functions of PLB8, polarity bit for PLB8 output
78	Table 29	P8PLBPOLB	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the B function of PLB8
79	Table 30	P8PLBIMKB	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the B function of PLB8
7A	Table 31	P8SFDPOLB	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB8
7B	Table 32	P8SFDIMKB	0x00	Ignore mask for all 7 SFD inputs (VH, two VBs, four VPs) to the B function of PLB8
80	Table 29	P9PLBPOLA	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the A function of PLB9
81	Table 30	P9PLBIMKA	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the A function of PLB9
82	Table 31	P9SFDPOLA	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB9
83	Table 32	P9SFDIMKA	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the A function of PLB9

Hex Address	Table	Name	Default Power-On Value	Description
84	Table 33	P9GPIPOL	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the A function of PLB9
85	Table 34	P9GPIIMK	0x00	Polarity sense and ignore mask bits for all four GPIs when used as inputs to the B function of PLB9
86	Table 35	P9WDICFG	0x00	Polarity sense and ignore mask bits for the pulsed and latched outputs of the watchdog detector when used as inputs to both A and B functions of PLB9
87	Table 36	PS9EN	0x00	Enable bits for A and B functions of PLB9, polarity bit for PLB9 output
88	Table 29	P9PLBPOLB	0x00	Polarity sense for all eight other PLB outputs when used as inputs to the B function of PLB9
89	Table 30	P9PLBIMKB	0x00	Ignore mask for all eight other PLB outputs when used as inputs to the B function of PLB9
8A	Table 31	P9SFDPOLB	0x00	Polarity sense for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB9
8B	Table 32	P9SFDIMKB	0x00	Ignore mask for all seven SFD inputs (VH, two VBs, four VPs) to the B function of PLB9

PLBA REGISTER BIT MAPS

Table 29. PnPLBPOLA/PnPLBPOLB Registers Bit Map (Power-On Default 0x00)

Bit	Name	R/W	Description								
7–0	POL9–POL1	R/W	If high, **invert** the **PLBn** input before it is used in function A or B.								
			PLB1	**PLB2**	**PLB3**	**PLB4**	**PLB5**	**PLB6**	**PLB7**	**PLB8**	**PLB9**
	Function A		**0x00**	**0x10**	**0x20**	**0x30**	**0x40**	**0x50**	**0x60**	**0x70**	**0x80**
	Function B		**0x08**	**0x18**	**0x28**	**0x38**	**0x48**	**0x58**	**0x68**	**0x78**	**0x88**
7			PLB9	PLB9	PLB9	PLB9	PLB9	PLB9	PLB9	PLB9	PLB8
6			PLB8	PLB8	PLB8	PLB8	PLB8	PLB8	PLB8	PLB7	PLB7
5			PLB7	PLB7	PLB7	PLB7	PLB7	PLB7	PLB6	PLB6	PLB6
4			PLB6	PLB6	PLB6	PLB6	PLB6	PLB5	PLB5	PLB5	PLB5
3			PLB5	PLB5	PLB5	PLB5	PLB4	PLB4	PLB4	PLB4	PLB4
2			PLB4	PLB4	PLB4	PLB3	PLB3	PLB3	PLB3	PLB3	PLB3
1			PLB3	PLB3	PLB2	PLB2	PLB2	PLB2	PLB2	PLB2	PLB2
0			PLB2	PLB1	PLB1	PLB1	PLB1	PLB1	PLB1	PLB1	PLB1

Table 30. PnPLBIMKA/PnPLBIMKB Registers Bit Map (Power-On Default 0x00)

Bit	Name	R/W	Description								
7–0	IGN9–IGN1	R/W	If high, **mask** the **PLBn** input before it is used in function A or B.								
			PLB1	**PLB2**	**PLB3**	**PLB4**	**PLB5**	**PLB6**	**PLB7**	**PLB8**	**PLB9**
	Function A		**0x01**	**0x11**	**0x21**	**0x31**	**0x41**	**0x51**	**0x61**	**0x71**	**0x81**
	Function B		**0x09**	**0x19**	**0x29**	**0x39**	**0x49**	**0x59**	**0x69**	**0x79**	**0x89**
7			PLB9	PLB9	PLB9	PLB9	PLB9	PLB9	PLB9	PLB9	PLB8
6			PLB8	PLB8	PLB8	PLB8	PLB8	PLB8	PLB8	PLB7	PLB7
5			PLB7	PLB7	PLB7	PLB7	PLB7	PLB7	PLB6	PLB6	PLB6
4			PLB6	PLB6	PLB6	PLB6	PLB6	PLB5	PLB5	PLB5	PLB5
3			PLB5	PLB5	PLB5	PLB5	PLB4	PLB4	PLB4	PLB4	PLB4
2			PLB4	PLB4	PLB4	PLB3	PLB3	PLB3	PLB3	PLB3	PLB3
1			PLB3	PLB3	PLB2	PLB2	PLB2	PLB2	PLB2	PLB2	PLB2
0			PLB2	PLB1	PLB1	PLB1	PLB1	PLB1	PLB1	PLB1	PLB1

Table 31. PnSFDPOLA/PnSFDPOLB Registers Bit Map (Power-On Default 0x00)

Bit	Name	R/W	Description								
7	Reserved	N/A	Cannot Be Used								
6–0	POL7–POL1	R/W	If high, **invert** the **SFDn** input before it is used in function A or B.								
			PLB1	**PLB2**	**PLB3**	**PLB4**	**PLB5**	**PLB6**	**PLB7**	**PLB8**	**PLB9**
	Function A		**0x02**	**0x12**	**0x22**	**0x32**	**0x42**	**0x52**	**0x62**	**0x72**	**0x82**
	Function B		**0x0A**	**0x1A**	**0x2A**	**0x3A**	**0x4A**	**0x5A**	**0x6A**	**0x7A**	**0x8A**
6			VP4	VP4	VP4	VP4	VP4	VP4	VP4	VP4	VP4
5			VP3	VP3	VP3	VP3	VP3	VP3	VP3	VP3	VP3
4			VP2	VP2	VP2	VP2	VP2	VP2	VP2	VP2	VP2
3			VP1	VP1	VP1	VP1	VP1	VP1	VP1	VP1	VP1
2			VH	VH	VH	VH	VH	VH	VH	VH	VH
1			VB2	VB2	VB2	VB2	VB2	VB2	VB2	VB2	VB2
0			VB1	VB1	VB1	VB1	VB1	VB1	VB1	VB1	VB1

Table 32. PnSFDIMKA/PnSFDIMKB Registers Bit Map (Power-On Default 0x00)

Bit	Name	R/W	Description								
7	Reserved	N/A	Cannot Be Used								
6–0	IGN7–IGN1	R/W	If high, **mask** the **SFDn** input before it is used in function A or B.								
			PLB1	**PLB2**	**PLB3**	**PLB4**	**PLB5**	**PLB6**	**PLB7**	**PLB8**	**PLB9**
	Function A		**0x03**	**0x13**	**0x23**	**0x33**	**0x43**	**0x53**	**0x63**	**0x73**	**0x83**
	Function B		**0x0B**	**0x1B**	**0x2B**	**0x3B**	**0x4B**	**0x5B**	**0x6B**	**0x7B**	**0x8B**
6			VP4	VP4	VP4	VP4	VP4	VP4	VP4	VP4	VP4
5			VP3	VP3	VP3	VP3	VP3	VP3	VP3	VP3	VP3
4			VP2	VP2	VP2	VP2	VP2	VP2	VP2	VP2	VP2
3			VP1	VP1	VP1	VP1	VP1	VP1	VP1	VP1	VP1
2			VH	VH	VH	VH	VH	VH	VH	VH	VH
1			VB2	VB2	VB2	VB2	VB2	VB2	VB2	VB2	VB2
0			VB1	VB1	VB1	VB1	VB1	VB1	VB1	VB1	VB1

Table 33. PnGPIPOL Registers Bit Map (Power-On Default 0x00)

Bit	Name	R/W	Description								
7–4	APOL4–APOL1	R/W	If high, **invert** the **GPIn** input before it is used in function A.								
3–0	BPOL4–BPOL1	R/W	If high, **invert** the **GPIn** input before it is used in function B.								
			PLB1	**PLB2**	**PLB3**	**PLB4**	**PLB5**	**PLB6**	**PLB7**	**PLB8**	**PLB9**
			0x04	**0x14**	**0x24**	**0x34**	**0x44**	**0x54**	**0x64**	**0x74**	**0x84**
7	**Function A**		GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1
6			GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2
5			GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3
4			GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4
3	**Function B**		GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1
2			GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2
1			GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3
0			GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4

Table 34. PnGPIIMK Registers Bit Map (Power-On Default 0x00)

Bit	Name	R/W	Description								
7–4	AIMK4–AIMK1	R/W	If **high**, mask the **GPIn** input before it is used in function A.								
3–0	BIMK4–BIMK1	R/W	If **high**, mask the **GPIn** input before it is used in function B.								
			PLB1	**PLB2**	**PLB3**	**PLB4**	**PLB5**	**PLB6**	**PLB7**	**PLB8**	**PLB9**
			0x05	**0x15**	**0x25**	**0x35**	**0x45**	**0x55**	**0x65**	**0x75**	**0x85**
7	**Function A**		GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1
6			GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2
5			GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3
4			GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4
3	**Function B**		GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1	GPI1
2			GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2	GPI2
1			GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3	GPI3
0			GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4	GPI4

Table 35. PnWDICFG Registers 0x06, 0x16, 0x26, 0x36, 0x46, 0x56, 0x66, 0x76, 0x86 (Power-On Default 0x00)

Bit	Name	R/W	Description
7	APOLP	R/W	If high, invert the pulsed WDI input before it is used in function A.
6	AIMKP	R/W	If high, mask the pulsed WDI input before it is used in function A.
5	APOLL	R/W	If high, invert the latched WDI input before it is used in function A.
4	AIMKL	R/W	If high, mask the latched WDI input before it is used in function A.
3	BPOLP	R/W	If high, invert the pulsed WDI input before it is used in function B.
2	BIMKP	R/W	If high, mask the pulsed WDI input before it is used in function B.
1	BPOLL	R/W	If high, invert the latched WDI input before it is used in function B.
0	BIMKL	R/W	If high, mask the latched WDI input before it is used in function B.

