Cascadable Super Sequencer with Margin Control and Fault Recording

Data Sheet **ADM1266**

FEATURES

- Complete supervisory and sequencing solution for up to 17 supplies
  - Expandable to 257 supplies with additional ADM1266 ICs connected to the 2-wire interdevice bus
  - Fully programmable sequencing engine
- 17 supply fault detectors enable real time supervision of supplies
  - 0.4 V to 15 V on VH1 to VH4 (VHx)
  - 0.4 V to 5 V on VP1 to VP13 (VPx)
  - Device powered by the higher of VH1 and VH2 inputs for improved operating redundancy
- 12-bit ADC for readback of all supervised voltages
- Black box nonvolatile fault recording
- 16 PDIOs
- 9 GPIOs
- 9 voltage output 8-bit DACs allow voltage margining adjustment via dc-to-dc converter trim/feedback node
- Main and backup memory
- Industry standard PMBus interface compliant
- Available in a 9 mm × 9 mm, 64-lead package

APPLICATIONS

- Communications infrastructure
- Industrial test and measurement

GENERAL DESCRIPTION

The ADM1266 Super Sequencer® is a configurable supervisory/sequencing device that offers a single-chip solution for supply monitoring and sequencing in systems with up to 17 supplies. For systems with more supplies (up to 257), the operation of up to 16 ADM1266 devices can be synchronized through a proprietary 2-wire interface (interdevice bus).

The sequencing engine (SE) monitors the supply fault detectors (SFDs), programmable driver input/outputs (PDIOs), general-purpose inputs/outputs (GPIOs), and timers, and controls the PDIOs and GPIOs to sequence the supplies up and down as required. The logical core of the device is an ARM® Cortex-M3 microcontroller. The firmware is supplied by Analog Devices, Inc., and all configuration is performed through an intuitive graphic user interface (GUI).

Additionally, the ADM1266 integrates an analog-to-digital converter (ADC) and voltage output digital-to-analog converters (DACs) that can be used to adjust either the feedback node or reference of a dc-to-dc converter to implement a closed-loop, autonomous, margining system.

A block of nonvolatile EEPROM is available to record voltage, time, and fault information when instructed to by the sequencing engine configuration.

FUNCTIONAL BLOCK DIAGRAM

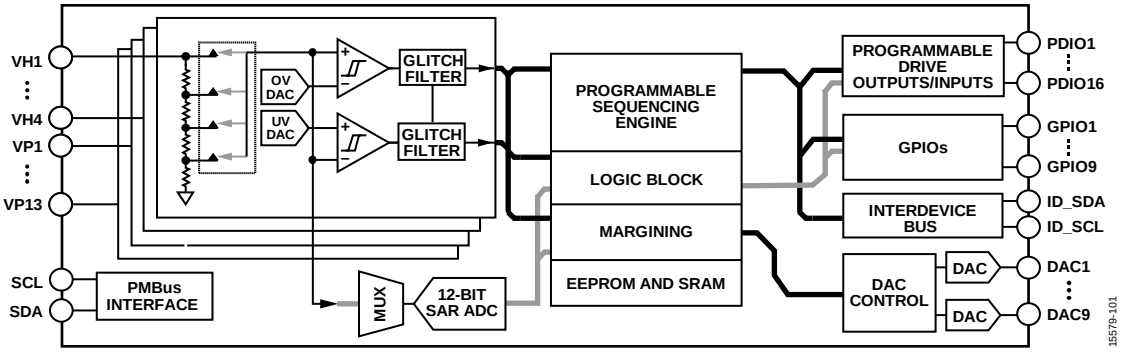


Figure 1.

Rev. B

Document Feedback

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700
Technical Support www.analog.com

TABLE OF CONTENTS

Features ... 1
Applications ... 1
General Description ... 1
Functional Block Diagram ... 1
Revision History ... 3
Detailed Functional Block Diagram ... 4
Specifications ... 5
Absolute Maximum Ratings ... 11
Thermal Resistance ... 11
ESD Caution ... 11
Pin Configuration and Function Descriptions ... 12
Typical Performance Characteristics ... 14
Theory of Operation ... 15
Powering the ADM1266 ... 15
Inputs ... 16
Programmable Driver Input/Outputs ... 17
General-Purpose Input/Outputs ... 18
Sequencing Engine (SE) ... 19
Overview ... 19
Power-Up and State 0 ... 19
State Sections ... 19
Action Types ... 19
Parallel Operation and Interdevice Bus ... 19
States ... 20
Supply Margining ... 21
Overview ... 21
Black Box (EEPROM) Fault Recording ... 23
Black Box Writes When External Supply Is Powering Down ... 23
Triggering a Black Box Write ... 23
Black Box Record Mode ... 23
Power-Up Counter ... 23
Black Box Write Time ... 23
Black Box Contents ... 23
Time Stamping ... 24
Setting UNIX Time Using SET_RTC ... 24
Internal Oscillator ... 24
External Oscillator ... 24
Multiple Device Time Stamping ... 24
System Logic Block ... 25
Password Protection ... 26
Unlocking the Device ... 26
Locking the Device ... 26
Changing the Password ... 26
Memory ... 27
Overview ... 27
Power-Up ... 27
Manual CRC Calculations ... 27
Refresh ... 27
Auto Refresh ... 27
Acceleration Factor ... 27
Applications Information ... 29
Overview ... 29
Powering the ADM1266 ... 29
PCB Assembly and Layout Suggestions ... 29
Capacitors ... 29
Ground Connections ... 29
PMBus/I²C ... 29
IDB ... 29
Voltage Sensing ... 29
PDIOs and GPIOs ... 29
DAC Outputs ... 29
Clock ... 29
Unused Pins ... 29
PMBus Digital Communication ... 30
PMBus Features ... 30
Overview ... 30
Transfer Protocol ... 30
Data Transfer Commands ... 31
Group Command Protocol ... 32
Clock Generation and Stretching ... 32
Start and Stop Conditions ... 32
Repeated Start Condition ... 32
General Call Support ... 32
PMBus Address Selection ... 33
Fast Mode ... 33
10-Bit Addressing ... 33
Packet Error Checking ... 33
Electrical Specifications ... 33
PMBus Commands ... 34

Standard PMBus Command Descriptions 35
Standard PMBus Commands 35
Outline Dimensions 63
Ordering Guide 63

REVISION HISTORY

7/2019—Rev. A to Rev. B

Added Table 3; Renumbered Sequentially 7
Change to Table 4 Summary 8
Added Table 5 10
Change to Setting UNIX Time Using SET_RTC Section 24
Changes to Refresh Section 27
Changes to Acceleration Factor Section 28
Added Table 12 28
Changes to Table 51 42
Changes to Table 80 52
Changes to Table 83 53
Changes to Table 88 54
Changes to Table 104 59
Changes to Ordering Guide 63

8/2018—Rev. 0 to Rev. A

Added Applications Section 1
Change to Bytes[3:1] Description Column, Table 50 41
Changes to Byte 1 Description Column, Table 90 54
Change to Byte Column, Table 94 55
Changes to Byte 1 Description Column, Table 100 57

5/2018—Revision 0: Initial Version

DETAILED FUNCTIONAL BLOCK DIAGRAM

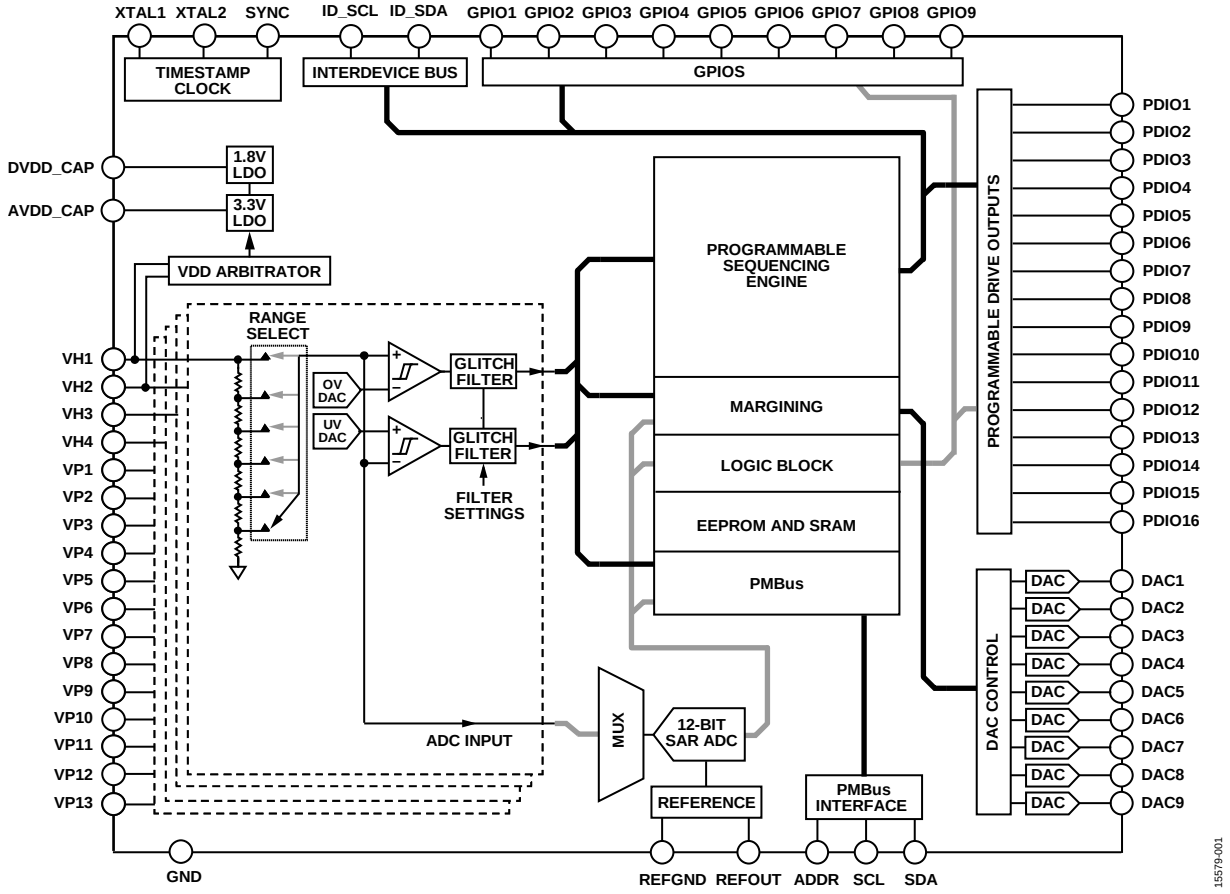


Figure 2.

SPECIFICATIONS

T_J = 0°C to +85°C, VH1 and VH2 > 3 V, unless otherwise noted. Accuracy (%) = (measured voltage − applied voltage) × 100/applied voltage.

Table 1.

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
ADC, SINGLE-ENDED					
Accuracy of VHx Pins with 16× Averaging					
6 V to 15 V			±0.64	%	VHx = 10.623 V
3 V to 7.5 V			±0.64	%	VHx = 5.311 V
1.5 V to 3.75 V			±0.62	%	VHx = 2.656 V
750 mV to 1.875 V			±0.66	%	VHx = 1.328 V
400 mV to 1 V			±0.69	%	VHx = 708 mV
Accuracy of VPx Pins with 16× Averaging					
2 V to 5 V			±0.79	%	VPx = 3.541 V
1.5 V to 3.75 V			±0.79	%	VPx = 2.656 V
750 mV to 1.875 V			±0.70	%	VPx = 1.328 V
400 mV to 1 V					
Direct			±0.67	%	VPx = 708 mV
High-Z			±0.66	%	VPx = 708 mV
SUPPLY FAULT DETECTORS					
Accuracy of VHx Pins					
6 V to 15 V			±0.65	%	VHx = 10.623 V
3 V to 7.5 V			±0.65	%	VHx = 5.311 V
1.5 V to 3.75 V			±0.64	%	VHx = 2.656 V
750 mV to 1.875 V			±0.79	%	VHx = 1.328 V
400 mV to 1 V			±1.02	%	VHx = 708 mV
Accuracy of VPx Pins					
2 V to 5 V			±1.05	%	VPx = 3.541 V
1.5 V to 3.75 V			±0.89	%	VPx = 2.656 V
750 mV to 1.875 V			±0.76	%	VPx = 1.328 V
400 mV to 1 V					
Direct			±0.76	%	VPx = 708 mV
High-Z			±0.71	%	VPx = 708 mV
ADC, DIFFERENTIAL					
Accuracy of VPx Pins with 16× Averaging					
2 V to 5 V			±0.90	%	VPx = 3.541 V
1.5 V to 3.75 V			±0.72	%	VPx = 2.656 V
750 mV to 1.875 V			±0.64	%	VPx = 1.328 V
400 mV to 1 V					
Direct			±0.63	%	VPx = 708 mV
High-Z			±0.61	%	VPx = 708 mV

T_J = −40°C to +85°C, VH1 and VH2 > 3 V, unless otherwise noted. Accuracy (%) = (measured voltage − applied voltage) × 100/applied voltage.

Table 2.

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
ADC, SINGLE-ENDED					
Accuracy of VHx Pins					
6 V to 15 V			±0.64	%	VHx = 10.623 V
3 V to 7.5 V			±0.64	%	VHx = 5.311 V
1.5 V to 3.75 V			±0.62	%	VHx = 2.656 V
750 mV to 1.875 V			±0.66	%	VHx = 1.328 V
400 mV to 1 V			±0.69	%	VHx = 708 mV
Accuracy of VPx Pins					
2 V to 5 V			±0.90	%	VPx = 3.541 V
1.5 V to 3.75 V			±0.79	%	VPx = 2.656 V
750 mV to 1.875 V			±0.70	%	VPx = 1.328 V
400 mV to 1 V					
Direct			±0.67	%	VPx = 708 mV
High-Z			±0.66	%	VPx = 708 mV
SUPPLY FAULT DETECTORS					
Accuracy of VHx Pins					
6 V to 15 V			±0.73	%	VHx = 10.623 V
3 V to 7.5 V			±0.67	%	VHx = 5.311 V
1.5 V to 3.75 V			±0.64	%	VHx = 2.656 V
750 mV to 1.875 V			±0.79	%	VHx = 1.328 V
400 mV to 1 V			±1.02	%	VHx = 708 mV
Accuracy of VPx Pins					
2 V to 5 V			±1.15	%	VPx = 3.541 V
1.5 V to 3.75 V			±0.98	%	VPx = 2.656 V
750 mV to 1.875 V			±0.85	%	VPx = 1.328 V
400 mV to 1 V					
Direct			±0.80	%	VPx = 708 mV
High-Z			±0.78	%	VPx = 708 mV
ADC, DIFFERENTIAL					
Accuracy of VPx Pins with 16× Averaging					
2 V to 5 V			±0.99	%	VPx = 3.541 V
1.5 V to 3.75 V			±0.82	%	VPx = 2.656 V
750 mV to 1.875 V			±0.64	%	VPx = 1.328 V
400 mV to 1 V					
Direct			±0.63	%	VPx = 708 mV
High-Z			±0.61	%	VPx = 708 mV

T_J = −40°C to +105°C, VH1 and VH2 > 3 V, unless otherwise noted. Accuracy (%) = (measured voltage − applied voltage) × 100/applied voltage.

Table 3.

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
ADC, SINGLE-ENDED					
Accuracy of VHx Pins					
6 V to 15 V			±0.74	%	VHx = 10.623 V
3 V to 7.5 V			±0.73	%	VHx = 5.311 V
1.5 V to 3.75 V			±0.76	%	VHx = 2.656 V
750 mV to 1.875 V			±0.73	%	VHx = 1.328 V
400 mV to 1 V			±0.77	%	VHx = 708 mV
Accuracy of VPx Pins					
2 V to 5 V			±0.95	%	VPx = 3.541 V
1.5 V to 3.75 V			±0.86	%	VPx = 2.656 V
750 mV to 1.875 V			±0.78	%	VPx = 1.328 V
400 mV to 1 V					
Direct			±0.75	%	VPx = 708 mV
High-Z			±0.74	%	VPx = 708 mV
SUPPLY FAULT DETECTORS					
Accuracy of VHx Pins					
6 V to 15 V			±0.98	%	VHx = 10.623 V
3 V to 7.5 V			±1.00	%	VHx = 5.311 V
1.5 V to 3.75 V			±0.98	%	VHx = 2.656 V
750 mV to 1.875 V			±1.13	%	VHx = 1.328 V
400 mV to 1 V			±1.28	%	VHx = 708 mV
Accuracy of VPx Pins					
2 V to 5 V			±1.15	%	VPx = 3.541 V
1.5 V to 3.75 V			±1.03	%	VPx = 2.656 V
750 mV to 1.875 V			±1.01	%	VPx = 1.328 V
400 mV to 1 V					
Direct			±0.97	%	VPx = 708 mV
High-Z			±0.97	%	VPx = 708 mV
ADC, DIFFERENTIAL					
Accuracy of VPx Pins with 16× Averaging					
2 V to 5 V			±0.99	%	VPx = 3.541 V
1.5 V to 3.75 V			±0.82	%	VPx = 2.656 V
750 mV to 1.875 V			±0.73	%	VPx = 1.328 V
400 mV to 1 V					
Direct			±0.94	%	VPx = 708 mV
High-Z			±0.75	%	VPx = 708 mV

T_J = −40°C to +105°C, VH1 and VH2 > 3 V, unless otherwise noted.

Table 4.

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
POWER SUPPLY					
VH1 and VH2	3.0		15.0	V	Minimum supply required on one of VH1/VH2 pins
Supply Current, $I_{VH1/VH2}$		16	50	mA	Depends on pin configuration
VH1 and VH2 Undervoltage Lockout (UVLO)	2.59	2.71	2.83	V	Voltage below which the device turns off
VH1 and VH2 UVLO Hysteresis		111		mV	Voltage, above the UVLO voltage level, at which the device turns on
VH1 and VH2 Arbitration Hysteresis	90			mv	VH1 and VH2 = 3.3 V
	317			mV	VH1 and VH2 = 5 V
	987			mV	VH1 and VH2 = 12 V
AVDD_CAP	3.2	3.3	3.355	V	Regulated AVDD_CAP low dropout (LDO) output; VH1 and VH2 > 3.6 V
DVDD_CAP	1.79	1.82	1.85	V	Regulated DVDD_CAP LDO output
SUPPLY FAULT DETECTORS					
VHx Pins					
Input Voltage Range	0		15	V	
Input Impedance					
VH1 and VH2		41		kΩ	
VH3 and VH4		153		kΩ	
VPx Pins					
Input Voltage Range	0		5	V	
Input Impedance		62		kΩ	
VPx Pins, Differential (Odd and Next Even) Common-Mode Voltage Offset	−100		+100	mV	Maximum voltage difference from VP2, VP4, VP6, VP8, VP10, and VP12 to GND in differential sense mode
Threshold Resolution		8		Bits	
Digital Glitch Filter		2		µs	Minimum programmable filter length
		100		µs	Maximum programmable filter length
PROGRAMMABLE DRIVER INPUT/OUTPUTS					
Input Voltage, High (V_{IH})	1.4			V	
Input Voltage, Low (V_{IL})			0.6	V	
Output Voltage, High (V_{OH})	2.8		AVDD_CAP	V	I_{OH} = 0.5 mA
Output Voltage, Low (V_{OL})	0		0.50	V	I_{OL} = 20 mA
Output Current, High (I_{OH})			500	µA	Maximum source current per PDIOx pin
Source Current (I_{SOURCE})			3	mA	Maximum total source for all PDIOx pins
Output Current, Low (I_{OL})			20	mA	Maximum sink current per PDIOx pin
Sink Current (I_{SINK})			60	mA	Maximum total sink for all PDIOx pins
Pull-Up Resistance ($R_{PULL-UP}$)		20		kΩ	Internal pull-up
Pull-Up Resistance ($R_{PULL-DOWN}$)		20		kΩ	Internal pull-down
Tristate Leakage Current			9	µA	V_{PDIO} = 21 V
			1	µA	V_{PDIO} < 3.6 V
GPIOs					
V_{IH}	1.63			V	
V_{IL}			0.8	V	
V_{OH}	2.6		AVDD_CAP	V	I_{OH} = 4 mA
V_{OL}	0		0.50	V	I_{OL} = 4 mA
I_{OH}			4	mA	Maximum source current per GPIOx pin
I_{SOURCE}			12	mA	Maximum total source for all GPIOx pins
I_{OL}			4	mA	Maximum sink current per GPIOx pin
I_{SINK}			12	mA	Maximum total sink for all GPIOx pins

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
Tristate Leakage Current			1	µA	
BUFFERED VOLTAGE OUTPUT DACs					
Resolution		8		Bits	
Code 0x7F Output Voltage					
0.2 V to 0.8 V	0.501	0.506	0.516	V	
0.3 V to 0.9 V	0.603	0.607	0.618	V	
0.5 V to 1.1 V	0.804	0.809	0.820	V	
0.7 V to 1.3 V	1.005	1.011	1.021	V	
0.95 V to 1.55 V	1.256	1.264	1.273	V	
Output Voltage Range		606		mV	Same range, independent of center point
LSB Step Size		2.376		mV	
DAC Supply Currents			3	mA	Maximum total source for all DAC pins
DAC Leakage Current			1	µA	
Maximum Load Current					
Source			0.25	mA	
Sink			0.25	mA	
Maximum Load Capacitance			50	pF	
Settling Time to 50 pF Load			2	µs	
ADC					
Signal Range	0		V_{REF}	V	
Resolution		12		Bits	
Round Robin Time		5		ms	
SERIAL BUS DIGITAL INPUTS (SCL, SDA)					
Input High Voltage, V_{IH}	2.1			V	
Input Low Voltage, V_{IL}			0.8	V	
Output Low Voltage, V_{OL}			0.4	V	
Clock Frequency, f_{SCLK}			400	kHz	
Bus Free Time, t_{BUF}	1.3			µs	
Start Setup Time, $t_{SU;STA}$	0.6			µs	
Stop Setup Time, $t_{SU;STO}$	0.6			µs	
Start Hold Time, $t_{HD;STA}$	0.6			µs	
SCL Low Time, t_{LOW}	1.3			µs	
SCL Low Timeout, $t_{LOW:MAX}$			35	ms	PMBus resets if this value is exceeded
SCL High Time, t_{HIGH}	0.6		50	µs	
SCL, SDA Rise Time, t_R			300	ns	
SCL, SDA Fall Time, t_F			300	ns	
Data Setup Time, $t_{SU;DAT}$	100			ns	
Data Hold Time, $t_{HD;DAT}$	300			ns	
ADDR PIN PULL-UP CURRENT	45	50	55	µA	
SYNC					
Input High Voltage, V_{IH}	2.1			V	
Input Low Voltage, V_{IL}			0.8	V	
Output High Voltage, V_{OH}	2.6		AVDD_CAP	V	
Output Low Voltage, V_{OL}			0.5	V	
Clock Frequency, f_{SCLK}		32.768		kHz	
INTERDEVICES BUS (IDB)					
Input High Voltage, V_{IH}	2.1			V	
Input Low Voltage, V_{IL}			0.8	V	
Output Low Voltage, V_{OL}			0.4	V	
Clock Frequency, f_{SCLK}			1	MHz	

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
REFERENCE OUTPUT					
Reference Output Voltage (V_{REF})	2.006	2.020	2.031	V	V_{REF}, no load
Load Regulation		−0.25		mV	Sourcing current, $I_{DACxMAX}$ = −100 µA
		0.25		mV	Sinking current, $I_{DACxMAX}$ = 100 µA
Minimum Load Capacitance	1			µF	Capacitor required for decoupling, stability
TEMPERATURE SHUTDOWN (TSD)					
TSD Rising		150		°C	
TSD Hysteresis		20		°C	

T_J = −40°C to +85°C, VH1 and VH2 > 3 V, unless otherwise noted.

Table 5.

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
EEPROM RELIABILITY					
Endurance[1]	10,000			Cycles	T_J = 85°C
Data Retention[2, 3]	10			Years	T_J = 85°C

[1] Endurance is qualified as per JEDEC Standard 22, Method A117.
[2] Retention lifetime equivalent at junction temperature (T_J) = 85°C as per JEDEC Standard 22, Method A117. Retention lifetime derates with junction temperature.
[3] For temperatures above 85°C. Refer to the Refresh section and Acceleration Factor section.

ABSOLUTE MAXIMUM RATINGS

Table 6.

Parameter	Rating
VHx, PDIOx to GND	21 V
VPx, AVDD_CAP to GND	5.5 V
DACx to GND	3.6 V
REFOUT to GND	3.6 V
ADDR to GND	3.6 V
REFGND, EPAD to GND	−0.3 V to +0.3 V
All Other Pins to GND	3.6 V
Maximum Junction Temperature (T_J max)	150°C
Storage Temperature Range[1]	−65°C to +125°C
ESD Rating, All Pins	
Charged Device Model	750 V
Human Body Model	2000 V

[1] See the Acceleration Factor section.

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

THERMAL RESISTANCE

Thermal performance is directly linked to printed circuit board (PCB) design and operating environment. Careful attention to PCB thermal design is required.

θ_{JA} is the natural convection junction to ambient thermal resistance measured in a one cubic foot sealed enclosure. The thermal resistance values specified in Table 7 are calculated based on JEDEC specs and must be used in compliance with JESD51-12.

Table 7. Thermal Resistance[1]

Package Type	θ_{JA}[2]	θ_{JC_BOTTOM}[3, 4]	Ψ_{JT}	Ψ_{JB}	Unit
CP-64-15	24.2	0.6	0.1	3.6	°C/W

[1] The values in Table 7 are calculated based on standard JEDEC test conditions, unless otherwise specified
[2] θ_{JA} is simulated using a 2S2P PCB with 49 standard JEDEC vias.
[3] For the θ_{JC_BOTTOM} test, 100 µm TIM is used. TIM is assumed to have 3.6 W/mK.
[4] θ_{JC_BOTTOM} is simulated using a 1S0P PCB with 49 standard JEDEC vias.

ESD CAUTION

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

PIN CONFIGURATION AND FUNCTION DESCRIPTIONS

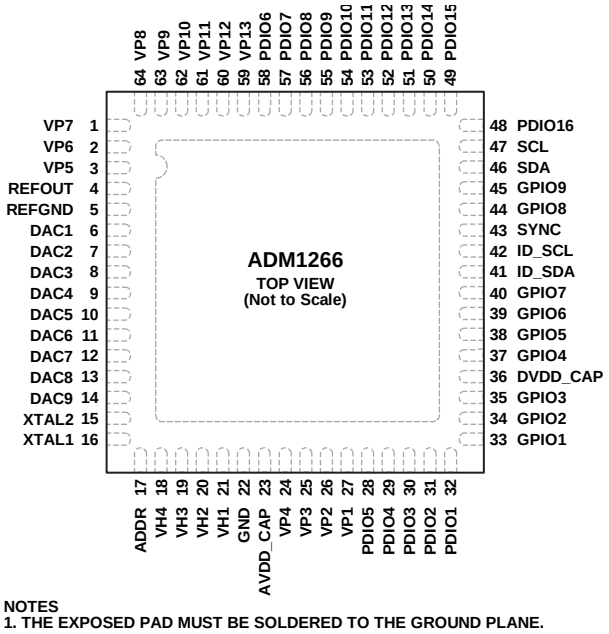


Figure 3. Pin Configuration

Table 8. Pin Function Descriptions

Pin No.	Mnemonic[1]	Description
1 to 3, 24 to 27, 59 to 64	VP1 to VP13	Low Voltage Inputs to Supply Fault Detectors. These pins monitor voltages of up to 5 V.
4	REFOUT	Reference Output. A capacitor must be connected between this pin and REFGND. A 2.2 µF capacitor is recommended for this purpose.
5	REFGND	Ground Return for On-Chip Reference Circuits. Star connect this ground to the GND pin.
6 to 14	DAC1 to DAC9	Voltage Output DACs. These DACs can be used for margining and trimming the supply rails.
15	XTAL2	Crystal Input 2. This pin is configured as the 32.768 kHz crystal input. It can also be configured as high impedance.
16	XTAL1	Crystal Input 1. This pin is configured as the 32.768 kHz crystal input. It can also be configured as high impedance.
17	ADDR	PMBus Address Select Resistor. A resistor to GND sets one of 16 addresses.
18 to 21	VH1 to VH4	High Voltage Input to Supply Fault Detectors. These pins can monitor voltages of up to 15 V. The highest voltage from VH1 to VH2 powers the ADM1266 via the supply arbitrator.
22	GND	Supply Ground.
23	AVDD_CAP	Analog Supply Voltage. Linearly regulated from the higher voltage of the VH1 and VH2 pins to 3.3 V (typical). Note that a capacitor must be connected between this pin and GND. A 68 µF or larger capacitor is recommended for this purpose
28 to 32, 48 to 58	PDIO1 to PDIO16	Programmable Driver Inputs/Outputs. These pins default to a 20 kΩ pull-down resistor at power-up.
33 to 35, 37 to 40, 44, 45	GPIO1 to GPIO9	General-Purpose Inputs/Outputs. The default start-up condition of these pins is high impedance.
36	DVDD_CAP	Digital Supply Voltage (1.8 V Typical). Note that a capacitor must be connected between this pin and GND. A 2.2 µF capacitor (or larger), type X5R/10 V (or better), size 0402 (or larger) is recommended for this purpose.
41	ID_SDA	Interdevice Communications Bus Data Signal. ID_SDA is a bidirectional, open-drain pin that requires an external pull-up resistor of 2.2 kΩ. It is recommended that the pull-up source be AVDD_CAP. The default start-up condition of this pin is high impedance.
42	ID_SCL	Interdevice Communications Bus Clock Signal. ID_SCL is a bidirectional, open-drain pin that requires an external pull-up resistor of 2.2 kΩ. It is recommended that the pull-up source be AVDD_CAP. The default start-up condition of this pin is high impedance.