Table 36. PnEN Register 0x07, 0x17, 0x27, 0x37, 0x47, 0x57, 0x67, 0x77, 0x87 (Power-On Default 0x00)

Bit	Name	R/W	Description
7–3	Reserved	N/A	Cannot Be Used
2	INVOP	R/W	If high, invert the PLB output.
1	ENA	R/W	If high, enable function A.
0	ENB	R/W	If high, enable function B.

PROGRAMMABLE DELAY BLOCK

Each output of the PLBA is fed into a separate programmable delay block (PDB). The PDB enables the user to add a delay to the logic block output before it is applied to either a PDO or one of the other PLBs (the output of a PLB can be the input to any of the other PLBs, but not itself). The PDB operation is similar to that of the glitch filter (discussed in the SFD section). There is an important difference between the two functions, however. The delay on the falling edge of an input to the PDB can be programmed independently of the rising edge. This allows the user to program the length of the pulse output from the PDB. Thus, for instance, the width of the pulse from the watchdog fault detector can be adjusted, or the user can ensure that a supply supervised by one of the SFDs is within its UV/OV range for a programmed period of time before asserting a PDO. A delay of 0 ms to 500 ms can be programmed in the PnPDBTIM registers. Four bits each are used to program the rising edge and falling edge.

Once programmed, the PDB operates as follows. If the user programs a delay on the rising edge of, say, 200 ms, the PDB looks for a rising edge on the input. Once it sees the edge it starts a timer. If the input remains high and the timer reaches 200 ms, the PDB immediately outputs a rising edge. If the input falls low before the timer has reached 200 ms, no edge is output from the PDB and the timer is reset. Because there is separate control over the falling edge, if no delay is programmed on the falling edge, the delay defaults to 0 ms and a falling edge on the input will immediately appear on the output. If a falling edge delay is programmed, the PDB operates exactly the opposite as it does for a rising edge. Again, if a delay of, say, 200 ms is programmed on the falling edge, the PDB looks for a falling edge on the input. Once it sees the edge, it starts a timer. If the input remains low and the timer reaches 200 ms, the output transitions from high to low. A valid rising edge must appear at the output before a falling edge delay can be activated. The function of the PDB is illustrated in Figure 22.

Aside from the extra timing flexibility, the programmable delay also provides a crude form of filtering. In much the same way as the glitch filter operates, an input must be high (or low) for a programmed period of time before being seen on the output. Transients that are shorter than the programmed timeouts will not appear on the output. The bit map for the register that controls both the rising and falling edges is shown in Table 38.

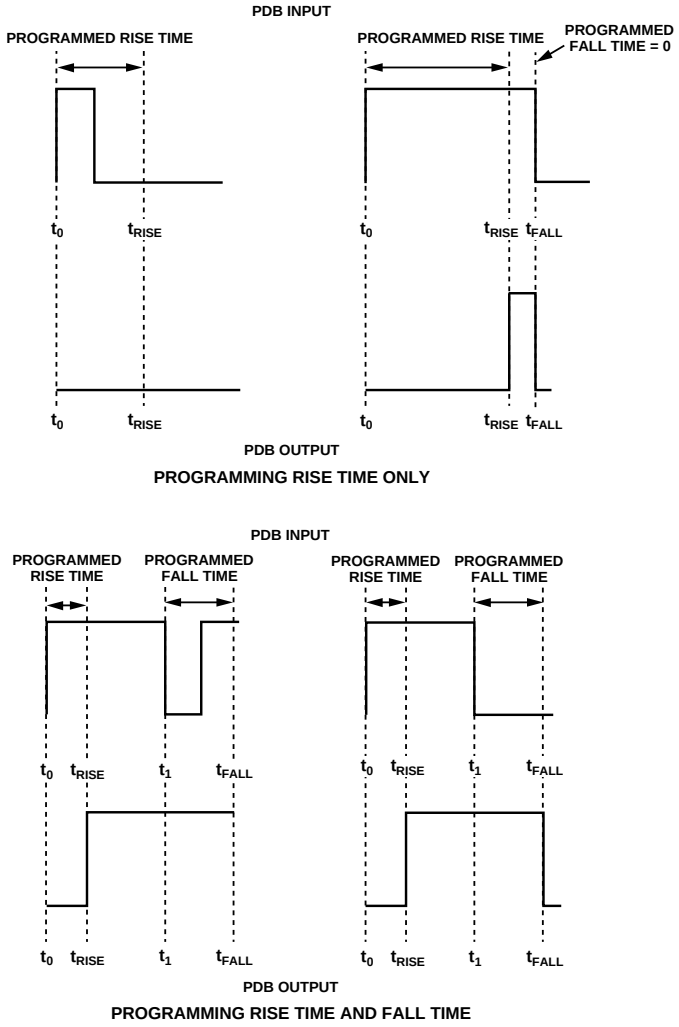

Figure 22. Programmable Delay Block (PDB) Functionality

Table 37. Programmable Delay Block (PDB) Registers

Hex Addr.	Table	Name	Default Power-On Value	Description
0C	Table 38	P1PDBTIM	0x00	Delay for PDB1. Delay for rising edge and falling edge programmed separately.
1C	Table 38	P2PDBTIM	0x00	Delay for PDB2. Delay for rising edge and falling edge programmed separately.
2C	Table 38	P3PDBTIM	0x00	Delay for PDB3. Delay for rising edge and falling edge programmed separately.
3C	Table 38	P4PDBTIM	0x00	Delay for PDB4. Delay for rising edge and falling edge programmed separately.
4C	Table 38	P5PDBTIM	0x00	Delay for PDB5. Delay for rising edge and falling edge programmed separately.
5C	Table 38	P6PDBTIM	0x00	Delay for PDB6. Delay for rising edge and falling edge programmed separately.
6C	Table 38	P7PDBTIM	0x00	Delay for PDB7. Delay for rising edge and falling edge programmed separately.
7C	Table 38	P8PDBTIM	0x00	Delay for PDB8. Delay for rising edge and falling edge programmed separately.
8C	Table 38	P9PDBTIM	0x00	Delay for PDB9. Delay for rising edge and falling edge programmed separately.

Table 38. PnPDBTIM Registers 0x0C, 0x1C, 0x2C, 0x3C, 0x4C, 0x5C, 0x6C, 0x7C, 0x8C

Bit	Name	R/W	Description	Bit	Name	R/W	Description
7–4	TR3–TR0	W	Programmed Rise Time	3–0	TF3–TF0	W	Programmed Fall Time

TR3	TR2	TR1	TR0	Delay (ms)	TF3	TF2	TF1	TF0	Delay (ms)
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	1	1
0	0	1	0	2	0	0	1	0	2
0	0	1	1	5	0	0	1	1	5
0	1	0	0	10	0	1	0	0	10
0	1	0	1	20	0	1	0	1	20
0	1	1	0	40	0	1	1	0	40
0	1	1	1	60	0	1	1	1	60
1	0	0	0	80	1	0	0	0	80
1	0	0	1	100	1	0	0	1	100
1	0	1	0	150	1	0	1	0	150
1	0	1	1	200	1	0	1	1	200
1	1	0	0	250	1	1	0	0	250
1	1	0	1	300	1	1	0	1	300
1	1	1	0	400	1	1	1	0	400
1	1	1	1	500	1	1	1	1	500

OUTPUTS

PROGRAMMABLE DRIVER OUTPUTS

The ADM1060 has nine programmable driver outputs (PDOs). These are the logic outputs of the device. Each PDO is normally controlled by a single PDB. Thus, the PDOs can be set up to assert when the conditions on the PDB are met, such as when the SFDs are in tolerance, the levels on the GPI are correct, the watchdog timer has not timed out, and so on. The PDOs can be used for a number of functions; for example, to provide a POWER_GOOD signal when all the SFDs are in tolerance, provide a reset generator output if one of the SFDs goes out of spec (which can be used as a status signal for a DSP or other microprocessor), or provide enable signals for LDOs on the supplies that the ADM1060 is supervising.

There are a number of pull-up options on the PDOs to enable the user to program the output level.

The outputs can be programmed as

- Open-drain (allows the user to connect an external pull-up resistor)
- Open-drain with weak pull-up to V_{DD}
- Push-pull to V_{DD}
- Open-drain with weak pull-up to VPn
- Push-pull to VPn
- Internally charge-pumped high drive (12 V)

The last option is only available on PDO1–4. This allows the user to directly drive the gate of an N-channel FET in the path of a power supply. The required pull-up is selected by programming Bits 0 to 3 in PnPDOCFG appropriately (see Table 40).