Pin No.	Mnemonic[1]	Description
43	SYNC	32.768 kHz Clock Timing Synchronization Input/Output. This pin can be used to provide a clock signal to other ADM1266 devices on the board, or a 32.768 kHz input signal can be provided to the ADM1266 from an external clock source. The default start-up condition of this pin is high impedance.
46	SDA	PMBus Data. SDA is a bidirectional, open-drain pin that requires an external pull-up resistance of 2.2 kΩ.
47	SCL	PMBus Clock. SCL is bidirectional, open-drain pin that requires an external pull-up resistance of 2.2 kΩ.
	EPAD	Exposed Pad. The exposed pad must be soldered to the ground plane.

[1] Connect all unused pins to GND.

TYPICAL PERFORMANCE CHARACTERISTICS

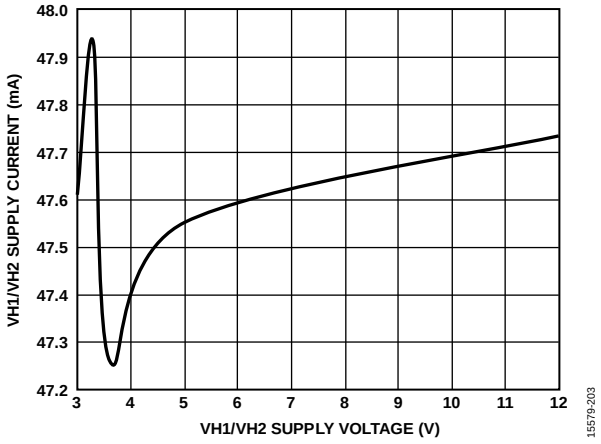


Figure 4. VH1/VH2 Current vs. VH1/VH2 Voltage with a 42 mA Load on AVDD_CAP

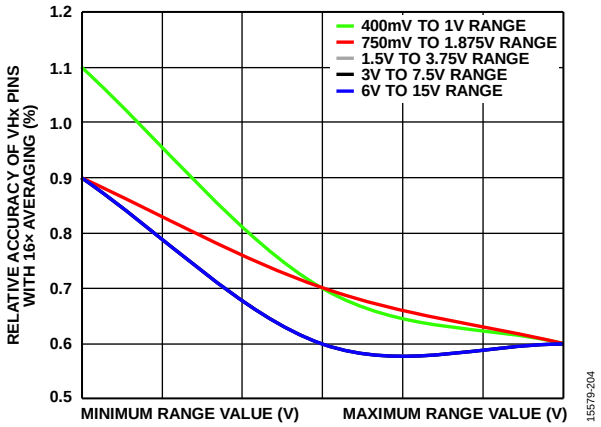


Figure 5. Relative Accuracy of VHx Pins with 16x Averaging (%) Across the Supply Fault Detector Range

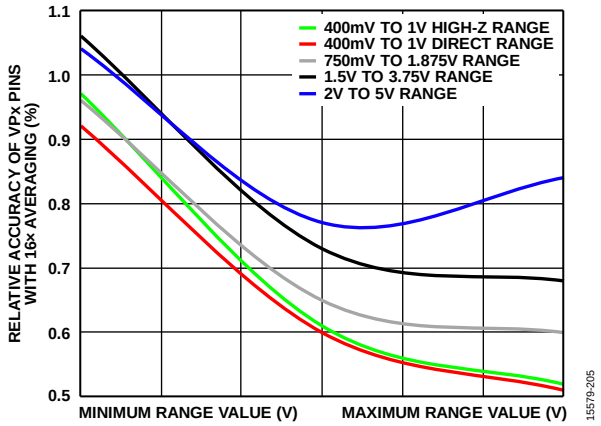


Figure 6. Relative Accuracy of VPx Pins with 16× Averaging (%) Across the Supply Fault Detector Range

THEORY OF OPERATION

POWERING THE ADM1266

The ADM1266 is powered from the highest voltage input on VH1 or VH2. This technique, called supply arbitration, offers improved redundancy because the device is not dependent on any one particular voltage rail to keep it operational. The AVDD_CAP arbitrator on the device chooses the supply to use. The arbitrator can be considered an OR'ing of two LDO regulators together. A supply comparator chooses the highest input to provide the on-chip supply. It is not recommended to connect both VH1 and VH2 to the same voltage levels because the ripple on the two voltages may cause the arbitrator circuit to constantly toggle. This architecture has minimal voltage drop, resulting in the ability to power the ADM1266 from a supply as low as 3 V. A 10 µF bypass capacitor and 0.1 µF decoupling capacitors are needed on both the VH1 and VH2 pins. Additionally, these capacitors ensure a successful arbitration when switching from VH1 to VH2 and vice versa. In a system with multiple ADM1266 devices, it is important that all the devices are powered from the same voltage rail.

An external capacitor from AVDD_CAP to GND is required to decouple the on-chip supply from noise, as shown in Figure 7. The capacitor has another use during brownouts (momentary loss of power). Under these conditions, when all the input supplies (VHx pins) fall below AVDD_CAP, the LDO regulators immediately turn off so that the VHx power supply does not pull AVDD_CAP down. The AVDD_CAP capacitor can then act as a reservoir to keep the ADM1266 active until the next highest supply takes over the powering of the device. A capacitor with a minimum value of 68 µF is recommended for this reservoir/decoupling function.

If all supplies fail, the value of the AVDD_CAP capacitor can be increased if it is necessary to guarantee that a complete fault record is written into EEPROM.

The VHx input pins can accommodate supplies of up to 15 V, which allows the ADM1266 to be powered using a 12 V backplane supply. In cases where this 12 V supply is hot swapped, it is recommended that the ADM1266 not be connected directly to the supply. Take suitable precautions, such as the use of a hot swap controller or RC filter network, to protect the device from transients that may cause damage during hot swap events.

When two or more supplies are within the VH1/VH2 arbitration hysteresis value of each other, the supply that first takes control of AVDD_CAP keeps control. For example, if VH1 is connected to a 5.0 V supply, AVDD_CAP powers up to 3.3 V (typical) through VH1. If VH2 is then connected to another 5.0 V supply, VH1 still powers the device, unless VH2 goes approximately 317 mV higher than VH1.

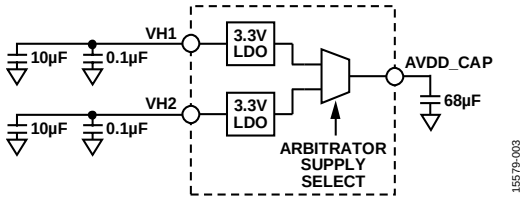


Figure 7. AVDD_CAP Arbitrator Operation

During power-up, the ADM1266 checks the main boot loader, the main firmware, the main configuration, and the backup configuration to ensure that the data in these sections is correct. If multiple devices are connected on the same IDB, all the devices individually check the main and backup configurations and send this information back to the master. Then, the master decides to run the correct configuration. The boot up time from VH1 or VH2 crossing 3 V to the device ready to execute State 1 varies based on the size of the configuration. On the top right corner of the GUI, an icon displays the size of the configuration memory in a percentage. Use this percentage in the following equation to calculate the boot up time:

$$\textit{Typical Boot Up Time}\ (\text{ms}) = 1.142 \times \textit{Percentage} + 192$$

For example, if 27% of the memory is used, then,

$$\textit{Boot Up Time} = 1.142 \times 27 + 192$$

$$\textit{Boot Up Time} = 223\ \text{ms}$$

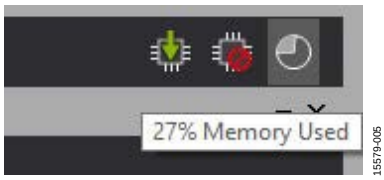


Figure 8. GUI Icon Showing Configuration Memory Size

INPUTS

Supply Fault Detectors

The ADM1266 has 17 programmable supply fault detector (SFD) inputs. These dedicated inputs are labeled VHx (VH1 to VH4) and VPx (VP1 to VP13). The ADM1266 is also capable of making precision differential voltage measurements on the VPx pins (the exception is that VP13 cannot be used for differential measurements). One differential measurement requires two VPx pins. The odd numbered VPx pin (for example, VP1) must always be the greater voltage. The next corresponding even number VPx pin (for example, VP2) is used for that differential measurement. Both differential VPx pins must have the same input range selections. The SFD for the odd numbered VPx pin responds to the differential measurement. Figure 9 shows the arrangement of the pins. Each SFD input can be configured to detect an undervoltage (UV) fault (the input voltage drops below a preprogrammed value), or an overvoltage (OV) fault (the input voltage rises above a preprogrammed value). A programmable (up to 100 µs) glitch filter allows the user to remove any spurious transitions such as supply bounce at turn on.

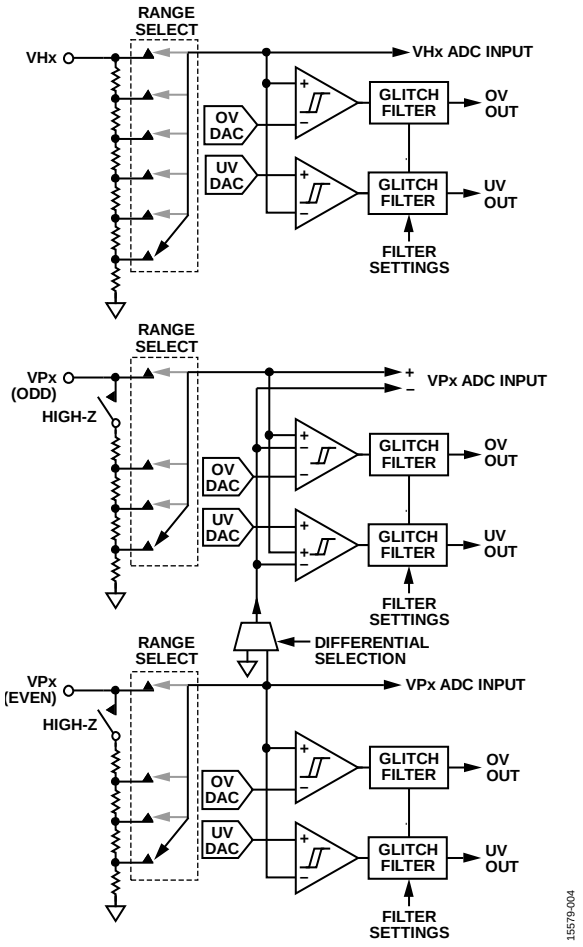


Figure 9. Supply Fault Detectors

The voltage range limits for threshold settings are

- 0.4 V to 1.0 V
- 0.75 V to 1.875 V
- 1.5 V to 3.75 V
- 2.0 V to 5.0 V (VPx pins only)
- 3.0 V to 7.5 V (VHx pins only)
- 6.0 V to 15.0 V (VHx pins only)

When connecting directly to the voltage source, a 100 Ω resistor in series is recommended to avoid any latch-ups on the VHx and VPx pins. VH1 and VH2 are supply pins and do not need the 100 Ω resistor in series.

Input Comparator Hysteresis

The UV and OV comparators shown in Figure 9 are always monitoring and sensing the voltage on VHx and VPx. To avoid chatter (multiple transitions when the input is very close to the set threshold level), these comparators have digitally programmable hysteresis. The hysteresis is added after a supply voltage goes out of tolerance. Therefore, the user can program the amount above the UV threshold to which the input must rise before a UV fault is deasserted. Similarly, the user can program the amount below the OV threshold to which an input must fall before an OV fault is deasserted.

Glitch Filter

The ADM1266 has a dedicated digital glitch filter at the output of each comparator. For the fault to trigger, the comparator must remain set for the time greater than the programmed glitch filter time. This time can be programmed from 2 µs to 100 µs and is used for filtering any transient noises that may occur on the VHx and VPx pins.

Using External Resistor Dividers

External resistor dividers can be used to sense higher voltages or to achieve higher accuracy. When using an external resistor divider, select the 0.4 V to 1 V high impedance range on the VPx pins. It is recommended that the resistor divider be sized such that, under nominal conditions, there is 0.7 V at the VPx pins to provide the highest range for the OV and UV settings.

The size of the external resistor divider can be input into the device using the VOUT_SCALE_MONITOR command (Register 0x2A).

Warnings

The UV and OV warnings are generated by comparing the VOUT_OV_WARN_LIMIT (Register 0x42) and (Register 0x43) VOUT_UV_WARN_LIMIT with the reading from the ADC. Because the ADC round robin time is 5 ms, the maximum delay from the warning occurring to the device detecting it is 5 ms.

Warnings are not sent to the sequence engine and cannot be used to trigger events in the state machine. Instead, the warnings are sent to the logic block and can be used to assert/deassert PDIOs and GPIOs.

Threshold Settings

The UV and OV thresholds are set using the commands in Table 9.

Table 9. UV and OV Threshold Commands

Command	Register	Description
VOUT_MODE	0x20	Used for setting the exponent for linear PMBus calculations for the following commands
VOUT_OV_WARN_LIMIT, VOUT_UV_WARN_LIMIT	0x42, 0x43	Used for setting mantissa for linear PMBus calculation of warning limits
VOUT_OV_FAULT_LIMIT, VOUT_UV_FAULT_LIMIT	0x40, 0x44	Used for setting mantissa for linear PMBus calculation of fault limits
VOUT_OV_HYST_LIMIT, VOUT_UV_HYST_LIMIT	0xD0, 0xD1	Used for setting mantissa for linear PMBus calculation of hysteresis limits

Voltage Readback and Status

The ADM1266 has an on-board, 12-bit accurate ADC for voltage readback over the PMBus using the READ_VOUT command (Register 0x8B). Inputs to the ADC consist of the 17 SFD inputs (VHx and VPx pins). The inputs to the ADC come from the back of the input attenuators on the VPx and VHx pins, as shown in Figure 9.

Supplies can also be connected to the input pins purely for ADC readback, even though these pins may go above the expected supervisory range limits (but not above the absolute maximum ratings on these pins). For example, a 1.5 V supply connected to the VP1 pin on the lowest range (0.4 V to 1.0 V) can be correctly read out as on the ADC, but it always sits above any supervisory limits that can be set on that pin.

Voltage Trimming

Use the VOUT_TRIM PMBus command (Register 0x22) to add an additional offset trim to all the threshold settings and for voltage readback. This command can be used to remove any inaccuracies generated by the external components.

PROGRAMMABLE DRIVER INPUT/OUTPUTS

Supply Sequencing Through Configurable Output Drivers

The programmable driver input/output (PDIOx) pins are typically used to drive logic enables on external supplies or as digital inputs into the sequencing engine. The sequence in which the PDIOx pins are asserted (and, therefore, the supplies are turned on) is controlled by the SE firmware. The SE determines the action that is taken with the PDIOx pins, based on the condition of the ADM1266 inputs. Therefore, the PDIOx pins can be set up to assert when the SFDs are in tolerance and no faults are received from any of the inputs of the device.

The PDIOx pins can also be used to provide a power-good signal, when all the SFDs are in tolerance, or a reset output if one of the SFDs goes out of specification (this power-good signal can be used as a status signal for a DSP, FPGA, or other microcontroller). The open-drain nature of the PDIOx pins also allows them to be used to drive status LEDs.

The output stage of the PDIOx pins has programmable pull-up and pull-down options. The PDIOx pins can be programmed as follows:

- Push/pull to AVDD_CAP. When using a PDIOx pin in a push/pull configuration, a 20 kΩ resistor in series is recommended to limit the current drawn from the PDIOx pin.
- Open drain with an internal 20 kΩ pull-up resistor to AVDD_CAP.
- Open drain with an external pull-up resistor up to 20 V.
- Open source with an internal 20 kΩ pull-down resistor to GND.
- Open source with external pull-down resistor to GND.
- High-Z.
- Internal 20 kΩ pull-up resistor to AVDD_CAP.
- Internal 20 kΩ pull-down resistor to GND.

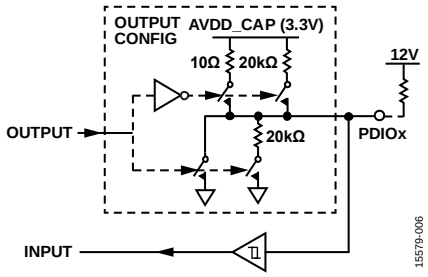


Figure 10. Programmable Driver Input/Output

Default Output Configuration

All of the internal registers in an unprogrammed ADM1266 device from the factory are set to 0. Because of this default setting, the PDIOx pins are pulled to GND by a weak (20 kΩ), on-chip, pull-down resistor.

As the input supply to the ADM1266 ramps up on VHx, all PDIOx pins behave as follows:

- Input supply = 0 V to 1.5 V. The PDIOx pins are high impedance.
- Input supply = 1.5 V to 2.7 V. The PDIOx pins are pulled to GND by a weak (20 kΩ), on-chip, pull-down resistor.
- Supply > 2.7 V. Factory programmed devices continue to pull all PDIOx pins to GND by a weak (20 kΩ), on-chip, pull-down resistor. Programmed devices download current EEPROM configuration data, and the programmed setup is latched. The PDIOx pin then goes to the state demanded by the configuration. This configuration provides a known condition for the PDIOx pins during power-up. If the pin is configured to output, after downloading the configuration and before the sequence is run, the ADM1266 senses the voltage on the pin and drives the pin to the same level as the voltage sensed on the pin.

The internal pull-down resistor can be overdriven with an external pull-up resistor of suitable value tied from the PDIOx pin to the required pull-up voltage. The 20 kΩ resistor must be accounted for when calculating a suitable value. For example, if PDIOx must be pulled up to 3.3 V, and 5 V is available as an external supply, the pull-up resistor (R_{UP}) value is given by

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

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

PDIOx as Inputs

The PDIOx pins can be configured as inputs to trigger the sequence engine and cause events in the state machine. The PDIOx pins can also be used as inputs to the logic block. They have a dedicated glitch filter that filters out any transient noises on the signals. The glitch filter can be programmed to values from 500 ns to 100 µs.

Additionally, these pins can be configured as inputs and outputs at the same time, which is particularly useful for multiple devices monitoring and controlling the same signal.

GENERAL-PURPOSE INPUT/OUTPUTS

There are nine dedicated pins that serve as GPIOs. Each pin can be configured as an input, an output, or both. The GPIOs have no internal glitch filter. The default start-up condition of the GPIOs is high impedance.

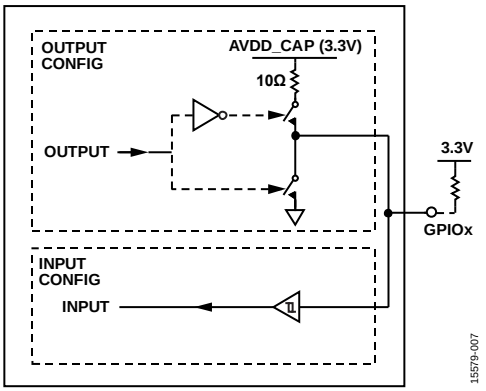


Figure 11. GPIOs

In input only mode, the GPIOs can be used to trigger an action in the sequence engine, or as inputs to logic block.

In output only mode, the GPIO pins can be configured in a push/pull configuration or as an open drain with an external pull-up resistor. In push/pull mode, the GPIOs are internally pulled up to 3.3 V. When using a GPIOx pin in a push/pull configuration, a 20 kΩ resistor in series is recommended to limit the current drawn from the GPIOx pin. In open-drain configuration, the GPIOs are pulled up using an external resistor up to 3.3 V. The output status of the GPIOs can be driven from the sequence engine or logic block.

The GPIOs in output mode can be used as a power-good or fault signal.

In input/output mode, the GPIOs can only be configured in open-drain configuration with an external pull-up resistor. In this mode, multiple GPIOs across several devices are OR'ed together to create a signal.

When disabled, the GPIOx pin is high impedance.

The GPIOs are configured using the GPIO_CONFIGURATION command (Register 0xE1).

SEQUENCING ENGINE (SE)

OVERVIEW

The ADM1266 SE provides the user with powerful and flexible control for sequencing multiple power rails. The SE implements state machine control of the PDIOx and GPIOx outputs, with state changes conditional on input events driven by VHx, VPx, PDIO, GPIOs, timers, and variables. The SE programs can enable complex control of boards such as power-up and power-down sequence control, fault event handling, and interrupt generation.

POWER-UP AND STATE 0

After the EEPROM data is downloaded, the ARM controller starts execution of the core sequencer and transitions to the following tasks, including but not limited to

- Performing a roll call of all ADM1266 devices present if more than 16 voltage rails are sequenced using more than one ADM1266 device.
- Checking the CRC status of the main and backup configurations of all devices.
- Synchronization of the black box ID between multiple ADM1266 devices.
- Waiting for a ready signal from all ADM1266 devices on the IDB bus to enter State 1.

If a fault is present in any of these tasks, the SE halts and terminates immediately. A power cycle or software reset using GO_COMMAND (Register 0xD8) can restart the sequence engine.

If all the operations of State 0 are successful, the device enters State 1 where sequencing beings.

STATE SECTIONS

To maintain maximum flexibility and ease of use, the SE is divided into two sections: enter actions and loop actions.

Enter Actions

The enter actions section consists of actions that are used to initialize the system or a state. Examples range from starting a timer to setting a PDIO. The actions programmed in this subsection are executed only once before entering loop actions.

Loop Actions

In the loop actions section, the SE provides monitoring and adjustment functions. After executing the enter actions, the ADM1266 transitions and executes the actions in the loop actions section. The device continues to execute these actions in a loop, until it encounters a go to action. When the device encounters a go to action, the device aborts the rest of the actions in the loop actions and proceeds to the next state.

Whenever an interrupt is generated because of a fault or a logic change, the SE is triggered to go to the first action in the loop actions section and starts executing the actions. By ordering the different actions in the loop actions, the user can set a priority on when the actions are executed to minimize any delays.

ACTION TYPES

The user can configure multiple actions. These actions are broadly classified into three categories: set actions, monitor actions, and special actions.

Set Actions

Set actions set the output of a PDIO or GPIO. These actions can also be used to set or reset variables and timers. These actions can be configured in the enter actions and loop actions sections of the state.

Monitor Actions

The fault monitoring action types are used to read the status of the VHx, VPx, PDIOx, and GPIOx pins. The monitoring function is extended to include the monitoring of the status of variables and timers as well. The individual status is compared to a threshold to determine the outcome of the action as true or false. When the outcome is determined, an action is undertaken.

To expand on the flexibility of the sequence engine over multiple rails, the user is allowed to program and monitor any logical combination of rails, timers, PDIO, and GPIOs to create a fault state.

Special Actions

Two special actions are available in the ADM1266. Use the go to action to proceed to a preprogrammed state of the SE. Use the black box action type to capture a snapshot of the status of all the pins and write it to the EEPROM. Refer to the Black Box (EEPROM) Fault Recording section for more details.

PARALLEL OPERATION AND INTERDEVICE BUS

If more than 16 rails are to be sequenced, multiple ADM1266 devices can be connected in parallel. Communication between the ADM1266 devices is facilitated by the IDB that operates at 1 MHz maximum, and follows the I^2C protocol. The IDB is a private bus and sends Analog Devices proprietary messages. A maximum of 16 ADM1266 devices can be connected on the IDB. One device is configured as a master, and the other devices are configured as slaves. All the slaves communicate their current status back to the master; the master, based on the user configuration and the status of all the devices, broadcasts to all the slaves the new state that they need to go to.

STATES

The user can configure up to 1023 states to form their desired state machine. The user can create their virtual state machine using the Analog Devices Power Studio™ software. If there is only one device, the virtual state machine and the state machine configured in the device are identical. If multiple devices are connected together, the software compiles the virtual state machine and programs each device with the corresponding state machine and IDB messages. This procedure is transparent to the user, meaning that the user does not need to individually create a state machine for each ADM1266 device. After the user creates the virtual state machine in the software, the software automatically creates the corresponding state machine for each device.

For example, the user creates a virtual state machine in the software consisting of 20 states. For a single device, the device has 20 states. For multiple devices, each device has 20 states. All the devices move through the different states in synchronization and work in parallel.

Breakpoints and Debug Mode

During development, the user can set the ADM1266 to be in debug mode. The user can set breakpoints for each of the 1023 states as desired. When the ADM1266 enters a state, if the breakpoint for the state is enabled, the SE pauses at the start of the state. The SE can resume by sending a start message using GO_COMMAND (Register 0xD8). When resuming, the SE executes the actions in that state, which is helpful in pausing the SE at the desired breakpoints without modifying the configured state machine. In normal mode, the breakpoints are ignored.

Stop, Start, and Reset

At any point, GO_COMMAND (Register 0xD8) can be issued to the ADM1266 to start or stop the SE. This command can also be used to reset the state machine to State 0. By default at power-up, the SE is in start mode and does not need a start command. If multiple devices are connected, GO_COMMAND (Register 0xD8) must be sent to all the devices as part of the group command protocol.

SUPPLY MARGINING

OVERVIEW

Due to tolerances of circuit components, input voltage ranges, and variations in reference voltages, load, and temperature, for example, the output voltage of the dc-to-dc converter deviates from the nominal setpoint value. The worst case conditions need to be simulated on the power supply during manufacturing and production, and the corner conditions can be measured to check for an out-of-limit condition. Additionally, the accuracy of the output voltage is also a critical factor for some applications and must be tightly maintained when the tolerance of the output voltage resistive divider is large (see Figure 12).

The procedure of ensuring this output voltage regulation is called margining (or voltage margining). This voltage margining is accomplished by the use of an on-chip DAC that pulls up/down the feedback node of the error amplifier of the power controller. A typical application circuit for margining is shown in Figure 12. Using nodal analysis and basic circuit theory, modifying the feedback node changes the output voltage and, typically, there is an inversely proportional relationship between the output of the DAC and the output voltage.

Because the ADM1266 has nine internal DACs, margining is possible on nine rails.

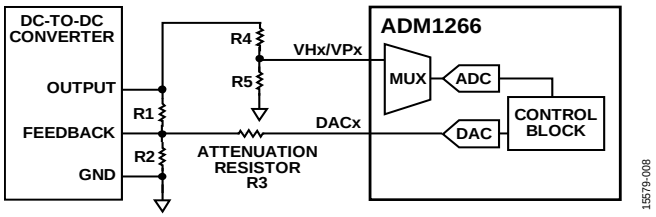


Figure 12. Typical Application Circuit for Margining

Margining can be performed two ways: open-loop margining and closed-loop margining.