The data driving each of the PDOs can come from one of three inputs. These inputs are enabled by a bit each in the PnPDOCFG registers. The inputs are

- The (delayed) output from the associated PLB (enabled by setting bit CFG4 to 1)
- Data that is driven directly over the SMBus interface (enabled by setting Bit CFG5 to 1). When set in this mode, the data from the PDB is disabled and the data on the PDO is the data on CFG4. Thus, the PDO can be software controlled to initiate a software power-up/power-down.
- An on-chip clock (enabled by setting Bit CFG6 to 1). A 100 kHz clock is available to clock an external device such as an LED.

More details on these data modes are given in the register map of Table 40.

The default setup of each of the PDOs is to be pulled low by a weak (20 kΩ) pull-down resistor. This is also the setup of the PDOs on power-up until the registers are loaded and the programmed conditions are latched. The outputs are actively pulled low once 1 V or greater is seen at any VPn or VH. Until there is a 1 V supply on the chip, the outputs are high impedance. This provides 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 PDO pin to the required pull-up voltage. The 20 kV resistor must be accounted for in calculating a suitable value. For example, if it is required to pull PDOn up to 3.3 V, and 5 V is available as an external supply, the pull-up resistor value is given by:

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

Therefore,

$$R_{UP} = (100\ kV - 66\ kV)/3.3 = 10\ kV$$

The register list and the bit map for the PDOs are shown in Table 39 and Table 40.

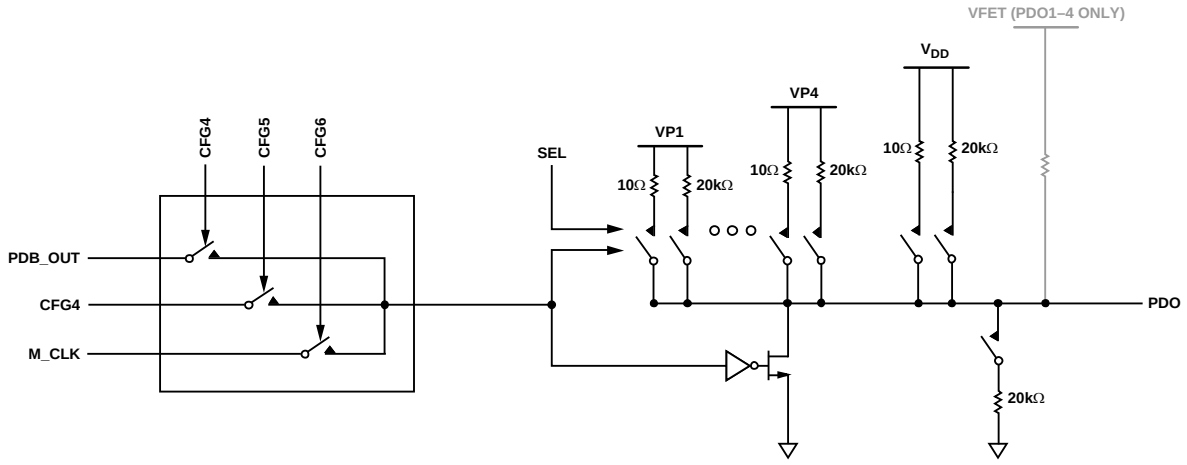

Figure 23. Programmable Driver Output

Table 39. Programmable Driver Outputs Registers

Hex Address	Table	Name	Default Power-On Value	Description
0D	Table 40	P1PDOCFG	0x00	Selects the format of the PDO1 output (open drain, open drain with internal pull-up, charge pumped, etc.).
1D	Table 40	P2PDOCFG	0x00	Selects the format of the PDO2 output (open drain, open drain with internal pull-up, charge pumped, etc.).
2D	Table 40	P3PDOCFG	0x00	Selects the format of the PDO3 output (open drain, open drain with internal pull-up, charge pumped, etc.).
3D	Table 40	P4PDOCFG	0x00	Selects the format of the PDO4 output (open drain, open drain with internal pull-up, charge pumped, etc.).
4D	Table 40	P5PDOCFG	0x00	Selects the format of the PDO5 output (open drain, open drain with internal pull-up, etc.). Note: charge pumped output is not available on this driver.
5D	Table 40	P6PDOCFG	0x00	Selects the format of the PDO6 output (open drain, open drain with internal pull-up, etc.). Note: charge pumped output is not available on this driver.
6D	Table 40	P7PDOCFG	0x00	Selects the format of the PDO7 output (open drain, open drain with internal pull-up, etc.). Note: charge pumped output is not available on this driver.
7D	Table 40	P8PDOCFG	0x00	Selects the format of the PDO8 output (open drain, open drain with internal pull-up etc.). Note: charge pumped output is not available on this driver.
8D	Table 40	P9PDOCFG	0x00	Selects the format of the PDO9 output (open drain, open drain with internal pull-up, etc.). Note: charge pumped output is not available on this driver.

Table 40. PnPDOCFG Register 0x0D, 0x1D, 0x2D, 0x3D, 0x4D, 0x5D, 0x6D, 0x7D, 0x8D (Power-On Default 0x00)

Bit	Name	R/W	Description					
7	Reserved	N/A	Cannot Be Used					
6–4	CFG6–CFG4	R/W	Controls the logical state of the PDO. These three bits determine what effect, if any, the logical input to the PDO has on its output.					
			CFG6	**CFG5**	**CFG4**	**PDO**	**State**	
			0	0	0	0	Disabled, with weak pull-down	
			0	0	1	PLB_OUT	Enabled, follows PLB logic output	
			0	1	0	0	Enables SMBus data, drive low	
			0	1	1	1	Enables SMBus data, drive high	
			1	X	X	MCLK	Enables MCLK out onto pin	
3–0	CFG3–CFG0	R/W	**CFG3**	**CFG2**	**CFG1**	**CFG0**	**Pull-Up Supply**	**Pull-Up Strength**
			0	0	0	X	none	N/A
			0	0	1	X	VCP	300 kΩ
			0	1	0	0	VP1	Low
			0	1	0	1	VP1	High
			0	1	1	0	VP2	Low
			0	1	1	1	VP2	High
			1	0	0	0	VP3	Low
			1	0	0	1	VP3	High
			1	0	1	0	VP4	Low
			1	0	1	1	VP4	High
			1	1	1	0	V_{DD}	Low
			1	1	1	1	V_{DD}	High

STATUS/FAULTS

FAULT/STATUS REPORTING ON THE ADM1060

As discussed previously, any number of the PDOs can be programmed to assert under a set of preprogrammed conditions. These conditions could be a fault on an SFD, a change in status on a GPI, a timeout on the watchdog detector, and so on. Because of the flexibility and the choice of combinations available on the ADM1060, the assertion of the PDO will tell the user nothing about what caused it to assert (unless it is programmed to assert with only one input).

To enable the user to debug the cause of the PDO assertion, a number of registers on the ADM1060 provide status and fault information on the various individual functions supervised by the device.

STATUS REGISTERS

A number of status registers indicate the logic state of all of the functions controlled by the ADM1060. These logic states include the output of both the UV and OV comparators of each of the seven SFDs, the logic output of the SFDs themselves, the logic state of the GPIs, the error condition on the WDI, and the logic state of each of the nine PDOs. The content of these registers, which is read-only, can be read at any time via the SMBus interface. The register and bit map for each of these status registers are described in the tables that follow.

FAULT REGISTERS

The ADM1060 also provides fault reporting. For example, if a fault occurs causing a PDO to change its status, the user can determine what function actually faulted. This is achieved by providing a "fault plane" consisting of two registers, LATF1 and LATF2, that the system controller can read out of the ADM1060 via the SMBus. Each bit in the two registers (with one important exception, see below) is assigned to one of the inputs of the devices as shown in Table 41.

Table 41. Fault Plane of ADM1060

Register	Bit	Assigned Function
LATF1	7	ANYFLT
	6	Logic Output of VP4 SFD
	5	Logic Output of VP3 SFD
	4	Logic Output of VP2 SFD
	3	Logic Output of VP1 SFD
	2	Logic Output of VH SFD
	1	Logic Output of VB2 SFD
	0	Logic Output of VB1 SFD
LATF2	7	
	6	
	5	
	4	Logic Output of WDI
	3	Logic Input on GPI4
	2	Logic Input on GPI3
	1	Logic Input on GPI2
	0	Logic Input on GPI1

Each bit represents the logical status of its assigned function, i.e., the logical output of the SFDs and WDI, and the logic level on the GPI inputs.

The important exception is the MSB of the LATF1 register. This is the ANYFLT bit. This bit goes high if one of the other bits in the two registers faults. A fault is defined as a change in polarity from the last time the fault registers were read. Once ANYFLT goes high, the contents of the two registers are latched, thus preventing more than one of the other bits from changing polarity before the content of the registers is read. Therefore, the first faulting input can be determined.