The margining is actuated by a DAC and a series resistor that are connected to the feedback node of the power supply controller (see Figure 12). The equivalent change in output voltage can be determined by the following equations:

$$\frac{V_{DAC} - V_{FB}}{R3} + \frac{V_{FB}}{R2} = \frac{V_{OUT} - V_{FB}}{R1} \quad (1)$$

$$V_{FB} = V_{OUT} \times \frac{R2}{R1 + R2} \quad (2)$$

Subtracting the two equations yields

$$\Delta V_{OUT} = \frac{R1}{R3}\left(V_{FB} - V_{DAC}\right)$$

Table 10. DAC_CODE_CONFIGURATION[3:1], Register 0xEB, DAC Ranges

Bits[3:1]	Midcode Voltage (V)	Minimum Voltage Output (V)	Maximum Voltage Output (V)
0x00= 3'b000	0.506	0.202	0.808
0x01= 3'b001	0.607	0.303	0.909
0x02= 3'b010	0.809	0.505	1.111
0x03= 3'b011	1.011	0.707	1.313
0x04= 3'b100	1.263	0.959	1.565

Open-Loop Margining

In open-loop margining, the user has direct access to the internal DACs. The DAC forces a voltage on the feedback node of the power controller, which causes a deviation in the output voltage. Typical values for this test are ±1%, ±2.5%, ±5%, ±7.5%, and ±10% of the nominal output voltage. The user can program up to 16 preset values, and can use a pointer command to instruct the device regarding the value that must be loaded into the DAC. Both the preset values and the value of the pointer can be saved into the memory. At power-up, the device downloads the settings and configures the DAC automatically.

Closed-Loop Margining

Closed-loop margining is the preferred method of margining. It determines the ability of the power supply to regulate the output under extreme corner conditions. It is recommended to use the Power Studio software because it provides all related calculations for resistors and parameters for this feature.

The ADM1266 uses the *PMBus Power System Management Protocol Specification* (Revision 1.2, September 6, 2010) command set that offers the margining commands through the following commands:

- OPERATION (Register 0x01)
- VOUT_MARGIN_HIGH (Register 0x25)
- VOUT_MARGIN_LOW (Register 0x26)
- VOUT_SCALE_LOOP (Register 0x29)
- VOUT_COMMAND (Register 0x21)
- VOUT_MARGIN_LOOP (Register 0xDA)
- MARGIN_CONFIGURATION (Register 0xDB)

These commands enable margining, position the output voltage at either the high or low value, monitor the feedback node, and set the ratio of R1 and R3.

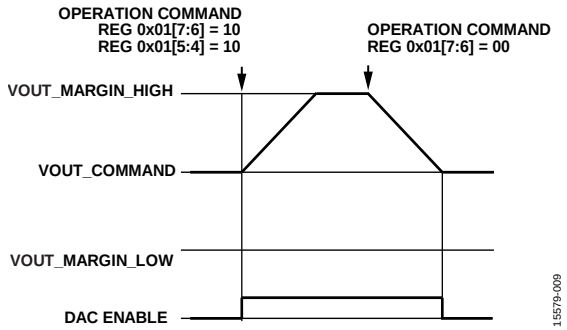


Figure 13. Margining Example 1

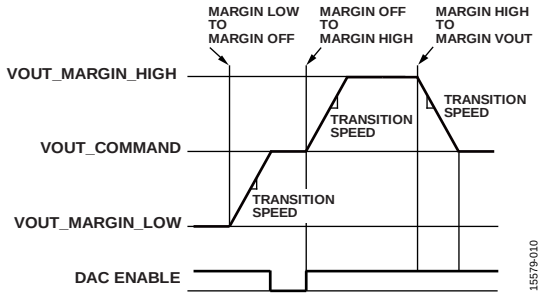


Figure 14. Margining Example 2

Figure 13 and Figure 14 show examples of margining. When margining is turned off, the DAC returns the output voltage to the nominal level at the transition rate (0xDB) and then enters a high-Z state.

To enable a smooth start of the margining process, the ADM1266 uses a smart connect mode. Smart connect mode calculates the DAC code that is equal to the feedback node such that there is no current flowing in Resistor R3 (see Figure 12). Smart connect mode prevents any sudden glitches in the output voltage. Following smart connect mode, the DAC code is changed as per the margin command.

The closed-loop margining process differs from open-loop margining with the following differences:

- The DAC is continuously repositioned until the 16 averages of the high accuracy ADC monitoring the output rail result in a value that equals the VOUT_MARGIN_x command. This repositioning ensures that the output voltage does indeed reach the command value. The output voltage is sampled using the ADC at a rate of 5 ms. Therefore, 16 average readings complete in approximately 80 ms.
- Whenever a margin command is issued, the DAC changes its output based on the rate programmed in Register 0xDB. Therefore, the output of the power supply rail also transitions at this rate. All the DACs are controlled using the same rate.

During the closed-loop margining process, the UV and OV faults are active and take the appropriate programmed action. The DAC is disabled (high-Z state) immediately and does not perform a soft disconnect.

One Shot Mode and Continuous Mode

The ADM1266 offers two modes of operation in closed-loop margining: one shot and continuous mode.

In one shot mode, the process of closed-loop margining (see the Closed-Loop Margining section) occurs once, that is, the DAC changes the output voltage as per the margin command and remains fixed, and no further changes in the DAC output are allowed. In continuous mode, this process occurs continuously. In one shot mode, a new margin command must be issued to change the DAC output.

Continuous mode can be used for increasing the accuracy of a power supply that suffers from wide tolerances in component or reference voltage levels. Use this method when the ADC accuracy is greater than the accuracy of the external components.

The Power Studio software provides all the extensive configurations, from selecting the DAC range to margining commands.

Closed-Loop Margining Enable Timings

If the rail is in steady state and the ADM1266 receives the operation command to go from the margin being off to the servo VOUT_COMMAND (margin high or margin low), the ADM1266 enables closed-loop margining. The ADM1266 also performs the smart connection, waiting 5 ms for an updated ADC reading before starting the ramp to obtain the desired voltage level.

If the rail is in a steady state and the ADM1266 receives the operation command to go from the servo VOUT_COMMAND to margin high (margin low or margin off), the ADM1266 immediately starts the ramp to obtain the desired voltage level. This process is also true when the starting point is margin high or margin low.

If the device is programmed to wake up and immediately start to the servo VOUT_COMMAND (margin high or margin low), the ADM1266 enables the rail. After the rail clears the UV threshold, the ADM1266 enables closed-loop margining and performs smart connect after 20 ms to 25 ms. Then, the ADM1266 waits 5 ms to obtain an updated ADC reading and starts the ramp to obtain the desired voltage level.

BLACK BOX (EEPROM) FAULT RECORDING

The ADM1266 has a configurable black box feature. Using this feature, the device is capable of recording to nonvolatile flash memory the vital data about the system status that caused the system to perform a black box write.

BLACK BOX WRITES WHEN EXTERNAL SUPPLY IS POWERING DOWN

When all the input supplies fail, the state machine can be programmed to trigger a write into the black box flash. Provided that the AVDD_CAP voltage remains above 3.0 V during the memory write, the entire fault record is written to the EEPROM. To ensure a complete black box write, it is recommended to place a capacitor of at least 68 µF on the AVDD_CAP pin.

TRIGGERING A BLACK BOX WRITE

Black box information can be captured in the loop action or enter action of a state, when the black box action is triggered.

If the black box action is triggered in the loop action, the device takes a snapshot immediately and writes it to the flash memory at the end of the enter actions of the next state.

If the black box action is triggered in the enter actions, the device takes a snapshot immediately and writes it at the end of the enter actions of the same state.

When multiple ADM1266 devices are connected though the IDB, a black box write trigger in each device initiates a black box write to ensure that the status of the entire system is captured. Each black box record has a unique ID that is the same across all the devices, which enables combining information together from multiple devices.

BLACK BOX RECORD MODE

There are two types of black box record mode: single mode and cyclic mode. Four pages of flash memory are reserved for a single mode black box record, and five pages of flash memory for cyclic mode. Each black box record has 64 bytes. The black box mode can be changed without power cycling the device.

Single Mode

In single mode, the black box can write up to 32 fault records. When the 32 records are filled, the ADM1266 black box does not write anymore until the records are erased. Single mode is useful for keeping the initial fault records and preventing them from being overwritten.

Cyclic Mode

In cyclic mode, the black box operates in a circular recording mode, and after writing the eighth record of any page, the next page is automatically erased to allow continuous black box recording. In cyclic mode, there can be up to 32 records at a time. Cyclic mode is useful to keep the most recent black box information.

POWER-UP COUNTER

The power-up counter in the ADM1266 keeps a record of the number of times the ADM1266 has been powered up. It is stored in nonvolatile memory. The power-up counter is 2 bytes and has a maximum count up to 65,535 power cycles. The counter increments automatically at every power cycle of ADM1266 and cannot be reset by the user.

BLACK BOX WRITE TIME

Writing 4 bytes of data to the flash memory takes 46 µs, and each fault record has 64 bytes of data. The total time taken to write one fault record is approximately 736 µs.

BLACK BOX CONTENTS

The total number of black box records in the device can be read from the record count byte of the BLACKBOX_INFORMATION register. The index of the last record that was written to the black box is pointed by the logic index byte of the BLACKBOX_INFORMATION register. The value is only valid when the record count is greater than zero.

The last black box record number can be read back by the READ_BLACKBOX register.

The black box record data can be read by READ_BLACKBOX. The record number and the last record index can be read back by BLACKBOX_INFORMATION.

TIME STAMPING

In the black box records, there is an option to save the time of the black box write, which is beneficial in tracking the time of the failures. ADM1266 has a real-time counter (RTC) that keeps track of time. The RTC is reset to zero when ADM1266 is powered down.

The RTC can be used in two ways. The RTC can be used to measure the time elapsed since the last time the ADM1266 was powered up, which is helpful in determining how much time passed when the system failed after powering up.

In a system, the host controller can send the UNIX® time to the ADM1266. If a UNIX time is received, the RTC can be used as a reference to start counting from the UNIX time. When the UNIX time is set, the device increments from this time and uses it in black box records to convert to real time. The UNIX time must be set every time the ADM1266 powers up, because the RTC resets at power-down.

SETTING UNIX TIME USING SET_RTC

The SET_RTC register consist of six bytes that can be used to set the time elapsed since January 1, 1970, according to the UNIX time system. Each LSB represents $1/(2^{16})$ s. In a system where multiple ADM1266 devices are connected together, use the SYNC pin to synchronize the time counter between all ADM1266 devices.

INTERNAL OSCILLATOR

The internal oscillator in ADM1266 can be used for RTC where the accuracy of time stamping is not critical. If the RTC is used for the UNIX time with the internal oscillator, it is recommended that the system host frequently send the time stamp to the ADM1266 to synchronize the UNIX time and reduce the time from drifting.

EXTERNAL OSCILLATOR

In an application where accurate time stamping is required, it is recommended to use an external 32,768 Hz crystal to generate a time base for the RTC. An external crystal is connected to the ADM1266 using the XTAL1 and XTAL2 pins. In a system with multiple ADM1266 devices, only one crystal is required.

MULTIPLE DEVICE TIME STAMPING

In a system where multiple ADM1266 devices are connected and an external crystal is used, connect the SYNC pins so that the RTC is using the same oscillator across all devices. This configuration minimizes drift in time between devices caused by variation in oscillating frequency, and requires one external crystal. Configure the SYNC pin of the device with the external crystal connected as an output, and the SYNC pin of the other devices as an input. The SYNC pin configuration can be set using the GPIO_SYNC_CONFIGURATION (Register 0xE1).

In a multiple device system, the real time can be set by sending the UNIX time to the SET_RTC register of only one device. The time is broadcast to the other devices in the system using the IDB to ensure that all the devices have the same real time.

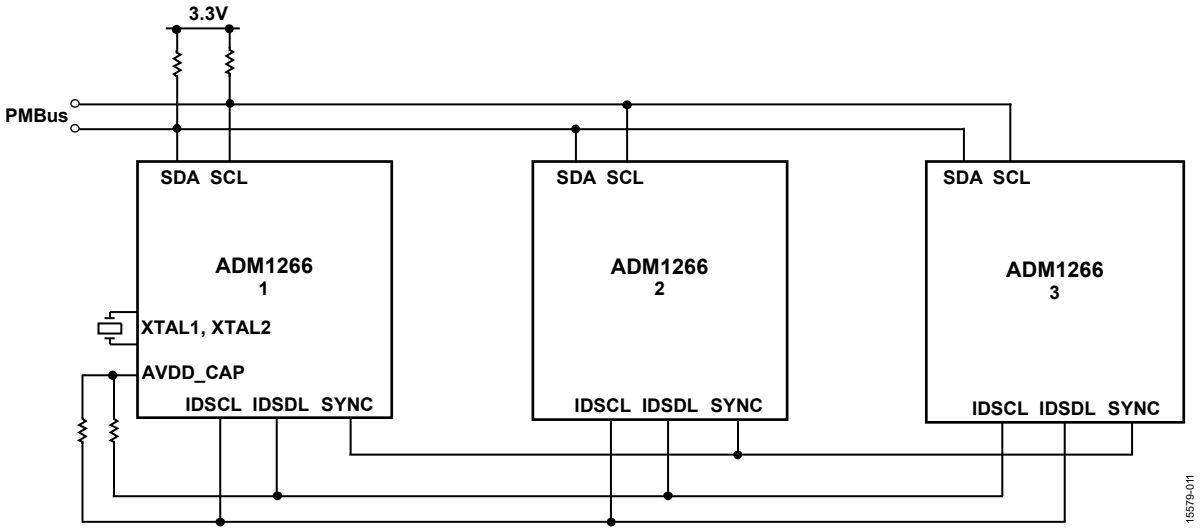


Figure 15. Setting Up Time Stamping in a Multidevice Setup

SYSTEM LOGIC BLOCK

The ADM1266 features a user configurable combinational logic block. The input to the logic block can be the status of VHx, VPx, a GPIO, or PDIO, and the output sets the PDIOs or GPIOs.

The logic block operates independent of the SE and margining block. The logic block has lower priority than the SE. As a result, if the sequencing engine is busy running a sequence, the output of the logic blocks is delayed until completion of the present task.

The logic function consists of five core logic elements: AND, OR, NAND, NOR, and NOT. Multiple logic elements can be cascaded to achieve any user defined logic combination.

The inputs to the logic gates can be a combination of PDIOs, GPIOs, VHx/VPx warnings, VHx/VPx faults, and the output of other logic gates.