The sequence in which the registers are read is determined by ANYFLT. As long as ANYFLT remains at 0, only the content of LATF1 is read. There are two reasons for this. The first is that ANYFLT = 0 implies that no fault has occurred and, therefore, there is no need to read the contents of LATF2. The second and more important reason is that reading register LATF2 actually resets the ANYFLT bit to 0. Thus, if a fault occurred on an SFD after LATF1 had been read but before LATF2 had been read, ANYFLT would change to 1, indicating that a fault had occurred, but would be reset to 0 once LATF2 was read, thus erasing the log of the fault. In summary then, LATF2 should only be read if ANYFLT = 1. Reading the registers in this sequence ensures that the contents are never reset before a fault has been logged over the SMBus, thus ensuring that the supervising processor or CPLD knows what function supervised by the ADM1060 caused the fault. The faulting function is determined by comparing the contents of the fault plane (i.e., the contents of the two registers) with the values read previously, and determining which bit changed polarity.

The functionality of the fault plane is best illustrated with an example. For instance, take VP1 to have an input supply of 5.0 V. A UV/OV window of 4.5 V to 5.5 V is set up on VP1. The supply is ramped in and out of this window, each time reading the contents of LATF1 and LATF2. The values recorded are as follows:

1. VP1 at 5 V: LATF1 = LATF2 = 00000000. This is expected. The supply is in tolerance, SFD output is 0, therefore no fault.
2. VP1 at 4.2 V: LATF1 = 10001000, LATF2 = 00000000. SFD output has changed status to 1, therefore ANYFLT goes high.
3. VP1 at 5.0 V: LATF1 = 10000000, LATF2 = 00000000. SFD output has changed status to 0, therefore ANYFLT goes high again.
4. VP1 at 5.8 V: LATF1 = 10001000, LATF2 = 00000000. SFD output again changed status from 0 to 1, so ANYFLT goes high.
5. VP1 at 4.2 V: LATF1 = 10000000, LATF2 = 00000000. At first glance, this would appear to be incorrect since the SFD output should be at 1 (4.2 V is an undervoltage fault). However, in ramping down from 5.8 V to 4.2 V, the supply passed into the UV/OV window, the SFD output changed status from 1 to 0, ANYFLT was set high, and the register contents were latched. It is these values that were read, before being reset by reading LATF2.

There are also two mask registers provided that enable the user to ignore a fault on a given function. The bits of the error mask registers are mapped in the same way as those of the fault registers with the exception that the ANYFLT bit cannot be masked. Setting a 1 in the error mask register results in the equivalent bit in the fault register always remaining at 0, regardless of whether there is a fault on that function or not. The register and bit maps for both the fault and error mask registers are shown below.

Table 42. Status Registers

Hex Addr.	Table	Name	Default Power-On Value	Description
D8	Table 43	UVSTAT	0x00	Logic output of the UV comparator on each of the seven SFDs
D9	Table 44	OVSTAT	0x00	Logic output of the OV comparator on each of the seven SFDs
DA	Table 45	SFDSTAT	0x00	Logic output (post Fault Type block) on each of the seven SFDs
DB	Table 46	GWSTAT	0x00	Logic state of the four GPIs and the Watchdog Fault Detector
DE	Table 49	PDOSTAT1	0x00	Logic output of PDOs 1 to 8
DF	Table 48	PDOSTAT2	0x00	Logic output of PDO 9

Table 43. Bit Map for UVSTAT Register 0xD8 (Power-On Default 0x00)

Bit	Name	R/W	Description
7	Reserved	N/A	Cannot Be Used
6	VP4UV	R	If high, voltage on VP4 input is lower than the UV threshold.
5	VP3UV	R	If high, voltage on VP3 input is lower than the UV threshold.
4	VP2UV	R	If high, voltage on VP2 input is lower than the UV threshold.
3	VP1UV	R	If high, voltage on VP1 input is lower than the UV threshold.
2	VHUV	R	If high, voltage on VH input is lower than the UV threshold.
1	VB2UV	R	If high, voltage on VB2 input is lower than the UV threshold.
0	VB1UV	R	If high, voltage on VB1 input is lower than the UV threshold.

Table 44. Bit Map for OVSTAT Register 0xD9 (Power-On Default 0x00)

Bit	Name	R/W	Description
7	Reserved	N/A	Cannot Be Used
6	VP4OV	R	If high, voltage on VP4 input is higher than the OV threshold.
5	VP3OV	R	If high, voltage on VP3 input is higher than the OV threshold.
4	VP2OV	R	If high, voltage on VP2 input is higher than the OV threshold.
3	VP1OV	R	If high, voltage on VP1 input is higher than the OV threshold.
2	VHOV	R	If high, voltage on VH input is higher than the OV threshold.
1	VB2OV	R	If high, voltage on VB2 input is higher than the OV threshold.
0	VB1OV	R	If high, voltage on VB1 input is higher than the OV threshold.

Table 45. Bit Map for SFDSTAT Register 0xDA (Power-On Default 0x00)

Bit	Name	R/W	Description
7	Reserved	N/A	Cannot Be Used
6	VP4FLT	R	If high, fault (UV, OV or Out-of-Window) has occurred on VP4 input.
5	VP3FLT	R	If high, fault (UV, OV or Out-of-Window) has occurred on VP3 input.
4	VP2FLT	R	If high, fault (UV, OV or Out-of-Window) has occurred on VP2 input.
3	VP1FLT	R	If high, fault (UV, OV or Out-of-Window) has occurred on VP1 input.
2	VHFLT	R	If high, fault (UV, OV or Out-of-Window) has occurred on VH input.
1	VB2FLT	R	If high, fault (UV, OV or Out-of-Window) has occurred on VB2 input.
0	VB1FLT	R	If high, fault (UV, OV or Out-of-Window) has occurred on VB1 input.

Table 46. Bit Map for GWSTAT Register 0xDB (Power-On Default 0x00)

Bit	Name	R/W	Description
7–5	Reserved	N/A	Cannot Be Used
4	WDISTAT	R	If high, timeout has elapsed on the Watchdog Detector.
3	GPI4STAT	R	Logic level currently being driven on GPI4 input.
2	GPI3STAT	R	Logic level currently being driven on GPI3 input.
1	GPI2STAT	R	Logic level currently being driven on GPI2 input.
0	GPI1STAT	R	Logic level currently being driven on GPI1 input.

Table 47. Bit Map for PDOSTAT1 Register 0xDE (Power-On Default 0x00)

Bit	Name	R/W	Description
7	PDO8STAT	R	Logic level currently being driven on PDO8 output.
6	PDO7STAT	R	Logic level currently being driven on PDO7 output.
5	PDO6STAT	R	Logic level currently being driven on PDO6 output.
4	PDO5STAT	R	Logic level currently being driven on PDO5 output.
3	PDO4STAT	R	Logic level currently being driven on PDO4 output.
2	PDO3STAT	R	Logic level currently being driven on PDO3 output.
1	PDO2STAT	R	Logic level currently being driven on PDO2 output.
0	PDO1STAT	R	Logic level currently being driven on PDO1 output.

Table 48. Bit Map for PDOSTAT2 Register 0xDF (Power-On Default 0x00)

Bit	Name	R/W	Description
7–1	Reserved	N/A	Cannot Be Used
0	PDO9STAT	R	Logic level currently being driven on PDO9 output.

FAULT REGISTERS

Table 49. List of Fault Registers

Hex Addr.	Table	Name	Default Power On Value	Description
DC	Table 50	LATF1	0x00	Fault Status Register for the seven SFDs
DD	Table 51	LATF2	0x00	Fault Status Register for the four GPIs and the Watchdog Detector

Table 50. Bit Map for LATF1 Register 0xDC (Power-On Default 0x00)

Bit	Name	R/W	Description
7	ANYFLT	R	If high, a change in logic status (fault) has been logged on one of the 12 functions monitored since the last time the Fault Registers were read.
6	VP4FLT	R	If high, a fault has occurred on supply at input VP4.
5	VP3FLT	R	If high, a fault has occurred on supply at input VP3.
4	VP2FLT	R	If high, a fault has occurred on supply at input VP2.
3	VP1FLT	R	If high, a fault has occurred on supply at input VP1.
2	VHFLT	R	If high, a fault has occurred on supply at input VH.
1	VB2FLT	R	If high, a fault has occurred on supply at input VB2.
0	VB1FLT	R	If high, a fault has occurred on supply at input VB1.

Table 51. Bit Map for LATF2 Register 0xDD (Power-On Default 0x00)

Bit	Name	R/W	Description
7–5	Reserved	N/A	Cannot Be Used
4	WDFLT	R	If high, the logic level on the WDI output has changed since the last time that the fault registers were read.
3	GPI4FLT	R	If high, the logic level on GPI4 input has changed since the last time that the fault registers were read.
2	GPI3FLT	R	If high, the logic level on GPI3 input has changed since the last time that the fault registers were read.
1	GPI2FLT	R	If high, the logic level on GPI2 input has changed since the last time that the fault registers were read.
0	GPI1FLT	R	If high, the logic level on GPI1 input has changed since the last time that the fault registers were read.

MASK REGISTERS

Table 52. List of Mask Registers

Hex Addr.	Table	Name	Default Power On Value	Description
9D	Table 53	ERRMASK1	0x00	Error Mask Register for the seven SFDs
9E	Table 54	ERRMASK2	0x00	Error Mask Register for the four GPIs and the Watchdog Detector

Table 53. Bit Map for ERRMASK1 Register 0x9D (Power-On Default 0x00)

Bit	Name	R/W	Description
7	Reserved	X	Unused
6	VP4MASK	R/W	If high, a fault occurring on the supply at input VP4 is ignored and not logged in LATF1.
5	VP3MASK	R/W	If high, a fault occurring on the supply at input VP3 is ignored and not logged in LATF1.
4	VP2MASK	R/W	If high, a fault occurring on the supply at input VP2 is ignored and not logged in LATF1.
3	VP1MASK	R/W	If high, a fault occurring on the supply at input VP1 is ignored and not logged in LATF1.
2	VHMASK	R/W	If high, a fault occurring on the supply at input VH is ignored and not logged in LATF1.
1	VB2MASK	R/W	If high, a fault occurring on the supply at input VB2 is ignored and not logged in LATF1.
0	VB1MASK	R/W	If high, a fault occurring on the supply at input VB1 is ignored and not logged in LATF1.