The output of the logic gates can be used to drive the GPIOs, PDIOs, or the input of other logic gates.

In a multiple device system, the inputs and outputs from multiple devices cannot be used in the same logic block because the logic function block does not use the IDB. The output of logic function from one ADM1266 device can be propagated to another by using a GPIO or PDIO.

The maximum number of inputs to a logic element or a cascaded logic element is 256.

The logic is configured using the manufacture specific command, LOGIC_CONFIGURATION (Register 0xE0). It is recommended to use the Power Studio software for programming the logic blocks to the specifications of the user.

For example, in a system with three voltage rails, UV or OV warning status of each rail can be logically OR'ed together to set a single status signal.

PASSWORD PROTECTION

Specific commands are password protected to avoid unintended modification of the firmware, sequence, and project configuration data in the ADM1266. These commands must be unlocked only when updating the firmware, sequence, and project configuration data. Table 11 shows the list of password protected commands. It is not required to unlock the device for regular operation. The password is 16 bytes, and for the default password, those 16 bytes are all 0xFF.

Table 11. Password Protected Commands

Command	Address
UPDATE_FW	0xFC
SEQUENCE_CONFIGURATION	0xD6
SYSTEM_CONFIGURATION	0xD7
LOGIC_CONFIGURATION	0xE0
USER_DATA	0xE3
STORE_USER_ALL	0x15
REFRESH_FLASH	0xF5
ERASE_MEMORY	0xFB
MEMORY_CONFIGURATION	0xF8

UNLOCKING THE DEVICE

The ADM1266 can be unlocked by performing two consecutives writes of the correct password to the FW_PASSWORD command (Register 0xFD). The block write command for unlocking the device is shown in Figure 16.

The unlock status can be confirmed by the PART_LOCKED bit of the STATUS_MFR_SPECFIC command (Register 0x80) which is set to 0 when the device is successfully unlocked.

LOCKING THE DEVICE

Upon a power cycle, the device is automatically locked. The device can also be locked by writing any 17 bytes of data once to the FW_PASSWORD command (Register 0xFD). The block write command for locking the device is shown in Figure 17.

The lock status can be confirmed by the PART_LOCKED bit of the STATUS_MFR_SPECFIC (Register 0x80) command, which is set to 1 when the device is successfully locked.

CHANGING THE PASSWORD

The password can be changed by the user to any 16-byte value. For password less than 16 bytes, set the remaining bytes as 0x00. To update the password, the device must be unlocked by following the procedure described in the Unlocking the Device section. After the device is unlocked, the new password must be written two consecutive times to the FW_PASSWORD command (Register 0xFD). The block write command for changing the password is shown in Figure 18.

After the password is updated, the new password is immediately committed to the memory. The device is automatically locked and must be unlocked using the new password.

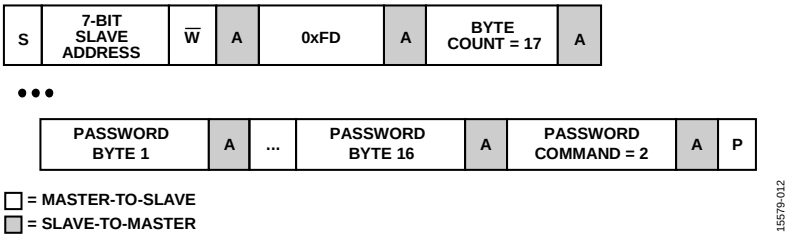


Figure 16. Block Write Command for Unlocking the Device

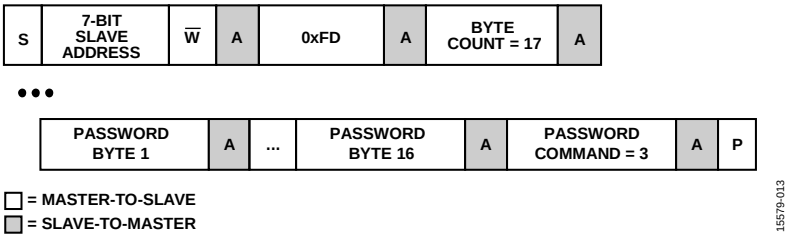


Figure 17. Block Write Command for Locking the Device

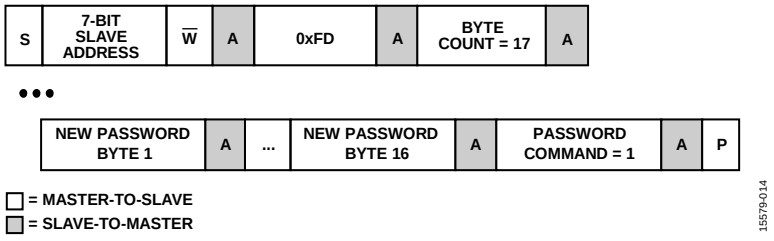


Figure 18. Block Write Command for Changing the Password

MEMORY

OVERVIEW

The ADM1266 contains internal EEPROM (nonvolatile memory) to store the mini boot loader, the boot loader, firmware, configuration settings, and fault log information. The mini boot loader, boot loader, firmware, and configuration settings each have a main copy and a backup copy in the memory. Each section has its own unique cyclic redundancy check (CRC), and each black box record has its own unique CRC.

POWER-UP

At power-up, the main mini boot loader checks the data of the main boot loader and compares it to the CRC. If the data is corrupted, the main mini boot loader checks the data of the backup bootloader and compares it with its CRC. If the backup bootloader data matches the CRC, this data is copied to the main boot loader and is fixed (the ADM1266 copies the data from backup memory to main memory and corrects the corrupted data). Then, the main boot loader starts to execute the boot loader. The main boot loader checks the data of the main firmware and compares it to the CRC. If the data is corrupted, the main boot loader checks the data of the backup firmware and compares this data to the CRC. If the backup firmware data matches the CRC, the ADM1266 copies it over to the main firmware and fixes the data. Then, the ADM1266 starts to execute the firmware. The firmware then checks the data of the main and backup configuration and compares it with the respective CRCs. If both sections match the calculated CRC value of memory with the saved CRC value, then the ADM1266 runs the main configuration. If one of the sections matches the CRC, the ADM1266 runs the correct configuration. In a multi-device system, all the devices share the information about their main and backup configuration with the master device. Then, the master device makes a decision and communicates to all the devices which section of the configuration memory to run.

If at any given point for any of the sections both the main and backup sections are corrupted, the device does not proceed.

MANUAL CRC CALCULATIONS

The ADM1266 has several commands to validate the condition of the memory. Use MEMORY_RECALCULATE_CRC (Register 0xF9) to trigger the device to recalculate the CRC of all the sections and to report the status in STATUS_MFR_SPECIFIC_2 (Register 0xED). The time required to recalculate the CRC of all the sections is approximately 500 ms.

REFRESH

The ADM1266 allows data to be copied from the main sections to the backup sections, and vice versa. REFRESH_FLASH (Register 0xF5) can be used to trigger this function. When this function is triggered, the ADM1266 checks the CRC of both the main and backup sections and copies data from the expected data (not corrupted) section over to the corrupted section, and vice versa. Based on the data written to REFRESH_FLASH, the user can choose to refresh certain sections of the memory.

To increase the reliability of the memory, it is recommended to run this refresh feature once every 30 days. For operating temperatures above 85°C, it is mandatory to refresh once every 30 days. Every time the refresh is run, the data retention timer resets.

When the refresh feature is running, it takes 32 ms to refresh each page. During this time, all faults are latched but not processed. After refreshing each page, if there is any sequence event, the refreshing is temporarily aborted and the sequencing and fault handling functions are executed. At the end of this process, the ADM1266 resumes the refreshing function. It takes approximately 9 sec to finish refreshing all the sections of the ADM1266.

PMBus write operations are not allowed when the refresh feature is running. PMBus read operations are clock stretched and processed at the end of refreshing each page.

AUTO REFRESH

The ADM1266 can be configured to run the refresh feature automatically after this feature is enabled and saved to the memory. After one day, each time the device is powered up the device automatically starts the refresh of the boot loader, firmware, and configuration sections. Once a day, the device runs the CRC check. If any of the mini boot loader, boot loader, firmware, or configuration sections are corrupted, the device automatically starts the refresh of the mini boot loader, the boot loader, firmware, and the configuration sections. After the initial refresh that occurs after one day, the ADM1266 can be pre-programmed to automatically start refresh every N days, where N varies from 1 day to 255 days. The default setting is 30 days.

ACCELERATION FACTOR

The ADM1266 contains internal EEPROM (nonvolatile memory) to store the mini boot loader, the boot loader, the firmware, configuration settings, and fault log information. EEPROM endurance and retention are specified over the operating junction temperature range (see the Absolute Maximum Ratings section and the Electrical Specifications section).

Nondestructive operation above T_J = 85°C is possible. However, the electrical specifications are not guaranteed and, in this case, the EEPROM degrades. Operating the EEPROM above T_J = 85°C may result in a degradation of retention characteristics. The fault logging function, which is useful in debugging system problems that may occur at high temperatures, only writes to fault log EEPROM locations. If occasional writes to these registers occur above T_J = 85°C, a slight degradation in the data retention characteristics of the fault log may occur. It is recommended that the EEPROM not be written using STORE_USER_ALL or bulk programming when T_J > 85°C. The degradation in EEPROM retention for temperatures T_J > 85°C can be approximated by

calculating the dimensionless acceleration factor using the following equation:

$$AF = e^{\left(\left(\frac{Ea}{k}\right)\times\left(\frac{1}{T_{USE}+273}-\frac{1}{T_{STRESS}+273}\right)\right)}$$

where:
AF is the acceleration factor.
Ea, the activation energy, = 0.6 eV.
$k = 8.617 \times 10^{-5}$ eV/°K.
T_{USE} = 85°C, the specified junction temperature.
T_{STRESS} is the actual junction temperature.

For example, calculate the effect on retention when operating at a junction temperature of 125°C for 10 hours.

T_{STRESS} = 125°C

AF = 7.062

The equivalent operating time at 85°C = 70.62 hours.

Therefore, the overall retention of the EEPROM degrades by 60.62 hours as a result of operation at a junction temperature of 125°C for 10 hours. The effect of this overstress is negligible when compared to the overall EEPROM retention rating of 87,600 hours at a maximum junction temperature of 85°C.

Using the previously mentioned equation, data retention can be calculated at different temperatures, as shown in Table 12.

Table 12. Data Retention vs. Temperature

Junction Temperature	Data Retention
85°C	10 years
95°C	5 years and 10 months
105°C	3 years and 7 months

APPLICATIONS INFORMATION

OVERVIEW

The ADM1266 Super Sequencer is capable of sequencing, margining, trimming, supervising output voltage for OV and UV conditions, providing fault management, and voltage read-back for 16 dc-to-dc converters. Multiple ADM1266 devices can be synchronized to operate in unison using the ID_SCL and ID_SDA pins. The ADM1266 uses a PMBus-compliant interface and command set.

POWERING THE ADM1266

The ADM1266 can be powered by applying a voltage from 3 V to 15 V on the VH1 or VH2 pin. Internal linear regulators convert this voltage down to 3.3 V, which drives all of the internal circuitry in each device. It is not recommended to connect both VH1 and VH2 to the same voltage levels because the ripple on the two voltages may cause the arbitrator circuit to constantly toggle. In a system with multiple ADM1266 devices, it is important that all the devices are powered from the same voltage rail.

PCB ASSEMBLY AND LAYOUT SUGGESTIONS

The ADM1266 requires capacitors (see the Capacitors section). To be effective, these capacitors must be high quality, ceramic dielectric capacitors, such as X5R or X7R, and must be placed as close to the chip as possible. The PCB layout must adhere to layout guidelines. A multilayer PCB that dedicates a layer to power and ground is recommended. Low resistance and low inductance power and ground connections are important to minimize power supply noise and ensure proper device operation.

CAPACITORS

Place a 10 µF bypass capacitor and a 0.1 µF decoupling capacitor on both the VH1 and VH2 pins.

Place a 68 µF capacitor and a 0.1 µF capacitor on the AVDD_CAP pin.

Place a 10 µF capacitor and a 0.1 µF capacitor on the DVDD_CAP pin

Place a 2.2 µF capacitor and a 0.1 µF capacitor between the REFOUT and REFGND pins.

GROUND CONNECTIONS

Connect the exposed pad to the GND pin. Star connect the GND pin to the REFGND pin.

PMBUS/I²C

Each ADM1266 must be configured for a unique address. The address can be set by connecting a resistor between the ADDR pin and the GND pin. See Table 13 for the corresponding address values. Check addresses for collision with other devices on the bus and any global addresses.

The pull-up resistors on the PMBus pins must not be connected to AVDD_CAP. If another device on the PMBus line provides a strong pull-down on AVDD_CAP, the ADM1266 shuts down or enters UVLO.

IDB

For a board with multiple ADM1266 devices that are part of the same system, connect the ID_SCL and ID_SDA pin, using an external pull-up resistor of 2.2 kΩ that is connected to the AVDD_CAP pin of any ADM1266.

VOLTAGE SENSING

If an external resistor divider is used, calculate the size of the resistors so that 0.7 V shows up on the VPx pins of the ADM1266.

When sensing directly, use a 100 Ω resistor in series to avoid any latch-ups on the pins. The VH1 and VH2 pins do not require this series resistance.

PDIOs AND GPIOs

The PDIOs have a weak, 20 kΩ, internal pull-down resistor. Therefore, the PDIOs do not require the external pull-down resistors during power-up.

Verify that the voltage and current ratings are not exceeded.

DAC OUTPUTS

Select an appropriate resistor for the desired margin range on a DAC output. Refer to the Power Studio GUI for assistance.

CLOCK

To use the accurate time stamping and clocking function, use an external oscillator and capacitors between the XTAL1 and XTAL2 pins. If this function is not used, an external clock source is not required.

On a board with multiple ADM1266 devices, only one external oscillator is required. Connect the SYNC pins of all the ADM1266 devices.

UNUSED PINS

Connect all unused pins to GND.

PMBus DIGITAL COMMUNICATION

The PMBus slave with packet error checking (PEC) allows a device to interface to a PMBus compliant master device, as specified by the *PMBus Power System Management Protocol Specification* (Revision 1.2, September 6, 2010). The PMBus slave is a 2-wire interface that can be used to communicate with other PMBus compliant devices and is compatible in a multimaster, multislave bus configuration. The PMBus slave can communicate with master PMBus devices that support packet error checking (PEC), as well as with master devices that do not support PEC.