Table 54. Bit Map for ERRMASK2 Register 0x9E (Power-On Default 0x00)

Bit	Name	R/W	Description
7–5	Reserved	X	Unused
4	WDIMASK	R/W	If high, a change in the logic level on the WDI output is ignored and not logged in LATF2.
3	GPI4MASK	R/W	If high, a change in the logic level on the GPI4 input is ignored and not logged in LATF2.
2	GPI3MASK	R/W	If high, a change in the logic level on the GPI3 input is ignored and not logged in LATF2.
1	GPI2MASK	R/W	If high, a change in the logic level on the GPI2 input is ignored and not logged in LATF2.
0	GPI1MASK	R/W	If high, a change in the logic level on the GPI1 input is ignored and not logged in LATF2.

PROGRAMMING

CONFIGURATION DOWNLOAD AT POWER-UP

The configuration of the ADM1060—the UV/OV thresholds, glitch filter timeouts, PLB combinations, PDO pull-ups, etc.—is dictated by the contents of the RAM. The RAM is comprised of local latches that set the configuration. These latches are double buffered and are actually comprised of two identical latches (Latch A and Latch B). An update of the double-buffered latch updates Latch A first and then Latch B. The advantage of this architecture is explained below. These latches are volatile memory and lose their contents at power-down. Therefore, at power-up the configuration in the RAM must be restored. This is achieved by downloading the contents of the EEPROM (nonvolatile memory) to the local latches. This download occurs in a number of steps.

1. With no power applied to the device, the PDOs are all high impedance.
2. Once 1 V appears on any of the inputs connected to the V_{DD} arbitrator (VH or VPn), the PDOs are all (weakly) pulled to GND.
3. Once 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. Once the contents of the EEPROM are completely downloaded, the device controller outputs a control pulse enabling all Latch As to download to all Latch Bs, thus completing the configuration download. Any attempt to communicate with the device prior to this download completion will result in a NACK being issued from the ADM1060.

UPDATING THE CONFIGURATION

Once the device is powered up with all of the configuration settings loaded from EEPROM into the RAM registers, the user may wish to alter the configuration of functions on the ADM1060; for example, change the UV or OV limit of an SFD, the fault output of an SFD, the timeout of the watchdog detector, the rise time delay of one of the PDOs, and so on.

The ADM1060 provides a number of options that allow the user to update the configuration differently over the SMBus interface. All of these options are controlled in the register UPDCFG. The options are

1. Update the configuration in real time. The user writes to RAM across the SMBus and the configuration is updated immediately.
2. Update the A Latches "offline" and then update all B Latches at the same time. With this method, the configuration of the ADM1060 will remain unchanged and continue to operate in the original setup until the instruction is given to update the B Latches.
3. Change EEPROM register contents offline and then download the revised EEPROM contents to the RAM registers. Again, with this method, the configuration of the ADM1060 will remain unchanged and continue to operate in the original setup until the instruction is given to change.

The instruction to download from the EEPROM in option 3 above is also a useful way to restore the original EEPROM contents if revisions to the configuration are unsatisfactory and the user wants the ADM1060 to return to a known operating mode.

This type of operation is possible because of the topology of the ADM1060. The local (volatile) registers, or RAM, are all double-buffered latches. Setting Bit 0 of the UPDCFG register to 1 leaves the double-buffered latches open at all times. If Bit 0 is set to 0, then when 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 of the second latches at once. EPROM writes work similarly.

A final bit in this register is used to enable EEPROM page erasure. If this bit is set high, the contents of an EEPROM page can all be set to 0. If 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 register UPDCFG is shown in Table 56. A flow chart for download at power-up and subsequent configuration updates is shown in Figure 24.

Table 55. List of Configuration Update Registers

Hex Addr.	Table	Name	Default Power-On Value	Description
90	Table 56	UPDCFG	0x00	Configuration Update Control register for changing configuration of the ADM1060 after power-up

Table 56. Bit Map for UPDCFG Register 0x90 (Power-On Default 0x00)

Bit	Name	R/W	Description
7–4	Reserved	N/A	Cannot be used
3	EE_ERASE	R/W	If set high, EEPROM page erasure can be programmed.
2	EEPROMLD	W	If set high, the ADM1060 will download the contents of its EEPROM to the RAM registers. This bit self-clears (returns to 0) after the download.
1	RAMLD	W	If set high, the ADM1060 will download the buffered RAM register data into the local latches. This bit self-clears (returns to 0) after the download.
0	UPD	R/W	If set high, the ADM1060 will update its configuration in real time as a word is written to a local RAM register via the SMBus.

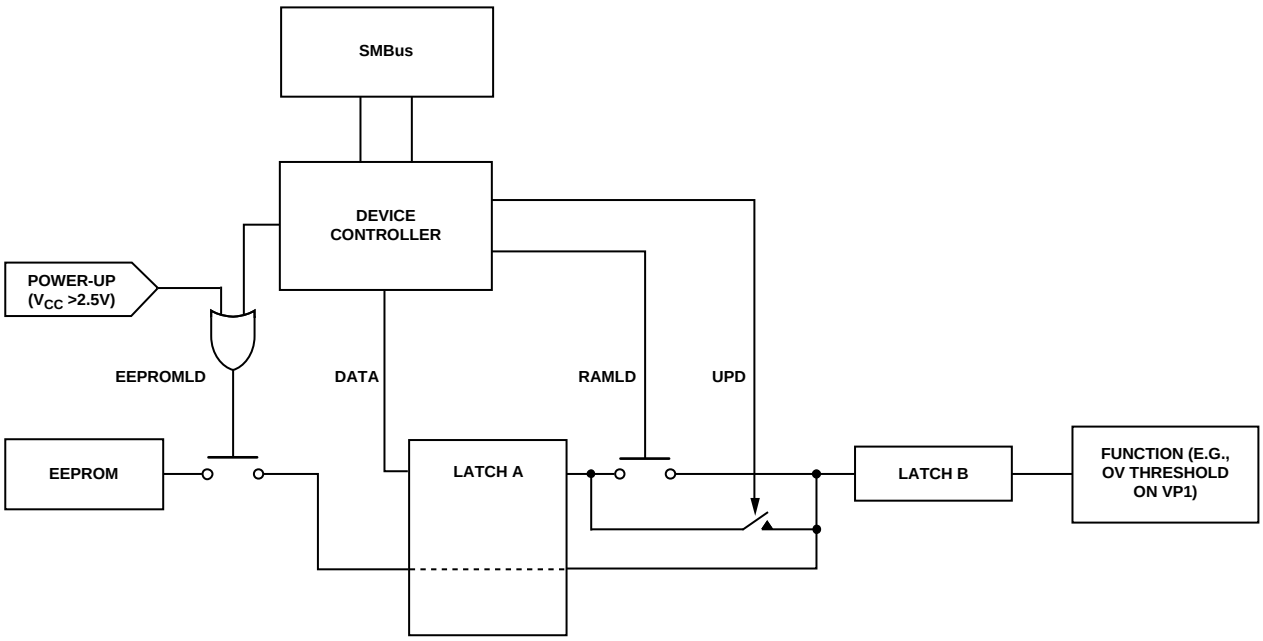

Figure 24. Configuration Update Flow Diagram

INTERNAL REGISTERS

The ADM1060 contains a large number of data registers. A brief description of the principal registers is given below. More detailed descriptions are given in the relevant sections of this data sheet.

Address Pointer Register. This register contains the address that selects one of the other internal registers. When writing to the ADM1060, the first byte of data is always a register address, which is written to the Address Pointer register.

Configuration Registers. These registers provide control and configuration for various operating parameters of the ADM1060.

Polarity Registers. These registers define the polarity of inputs to the PLBA.

Mask Registers. These registers allow masking of individual inputs to the PLBA and masking of faults in the fault reporting registers.

EEPROM

The ADM1060 has 512 bytes of nonvolatile, electrically erasable programmable read-only memory (EEPROM) from register addresses 0xF800 to 0xF9FF. This may be used for permanent storage of data that will not be lost when the ADM1060 is powered down, unlike the data in the volatile registers. Although referred to as read-only memory, the EEPROM can be written to (as well as read from) via the serial bus in exactly the same way as the other registers. The only major differences between the EEPROM and other registers are

1. An EEPROM location must be blank before it can be written to. If it contains data, it must first be erased.
2. Writing to EEPROM is slower than writing to RAM.
3. 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 EEPROM is split into 16 (0 to 15) pages of 32 bytes each. Pages 0 to 6, starting at address 0xF800, hold the configuration data for the applications on the ADM1060 (the PLB, SFDs, GPIs, WDI, PDOs, etc.). These EEPROM addresses are the same as the RAM register addresses, prefixed by 0xF8. Page 7 is reserved. Pages 8 to 15 are for customer use. Data can be downloaded from EEPROM to RAM in one of two ways:

1. At power-up, pages 0 to 6 are downloaded.
2. Setting Bit 2 of the UPDCFG register (0x90) performs a user download of pages 0 to 6.

SERIAL BUS INTERFACE

Control of the ADM1060 is carried out via the serial system management bus (SMBus). The ADM1060 is connected to this bus as a slave device under the control of a master device. It takes approximately 2 ms after power-up for the ADM1060 to download from its EEPROM. Therefore, access to the ADM1060 is restricted until the download is completed.

IDENTIFYING THE ADM1060 ON THE SMBus

The ADM1060 has a 7-bit serial bus slave address. When the device is powered up, it will do so with a default serial bus address. The five MSBs of the address are set to 10101, and the two LSBs are determined by the logical states of Pins A1 and A0. This allows the connection of four ADM1060s to the one SMBus. The device also has a number of identification registers (read only) that can be read across the SMBus. These are

Name	Address	Value	Function
MANID	0x93	0x41	Manufacturer ID for Analog Devices
DEVID	0x94	0x3E	Device ID
REVID	0x95	0x– –	Silicon Revision
MARK1	0x96	0x– –	S/W Brand
MARK2	0x97	0x– –	S/W Brand

GENERAL SMBus TIMING

Figure 25 and Figure 26 show timing diagrams for general read and write operations using the SMBus. The SMBus specification defines specific conditions for different types of read and write operation, which are discussed later. The general SMBus protocol operates as follows:

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 will follow. 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 a R/$\overline{W}$ bit, which determines the direction of the data transfer, i.e., whether data will be 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 holding it low during the high period of this clock pulse. All other devices on the bus now remain idle while the selected device waits for data to be read from or written to it. If the R/$\overline{W}$ bit is a 0, the master will write to the slave device. If the R/$\overline{W}$ bit is a 1, the master will read from the slave device.

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, as 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 tells the slave device what to expect next. It may be an instruction such as telling the slave device to expect a block write, or it may simply be a register address that tells the slave where subsequent data is to be written.

   Since data can flow in only one direction as defined by the R/$\overline{W}$ bit, it is not possible to send a command to a slave device during a read operation. Before doing a read operation, it may first be necessary to do a write operation to tell the slave what sort of read operation to expect and/or the address from which data is to be read.

3. When all data bytes have been read or written, stop conditions are established. In WRITE mode, the master will pull the data line high during the 10th clock pulse to assert a STOP condition. In READ mode, the master device will release the SDA line during the low period before the ninth clock pulse, but the slave device will not pull it low. This is known as No Acknowledge. The master will then take the data line low during the low period before the 10th clock pulse, then high during the 10th clock pulse to assert a STOP condition.

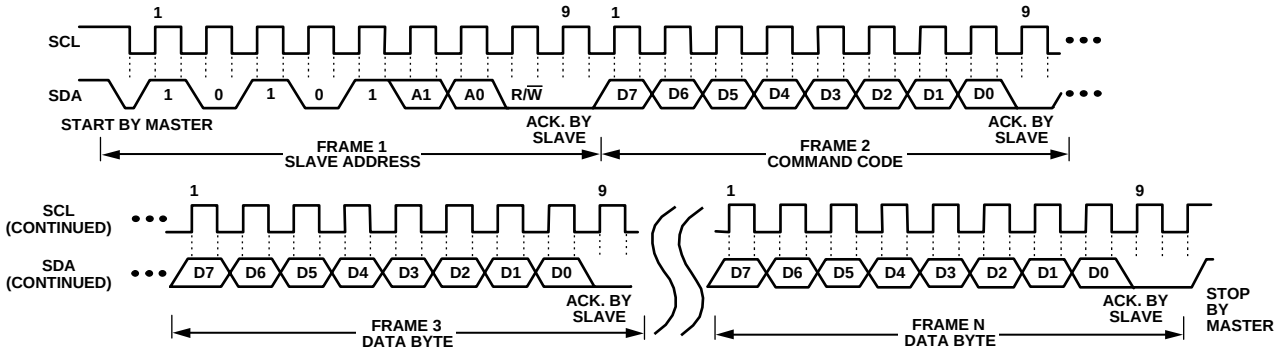

Figure 25. General SMBus Write Timing Diagram

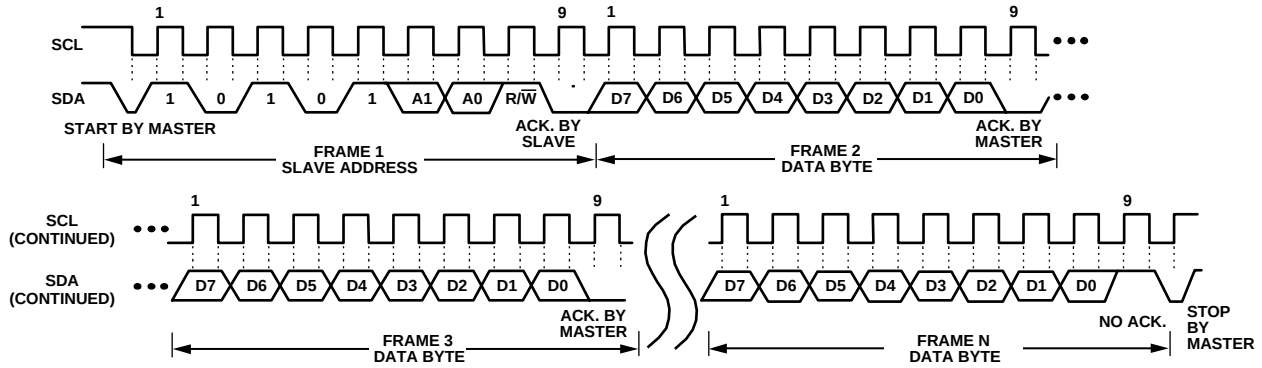

Figure 26. General SMBus Read Timing Diagram

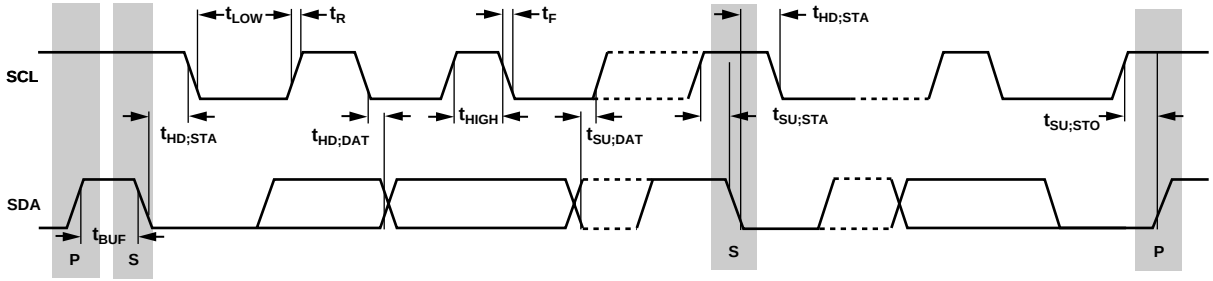

Figure 27. Serial Bus Timing Diagram

SMBus PROTOCOLS FOR RAM AND EEPROM

The ADM1060 contains volatile registers (RAM) and nonvolatile EEPROM. User RAM occupies address locations from 0x00 to 0xDF, while EEPROM occupies addresses from 0xF800 to 0xF9FF.

Data can be written to and read from both RAM and EEPROM as single data bytes.

Data can be written only to unprogrammed EEPROM locations. To write new data to a programmed location, it is first necessary to erase it. EEPROM erasure cannot be done at the byte level; the EEPROM is arranged as 16 pages of 32 bytes, and an entire page must be erased.

Page erasure is enabled by setting Bit 3 in register UPDCFG (address 0x90) to 1. If this 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 ones used in the ADM1060 are discussed below. The following abbreviations are used in the diagrams:

S	START
P	STOP
R	READ
W	WRITE
A	ACKNOWLEDGE
$\overline{A}$	NO ACKNOWLEDGE

The ADM1060 uses the following SMBus write protocols:

SEND BYTE

In this 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 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 ADM1060, the send byte protocol is used for two purposes:

1. To write a register address to RAM for a subsequent single byte read from the same address or block read, or to write starting at that address. This is illustrated in Figure 28.

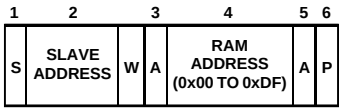

Figure 28. Setting a RAM Address for Subsequent Read

2. 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 or pages containing those locations must first be erased. EEPROM memory is erased by writing a command byte.

   The master sends a command code that tells the slave device to erase the page. The ADM1060 command code for a page erasure is 0xFE (1111 1110 binary). Note that in order for page erasure to take place, the page address has to be given in the previous write word transaction (see write byte below). Also, Bit 3 in register UPDCFG (address 0x90) must be set to 1.

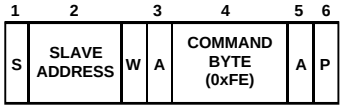

Figure 29. EEPROM Page Erasure

   As soon as the ADM1060 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 ADM1060 is accessed before erasure is complete, it will respond with No Acknowledge.

WRITE BYTE/WORD

In this 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 may assert STOP at this point).
9. The slave asserts ACK on SDA.
10. The master asserts a STOP condition on SDA to end the transaction.