The pull-up resistors on the PMBus pins must not be connected to AVDD_CAP. If another device on the PMBus line provides a strong pull-down AVDD_CAP, the ADM1266 shuts down or enters UVLO.

PMBus FEATURES

The function of the PMBus slave is to decode the command sent from the master device and to respond as requested. Communication is established using an I²C like 2-wire interface with a clock line (SCL) and data line (SDA). The PMBus slave is designed to externally move blocks of 8-bit data (bytes) while maintaining compliance with the PMBus protocol. The PMBus protocol is based on the *SMBus Specification* (Version 2.0, August 2000). The SMBus specification is, in turn, based on the Philips *I²C Bus Specification* (Version 2.1, January 2000). The PMBus incorporates the following features:

- Slave operation on multiple device systems
- 7-bit addressing
- 100 kbps and 400 kbps data rates
- PEC
- Support for the group command protocol
- Support for the alert response address protocol with arbitration
- General call address support
- Support for clock low extension (clock stretching)
- Separate multiple byte receive and transmit first in, first out (FIFO)
- Extensive fault monitoring

OVERVIEW

The PMBus slave module is a 2-wire interface that can be used to communicate with other PMBus compliant devices. Its transfer protocol is based on the Philips I²C transfer mechanism. The ADM1266 is always configured as a slave device in the overall system. The ADM1266 communicates with the master device using one data pin (SDA) and one clock pin (SCL). Because the ADM1266 is a slave device, it cannot generate the clock signal. However, the ADM1266 is capable of clock stretching the SCL line to put the master device in a wait state when the ADM1266 is not ready to respond to the request of the master.

Communication is initiated when the master device sends a command to the PMBus slave device. Commands can be read or write commands. Data is transferred between the devices in a byte wide format. Commands can also be send commands. The command is executed by the slave device upon receiving the stop bit. The stop bit is the last bit in a complete data transfer, as defined in the PMBus/SMBus/I²C communication protocol. During communication, the master and slave devices send acknowledge or no acknowledge bits as a method of handshaking between devices.

In addition, the PMBus slave on the ADM1266 supports PEC to improve reliability and communication robustness. The ADM1266 can communicate with master PMBus devices that support PEC, as well as with master devices that do not support PEC. See the *SMBus Specification* (Version 2.0) for a more detailed description of the communication protocol.

When communicating with the master device, it is possible for illegal or corrupted data to be received by the PMBus slave device. In this case, the PMBus slave device responds to the invalid command or data, as defined by the PMBus specification, and indicates to the master device that an error or fault condition has occurred. This method of handshaking can be used as a first level of defense against inadvertent programming of the slave device that can potentially damage the chip or system.

The PMBus specification defines a set of generic PMBus commands that are recommended for a power management system. However, each PMBus device manufacturer can choose to implement and support certain commands as the manufacturer deems fit for a specific system. In addition, the PMBus device manufacturer can choose to implement manufacturer specific commands with functions not included in the generic PMBus command set.

TRANSFER PROTOCOL

The PMBus slave follows the transfer protocol of the *SMBus Specification* (Version 2.0), which is based on the fundamental transfer protocol format of the Philips *I²C Bus Specification* (Version 2.1). Data transfers are byte wide, lower byte first. Each byte is transmitted serially, most significant bit (MSB) first. Figure 19 shows a basic transfer.

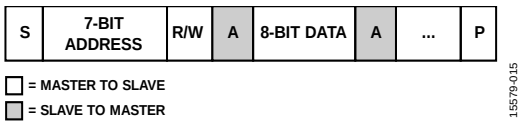


Figure 19. Basic Data Transfer

For an in depth description of the transfer protocols, see the SMBus and I²C specifications.

DATA TRANSFER COMMANDS

Data transfer using the PMBus slave is established using PMBus commands. The PMBus specification requires that all PMBus commands start with a slave address with the R/$\overline{W}$ bit cleared (set to 0), followed by the command code. (The only exception is the alert response address protocol.)

All PMBus commands supported by the ADM1266 device follow one of the protocol types shown in Figure 20 to Figure 27. (For PMBus master devices that do not support PEC, the PEC byte is removed.) Figure 20 to Figure 27 use the following abbreviations:

- S is the start condition
- P is the stop condition
- Sr is the repeated start condition
- W is the write bit (0)
- R is the read bit (1)
- A is the acknowledge bit (0)
- NA is the no acknowledge bit (1)

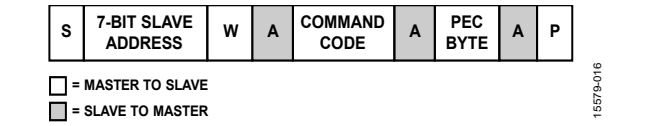


Figure 20. Send Protocol with PEC

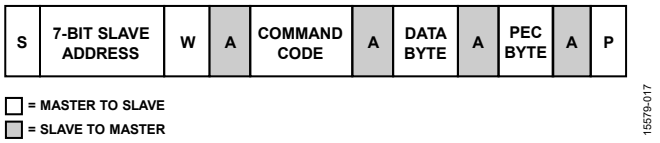


Figure 21. Write Byte Protocol with PEC

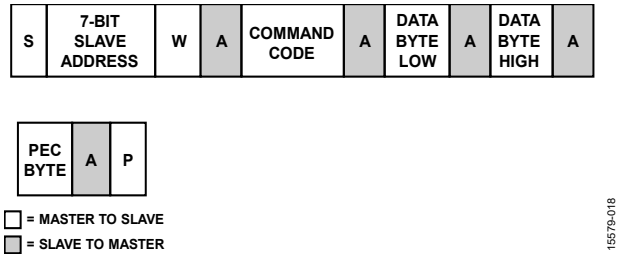


Figure 22. Write Word Protocol with PEC

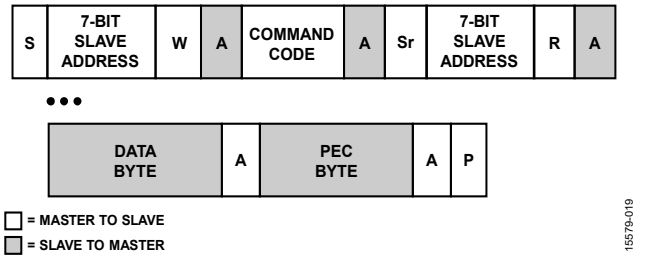


Figure 23. Read Byte Protocol with PEC

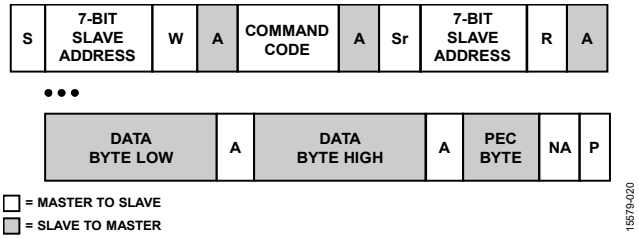


Figure 24. Read Word Protocol with PEC

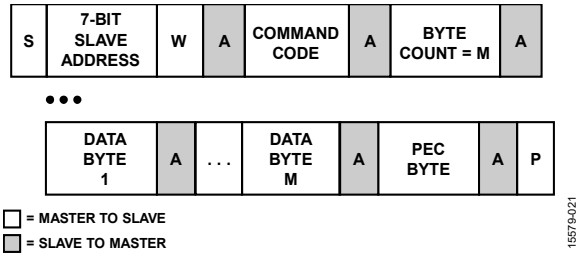


Figure 25. Block Write Protocol with PEC

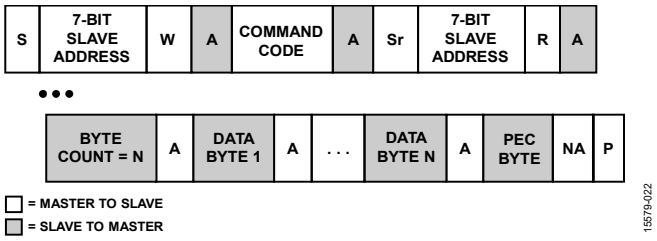


Figure 26. Block Read Protocol with PEC

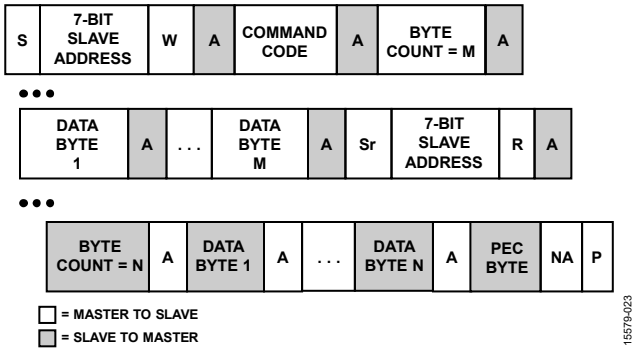


Figure 27. Block Write and Block Read Protocol with PEC

The PMBus slave module of the ADM1266 also supports manufacturer specific extended commands. These commands follow the same protocol as the standard PMBus commands. However, the command code consists of two bytes:

- Command code extension: 0xFE
- Extended command code: 0x00 to 0xFF

Using the manufacturer specific extended commands, the PMBus device manufacturer can add an additional 256 manufacturer specific commands to its PMBus command set.

GROUP COMMAND PROTOCOL

In addition to the communication protocols described in the Data Transfer Commands section, the PMBus slave supports a special group command in which commands are sent to multiple slaves in a single serial transmission. The commands to each slave can be different from one another, with each set of slave address and command separated by a repeated start (Sr) bit (see Figure 28). At the end of a transmission to all slaves, a single stop (P) bit is sent to initiate concurrent execution of the received commands by all slaves.

The PEC byte transmitted to each slave is calculated using only its slave address, command code, and data bytes.

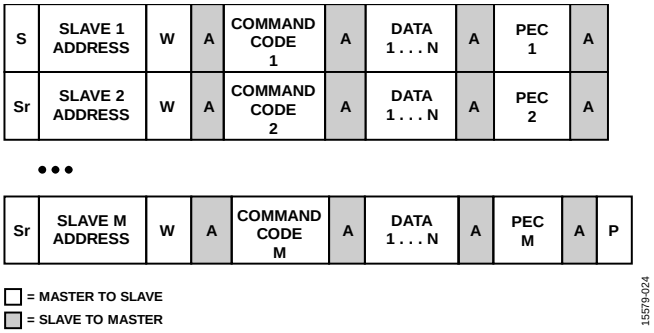


Figure 28. Group Command Protocol with PEC

CLOCK GENERATION AND STRETCHING

The ADM1266 is always a PMBus slave device in the overall system; therefore, the device never needs to generate the clock, which is performed by the master device in the system. However, the PMBus slave device is capable of clock stretching to place the master in a wait state. By stretching the SCL signal during the low period, the slave device communicates to the master device that it is not ready and that the master device must wait.

Conditions where the PMBus slave device stretches the SCL line low include the following:

- The master device is transmitting at a higher baud rate than the slave device.
- The receive FIFO buffer of the slave device is full and must be read before continuing to prevent a data overflow condition.
- The slave device is not ready to send data that the master has requested.

The slave device can stretch the SCL line only during the low period. Whereas the I²C specification allows indefinite stretching of the SCL line, the PMBus specification limits the maximum time that the SCL line can be held low to 25 ms, after which the ADM1266 must release the communication lines and reset its state machine.

START AND STOP CONDITIONS

Start and stop conditions involve serial data transitions while the serial clock is at a logic high level. The PMBus slave device monitors the SDA and SCL lines to detect the start and stop conditions and transition to its internal state machine accordingly. Figure 29 shows typical start and stop conditions.

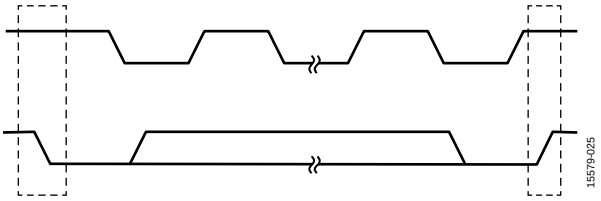


Figure 29. Start and Stop Transitions

REPEATED START CONDITION

In general, a repeated start (Sr) condition is the absence of a stop condition between two transfers. The PMBus communication protocol makes use of the repeated start condition only when performing a read access (read byte, read word, and block read). Other uses of the repeated start condition are not allowed.

GENERAL CALL SUPPORT

The PMBus slave is capable of decoding and acknowledging a general call address. The PMBus device responds to both its own address and the general call address (0x00).

All PMBus commands must start with the slave address with the R/W bit cleared (set to 0), followed by the command code, when using the general call address to communicate with the PMBus slave device.

PMBus ADDRESS SELECTION

Control of the ADM1266 is implemented via the I²C interface. The ADM1266 device is connected to the I²C bus as a slave device under the control of a master device. The PMBus address of the ADM1266 is set by connecting an external resistor from the ADDR pin to GND. Table 13 lists the recommended resistor values and associated PMBus addresses.

Table 13. PMBus Address Settings

PMBus Address	1% Resistor (kΩ) (E96 Series)
0x40	0.422
0x41	1.5
0x42	2.67
0x43	4.12
0x44	5.36
0x45	7.15
0x46	8.87
0x47	10.7
0x48	12.7
0x49	14.7
0x4A	16.9
0x4B	19.1
0x4C	21.5
0x4D	24.3
0x4E	27.4
0x4F	31.6

FAST MODE

Fast mode (400 kHz) uses essentially the same mechanics as the standard mode of operation. The PMBus slave is capable of communicating with a master device operating in standard mode (100 kHz) or fast mode.

10-BIT ADDRESSING

The PMBus slave device does not support 10-bit addressing as defined in the I²C specification.

PACKET ERROR CHECKING

The PMBus controller implements PEC to improve reliability and communication robustness. Packet error checking is implemented by appending a PEC byte at the end of the message transfer. The PEC byte is calculated using a CRC-8 algorithm on all address, command, and data bytes from the start to stop bits (excluding the acknowledge, no acknowledge, start, restart, and stop bits). The PEC byte is appended to the end of the message by the device that supplied the last data byte. The receiver of the PEC byte is responsible for calculating its internal PEC code and comparing it to the received PEC byte.