In the ADM1060, the write byte/word protocol is used for three purposes:

1. To write a single byte of data to RAM. In this case the command byte is the RAM address from 0x00 to 0xDF and the (only) data byte is the actual data. This is illustrated in Figure 30.

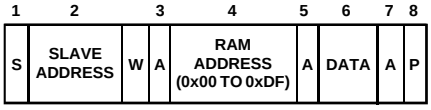

Figure 30. Single Byte Write to RAM

2. 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 the EEPROM address from 0xF8 to 0xF9. The (only) data byte is the low byte of the EEPROM address. This is illustrated in Figure 31.

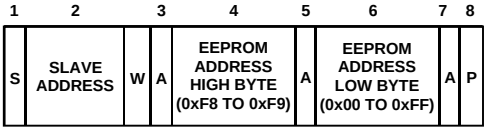

Figure 31. Setting an EEPROM Address

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

3. To write a single byte of data to EEPROM. In this case the command byte is the high byte of the EEPROM address from 0xF8 to 0xF9. The first data byte is the low byte of the EEPROM address and the second data byte is the actual data. This is illustrated in Figure 32.

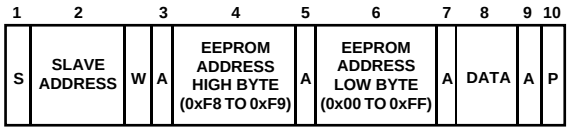

Figure 32. Single Byte Write to EEPROM

BLOCK WRITE

In this operation, the master device writes a block of data to a slave device. The start address for a block write must previously have been set. In the case of the ADM1060, this is done by a Send Byte operation to set a RAM address or a Write Byte/Word operation to set an EEPROM address.

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 write. The ADM1060 command code for a block write is 0xFC (1111 1100 binary).
5. The slave asserts ACK on SDA.
6. The master sends a data byte that tells the slave device how many data bytes will be sent. The SMBus specification allows a maximum of 32 data bytes to be sent in a block write.
7. The slave asserts ACK on SDA.
8. The master sends N data bytes.
9. The slave asserts ACK on SDA after each data byte.
10. The master asserts a STOP condition on SDA to end the transaction.

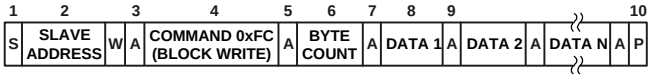

Figure 33. Block Write to EEPROM or RAM

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

1. There must be at least N locations from the start address to the highest EEPROM address (0xF9FF) to avoiding writing to invalid addresses.
2. If the addresses cross a page boundary, both pages must be erased before programming.

Note that the ADM1060 features a clock extend function for writes to EEPROM. Programming an EEPROM byte takes approximately 250 µs, which would limit the SMBus clock for repeated or block write operations. The ADM1060 pulls SCL low and extends the clock pulse when it cannot accept any more data.

READ OPERATIONS

The ADM1060 uses the following SMBus read protocols:

RECEIVE BYTE

In this operation, the master device receives a single byte from 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 read bit (high).
3. The addressed slave device asserts ACK on SDA.
4. The master receives a data byte.
5. The master asserts NO ACK on SDA.
6. The master asserts a STOP condition on SDA and the transaction ends.

In the ADM1060, 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. This is illustrated in Figure 34.

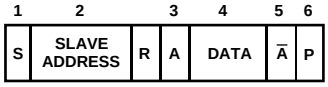

Figure 34. Single Byte Read from EEPROM or RAM

BLOCK READ

In this operation, the master device reads a block of data from a slave device. The start address for a block read must previously have been set. In the case of the ADM1060, 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 ADM1060 command code for a block read is 0xFD (1111 1101 binary).
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 ADM1060 sends a byte count data byte that tells the master how many data bytes to expect. The ADM1060 will always return 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.

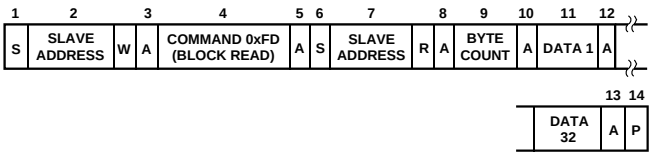

Figure 35. Block Read from EEPROM or RAM

ERROR CORRECTION

The ADM1060 provides the option of issuing a PEC (packet error correction) byte after a write to RAM, a write to EEPROM, a block write to RAM/EEPROM, or a block read from RAM/EEPROM. This enables the user to verify that the data received by or sent from the ADM1060 is correct. The PEC byte is an optional byte sent after the last data byte has been written to or read from the ADM1060. The protocol is as follows:

1. The ADM1060 issues a PEC byte to the master. The master should check the PEC byte and issue another block read if the PEC byte is incorrect.
2. A NACK is generated after the PEC byte to signal the end of the read.

Note: The PEC byte is calculated using CRC-8. The Frame Check Sequence (FCS) conforms to CRC-8 by the polynomial

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

Consult the SMBus 1.1 specification for more information. An example of a block read with the optional PEC byte is shown in Figure 36.

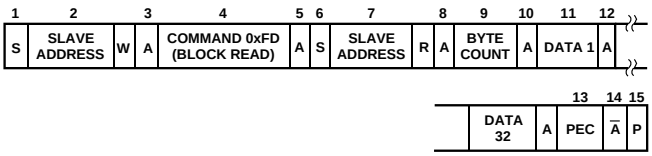

Figure 36. Block Read from EEPROM or RAM with PEC

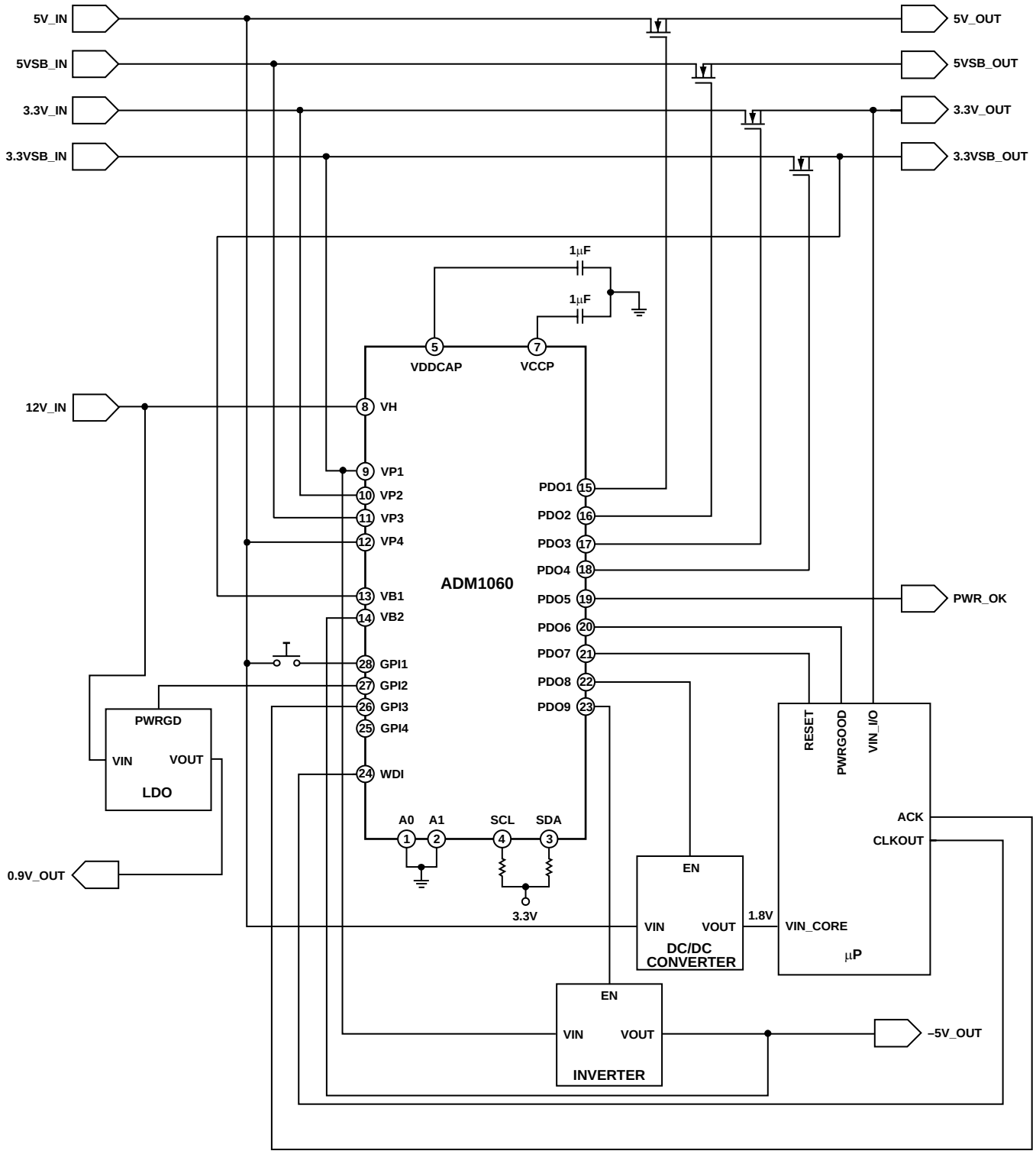

Figure 37. ADM1060 Application Diagram

Table 57. ADM1060 Register Map

BLOCK		0	1	2	3	4	5	6	7	8	9	A	B	C	D
PLB1	0	P1PLBPOLA	P1PLBIMKA	P1SFDPOLA	P1SFDIMKA	P1GPIPOL	P1GPIIMK	P1WDICFG	P1EN	P1PLBPOLB	P1PLBIMKB	P1SFDPOLB	P1SFDIMKB	P1PDBTIM	P1PDOCFG
PLB2	1	P2PLBPOLA	P2PLBIMKA	P2SFDPOLA	P2SFDIMKA	P2GPIPOL	P2GPIIMK	P2WDICFG	P2EN	P2PLBPOLB	P2PLBIMKB	P2SFDPOLB	P2SFDIMKB	P2PDBTIM	P2PDOCFG
PLB3	2	P3PLBPOLA	P3PLBIMKA	P3SFDPOLA	P3SFDIMKA	P3GPIPOL	P3GPIIMK	P3WDICFG	P3EN	P3PLBPOLB	P3PLBIMKB	P3SFDPOLB	P3SFDIMKB	P3PDBTIM	P3PDOCFG
PLB4	3	P4PLBPOLA	P4PLBIMKA	P4SFDPOLA	P4SFDIMKA	P4GPIPOL	P4GPIIMK	P4WDICFG	P4EN	P4PLBPOLB	P4PLBIMKB	P4SFDPOLB	P4SFDIMKB	P4PDBTIM	P4PDOCFG
PLB5	4	P5PLBPOLA	P5PLBIMKA	P5SFDPOLA	P5SFDIMKA	P5GPIPOL	P5GPIIMK	P5WDICFG	P5EN	P5PLBPOLB	P5PLBIMKB	P5SFDPOLB	P5SFDIMKB	P5PDBTIM	P5PDOCFG
PLB6	5	P6PLBPOLA	P6PLBIMKA	P6SFDPOLA	P6SFDIMKA	P6GPIPOL	P6GPIIMK	P6WDICFG	P6EN	P6PLBPOLB	P6PLBIMKB	P6SFDPOLB	P6SFDIMKB	P6PDBTIM	P6PDOCFG
PLB7	6	P7PLBPOLA	P7PLBIMKA	P7SFDPOLA	P7SFDIMKA	P7GPIPOL	P7GPIIMK	P7WDICFG	P7EN	P7PLBPOLB	P7PLBIMKB	P7SFDPOLB	P7SFDIMKB	P7PDBTIM	P7PDOCFG
PLB8	7	P8PLBPOLA	P8PLBIMKA	P8SFDPOLA	P8SFDIMKA	P8GPIPOL	P8GPIIMK	P8WDICFG	P8EN	P8PLBPOLB	P8PLBIMKB	P8SFDPOLB	P8SFDIMKB	P8PDBTIM	P8PDOCFG
PLB9	8	P9PLBPOLA	P9PLBIMKA	P9SFDPOLA	P9SFDIMKA	P9GPIPOL	P9GPIIMK	P9WDICFG	P9EN	P9PLBPOLB	P9PLBIMKB	P9SFDPOLB	P9SFDIMKB	P9PDBTIM	P9PDOCFG
FLT/STS GPI/WDI	9	UPDCFG	PDEN		MANID	DEVID	REVID	MARK1	MARK2	GPI1CFG	GPI2CFG	GPI3CFG	GPI4CFG	WDICFG	ERRMASK1 E ERRMASK2
BSFD1/2	A	BS1OVTH	BS1OVHYST	BS1UVTH	BS1UVHYST	BS1SEL				BS2OVTH	BS2OVHYST	BS2UVTH	BS2UVHYST	BS2SEL	
H/PSFD1	B	HSOVTH	HSOVHYST	HSUVTH	HSUVHYST	HSSEL				PS1OVTH	PS1OVHYST	PS1UVTH	PS1UVHYST	PS1SEL	
PSFD2/3	C	PS2OVTH	PS2OVHYST	PS2UVTH	PS2UVHYST	PS2SEL				PS3OVTH	PS3OVHYST	PS3UVTH	PS3UVHYST	PS3SEL	
PSFD4/ FLT/STS	D	PS4OVTH	PS4OVHYST	PS4UVTH	PS4UVHYST	PS4SEL				UVSTAT	OVSTAT	SFDSTAT	GWSTAT	LATF1	LATF2 E PDOSTAT1 F PDOSTAT2

PIN CONFIGURATION AND FUNCTIONAL DESCRIPTIONS

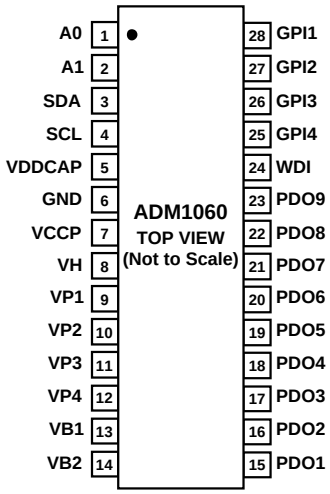

Figure 38. Pin Configuration

Table 58. Pin Function Descriptions

Pin	Mnemonic	Function
1	A0	Logic Input. Controls the seventh bit (LSB) of the 7-bit Serial Bus Address.
2	A1	Logic Input. Controls the sixth bit of the 7-bit Serial Bus Address.
3	SDA	Serial Bus Data I/O Pin. Open-Drain output. Requires 2.2 kΩ pull-up resistor.
4	SCL	Open-Drain Serial Bus Clock Pin. Requires 2.2 kΩ pull-up resistor.
5	VDDCAP	V_{DD} Bypass Capacitor Pin. A capacitor from this pin to GND stabilizes the V_{DD} Arbitrator. A 1 µF capacitor is recommended for this function.
6	GND	Ground. Connect to common of power supplies.
7	VCCP	Reservoir Capacitor for Central Charge Pump. This provides the first stage in the tripler circuits used to produce 12 V of gate drive on PDOs 1 to 4. A 1 µF capacitor is recommended for this function.
8	VH	High Voltage Supply Input. Two input ranges. A supply of between 2 V and 6 V or between 4.8 V and 14.4 V can be applied to this pin. The V_{DD} arbitrator will select this supply to power the ADM1060 if it is the highest supply supervised.
9–12	VP1–4	Positive Only Supply Inputs. Three input ranges. A supply of between 0.6 V and 1.8 V, 1 V and 3 V, or 2 V and 6 V can be applied to this pin. The V_{DD} arbitrator will select one of these supplies to power the ADM1060 if it is the highest supply supervised.
13–14	VB1–2	Bipolar Supply Inputs. Two modes. Two input ranges in positive mode. One input range in negative mode. A supply of between –6 V and –2 V can be applied to this pin when set in negative mode. A supply of between 1 V and 3 V or between 2 V and 6 V can be applied to this pin when set in positive mode.
15–23	PDO1–9	Programmable Driver Output Pin. All nine can be programmed as logic outputs with multiple pull-up options to V_{DD} or VPn. PDOs 1 to 4 can also provide a charge-pump generated gate drive for external N-channel FETs.
24	WDI	Watchdog Input. Used to monitor a processor clock and asserts a fault condition if the clock fails to transition from low-to-high or high-to-low within a programmed timeout period (up to 18 sec).
25–28	GPI4–1	General-Purpose Logic Input. TTL compatible logic. Can be used as, for example, a manual reset, a chip enable pin, or an input for a control logic signal that may be used to initiate the power-up/power-down sequence of the supplies under control.

OUTLINE DIMENSIONS

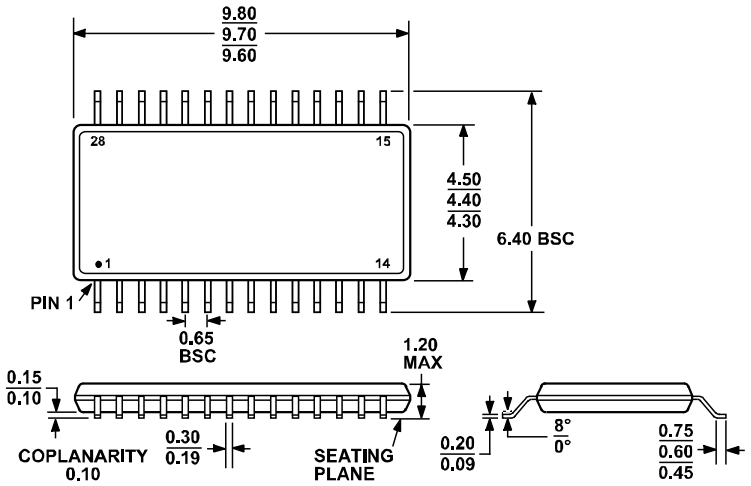

Figure 39. 28-Lead Thin Shrink Small Outline Package [TSSOP]
(RU-28)
Dimensions shown in millimeters

ESD CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although this product features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.

Ordering Guide

Model	Temperature Range	Package Description	Package Option
ADM1060ARU	–40°C to +85°C	28-lead TSSOP	RU-28
ADM1060ARU–REEL	–40°C to +85°C	28-lead TSSOP	RU-28
ADM1060ARU–REEL7	–40°C to +85°C	28-lead TSSOP	RU-28
EVAL–ADM1060EB[1]		Evaluation Board	

[1]Contact factory for availability of the evaluation board.

For general ADM1060 support, send email to: ADM1060.support@analog.com

NOTES

NOTES

C03470–0–12/03(B)