The ADM1266 can communicate with master PMBus devices that support PEC, as well as with master devices that do not support PEC. If a PEC byte is available, the PMBus device checks the PEC byte and issues an acknowledge if the PEC byte is correct. If the PEC byte comparison fails, the PMBus device issues a no acknowledge in response to the PEC byte and does not process the command sent from the master.

The PMBus device uses built in hardware to calculate the PEC code using the CRC-8 polynomial, $C(x) = x^8 + x^2 + x^1 + 1$. The PEC code is calculated one byte at a time, in the order that the bytes are received. In a read transaction, the PMBus device appends the PEC byte following the last data byte. In a write transaction, the PMBus device compares the received PEC byte to the internally calculated PEC code.

ELECTRICAL SPECIFICATIONS

All logic complies with the electrical specification outlined in the *PMBus Power System Management Protocol Specification Part 1* (Revision 1.2, September 6, 2010).

PMBus COMMANDS

Table 14 lists the standard PMBus commands that are implemented on the ADM1266. Many of these commands are implemented in registers that share the same hexadecimal value as the PMBus command code.

Table 14. PMBus Command List

Code	Name	Type[1]	Bytes
0x00	PAGE	R/W	1
0x01	OPERATION	R/W	1
0x03	CLEAR_FAULTS	S	0
0x15	STORE_USER_ALL	S	0
0x16	RESTORE_USER_ALL	S	0
0x19	CAPABILITY	R	1
0x20	VOUT_MODE	R/W	1
0x21	VOUT_COMMAND	R/W	2
0x22	VOUT_TRIM	R/W	2
0x25	VOUT_MARGIN_HIGH	R/W	2
0x26	VOUT_MARGIN_LOW	R/W	2
0x29	VOUT_SCALE_LOOP	R/W	2
0x2A	VOUT_SCALE_MONITOR	R/W	2
0x40	VOUT_OV_FAULT_LIMIT	R/W	2
0x42	VOUT_OV_WARN_LIMIT	R/W	2
0x43	VOUT_UV_WARN_LIMIT	R/W	2
0x44	VOUT_UV_FAULT_LIMIT	R/W	2
0x78	STATUS_BYTE	R/W	1
0x79	STATUS_WORD	R/W	2
0x7A	STATUS_VOUT	R	1
0x7E	STATUS_CML	R	1
0x80	STATUS_MFR_SPECIFIC	R	1
0x8B	READ_VOUT	R	2
0x98	PMBUS_REVISION	R	1
0x99	MFR_ID	Block WR/W	1 to 32
0x9A	MFR_MODEL	Block WR/W	1 to 32
0x9B	MFR_REVISION	Block WR/W	1 to 8
0x9C	MFR_LOCATION	Block WR/W	1 to 48
0x9D	MFR_DATE	Block WR/W	1 to 16
0x9E	MFR_SERIAL	Block WR/W	1 to 32
0xAD	IC_DEVICE_ID	Block R	3
0xAE	IC_DEVICE_REV	Block R	8
0xD0	VOUT_OV_HYST_LIMIT	R/W	2
0xD1	VOUT_UV_HYST_LIMIT	R/W	2
0xD2	Vx_CONFIGURATION	R/W	2
0xD3	BLACKBOX_CONFIGURATION	R/W	2
0xD4	PDIO_CONFIGURATION	Block WR, Block W	2 to 32, 3 to 33
0xD5	DAC_CONFIGURATION	Block WR, Block W	2 to 18, 3 to 19
0xD6	SEQUENCE_CONFIGURATION	Block WR/W	3 to 250
0xD7	SYSTEM_CONFIGURATION	Block WR/W	3 to 250
0xD8	GO_COMMAND	R/W	2
0xD9	READ_STATE	R	2
0xDA	VOUT_MARGIN_LOOP	R/W	2
0xDB	MARGIN_CONFIGURATION	R/W	2
0xDC	BREAKPOINTS	Block WR/W	1 to 128
0xDD	ICB_CONFIGURATION	Block R/W	8
0xDE	READ_BLACKBOX	Block WR, Block W	65, 2
0xDF	SET_RTC	Block R/W	6
0xE0	LOGIC_CONFIGURATION	Block WR/W	3 to 250
0xE1	GPIO_SYNC_CONFIGURATION	Block WR	1
		Block W	2
0xE3	USER_DATA	Block WR/W	3 to 250
0xE4	POWERUP_COUNTER	Block R	2
0xE5	VOUT_RESISTOR	Block WR/W	5 to 16
0xE6	BLACKBOX_INFORMATION	Block R	4
0xE7	ALL_STATUS_VOUT	Block R	17
0xE8	ALL_READ_VOUT_MODE	Block R	51
0xE9	PDIO_STATUS	Block R	2
0xEA	GPIO_STATUS	Block R	2
0xEB	DAC_CODE_CONFIGURATION	Block WR, Block W	3 to 19, 4 to 20
0xEC	RTS_CONFIGURATION	R/W	2
0xED	STATUS_MFR_SPECIFIC_2	R	2
0xF4	REFRESH_CONFIGURATION	Block W	3 to 4
		Block WR	2 to 9
0xF5	REFRESH_FLASH	R/W	2
0xF6	HITLESS_TIMEOUT	R/W	2
0xF7	VAR_VALUE	Block WR	2 to 5
0xF8	MEMORY_CONFIGURATION	Block R/W	3
0xF9	MEMORY_RECALCULATE_CRC	W	2
0xFA	SWITCH_MEMORY	Block W	1
0xFB	ERASE_MEMORY	Block W	1
0xFC	UPDATE_FW	W	2
0xFD	FW_PASSWORD	Block W	17

[1] S is the send byte command, no data. Block WR is the block write parameter and block read data. Block W is the block write command. Block R is the block read command. Block WR/W means a standard block write is performed to write to the command, but the command must be written first before it can be read back.

STANDARD PMBus COMMAND DESCRIPTIONS

All commands designated as Block WR/W consist of two writes and a read. A standard block write is performed to write to the command, but the command must be written first before it can be read back.

STANDARD PMBus COMMANDS

Page

The page command provides the ability to configure, control, and monitor using only one physical address.

Table 15. Register 0x00—Page

Bits	Bit Name	Type	Description
[7:0]	PAGE	R/W	00000 = VH1. 00001 = VH2. 00010 = VH3. 00011 = VH4. 00100 = VP1. 00101 = VP2. 00110 = VP3. 00111 = VP4. 01000 = VP5. 01001 = VP6. 01010 = VP7. 01011 = VP8. 01100 = VP9. 01101 = VP10. 01110 = VP11. 01111 = VP12. 10000 = VP13. Setting the page to 0xFF means that all following commands are to be applied to all inputs.

Operation

The operation command turns the closed-loop margining on and off, and determines which voltage to margin to.

Table 16. Register 0x01—Operation

Bits	Bit Name	R/W	Description
[7:6]	MARGIN_EN	R/W	01 is soft off, 10 is margin on, and others are reserved.
[5:4]	MARGIN_VOLTAGE	R/W	00 is VOUT_COMMAND (closed-loop servo to the voltage set in VOUT_COMMAND), 01 is margin low, 10 is margin high, and others are reserved.
[3:2]	Fault	R/W	01 is ignore fault and others are reserved.
[1:0]	Reserved	R	Reserved.

CLEAR_FAULTS

The CLEAR_FAULTS command is a send byte, with no data. This command clears all fault bits in all PMBus status registers simultaneously.

Table 17. Register 0x03—CLEAR_FAULTS

Bits	Bit Name	Type	Description
Not Applicable	CLEAR_FAULTS	Send	Clears all bits in the PMBus status registers (Register 0x78 to Register 0x7A) simultaneously.

STORE_USER_ALL

Table 18. Register 0x15—STORE_USER_ALL

Bits	Bit Name	Type	Description
Not Applicable	STORE_USER_ALL	Send	This command copies the entire contents of the operating memory into the device memory.

RESTORE_USER_ALL

Table 19. Register 0x16—RESTORE_USER_ALL

Bits	Bit Name	Type	Description
Not Applicable	RESTORE_USER_ALL	Send	This command downloads the stored user settings from device memory into operating memory.

Capability

This command allows host systems to determine the capabilities of the PMBus device.

Table 20. Register 0x19—Capability

Bits	Bit Name	R/W	Description
7	Packet error checking	R	Checks packet error capability of the device. 1 is supported.
[6:5]	Maximum bus speed	R	Checks the PMBus speed capability of the device. 01 is the maximum supported bus speed, 400 kHz.
4	$\overline{\text{SMBALRT}}$	R	Checks support for the SMBus alert pin and the SMBus alert response address protocol. 0 = not supported.
[3:0]	Reserved	R	Reserved.

VOUT_MODE

The VOUT_MODE command sets the data format for output voltage related data. The data byte for the VOUT_MODE command consists of a 3-bit mode and 5-bit exponent parameter. The 3-bit mode determines whether the device uses linear format or direct format for the output voltage related commands. The 5-bit parameter sets the exponent value for linear format.

Table 21. Register 0x20—VOUT_MODE

Bits	Bit Name	R/W	Description
[7:5]	Mode	R	Returns the output voltage data format. The value is fixed at 000, which means that only linear data format is supported.
[4:0]	Exponent N	R/W	Twos complement of Exponent N used in the output voltage related commands in linear data format ($V = Y \times 2^N$), where Y is mantissa.

VOUT_COMMAND

The VOUT_COMMAND command sets the output voltage. Exponent N is set using VOUT_MODE[4:0].

Table 22. Register 0x21—VOUT_COMMAND

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R/W	16-bit unsigned integer Y value for linear data format ($V = Y \times 2^N$). N is defined using VOUT_MODE[4:0].

VOUT_TRIM

The VOUT_TRIM command applies a fixed offset voltage to the VOUT_COMMAND value.

Table 23. Register 0x22—VOUT_TRIM

Bits	Bit Name	R/W	Description
[15:0]	Offset trim	R/W	Twos complement integer that applies a fixed offset voltage to the VOUT_COMMAND value.

VOUT_MARGIN_HIGH

The VOUT_MARGIN_HIGH command sets the margin high voltage. Exponent N is set using VOUT_MODE[4:0].

Table 24. Register 0x25—VOUT_MARGIN_HIGH

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R/W	16-bit unsigned Integer Y value for linear data format ($V = Y \times 2^N$). N is defined using VOUT_MODE[4:0].

VOUT_MARGIN_LOW

The VOUT_MARGIN_LOW command sets the margin low voltage. Exponent N is set using VOUT_MODE[4:0].

Table 25. Register 0x26—VOUT_MARGIN_LOW

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R/W	16-bit unsigned Integer Y value for linear data format ($V = Y \times 2^N$). N is defined using VOUT_MODE[4:0].

VOUT_SCALE_LOOP

The VOUT_SCALE_LOOP command sets the gain (K_R) by which the commanded voltage (V_{OUT}) is scaled to generate the internal reference voltage (V_{REF}). $V_{REF} = V_{OUT} \times K_R$, where $K_R = Y \times 2^N$.

Table 26. Register 0x29—VOUT_SCALE_LOOP

Bits	Bit Name	R/W	Description
[15:11]	Exponent N	R/W	Twos complement of Exponent N used in linear data format ($X = Y \times 2^N$).
[10:0]	Mantissa Y	R/W	Twos complement of Mantissa Y used in linear data format ($X = Y \times 2^N$).

VOUT_SCALE_MONITOR

The VOUT_SCALE_MONITOR command sets the gain (K_{VOUT}) by which the sensed output voltage at the device under test (DUT) (V_{OUT_DUT}) is scaled to generate the reading for the READ_VOUT command. READ_VOUT = $V_{OUT\_DUT} \times K_{VOUT}$, where $K_{VOUT} = Y \times 2^N$.

Table 27. Register 0x2A—VOUT_SCALE_MONITOR

Bits	Bit Name	R/W	Description
[15:11]	Exponent N	R/W	Twos complement of Exponent N used in linear data format ($X = Y \times 2^N$).
[10:0]	Mantissa Y	R/W	Twos complement of Mantissa Y used in linear data format ($X = Y \times 2^N$).

VOUT_OV_FAULT_LIMIT

The VOUT_OV_FAULT_LIMIT command sets the overvoltage threshold (in volts) measured at the sense/output pin, VHx/VPx, that causes an overvoltage fault condition. Exponent N is set using VOUT_MODE[4:0].

Table 28. Register 0x40—VOUT_OV_FAULT_LIMIT

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R/W	Unsigned Mantissa Y used in output voltage related commands in linear data format ($V = Y \times 2^N$).

VOUT_OV_WARN_LIMIT

The VOUT_OV_WARN_LIMIT command sets the overvoltage threshold (in volts) measured at the sense/output pin, VHx/VPx, that causes an overvoltage warning condition. Exponent N is set using VOUT_MODE[4:0].

Table 29. Register 0x42—VOUT_OV_WARN_LIMIT

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R/W	Unsigned Mantissa Y used in output voltage related commands in linear data format ($V = Y \times 2^N$).

VOUT_UV_WARN_LIMIT

The VOUT_UV_WARN_LIMIT command sets the undervoltage threshold (in volts) measured at the sense/output pin, VHx/VPx, that causes an undervoltage warning condition. Exponent N is set using VOUT_MODE[4:0].

Table 30. Register 0x43—VOUT_UV_WARN_LIMIT

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R/W	Unsigned Mantissa Y used in output voltage related commands in linear data format ($V = Y \times 2^N$).

VOUT_UV_FAULT_LIMIT

The VOUT_UV_FAULT_LIMIT command sets the undervoltage threshold value (in volts) measured at the sense/output pin, VHx/VPx, that causes an undervoltage fault condition. Exponent N is set using VOUT_MODE[4:0].

Table 31. Register 0x44—VOUT_UV_FAULT_LIMIT

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R/W	Unsigned Mantissa Y used in output voltage related commands in linear data format ($V = Y \times 2^N$).

STATUS_BYTE

The STATUS_BYTE command returns one byte of information with a summary of the most critical faults.

Table 32. Register 0x78—STATUS_BYTE

Bits	Bit Name	R/W	Description
[7:6]	Reserved	R	Reserved.
5	VOUT_OV_FAULT	R/W	V_{OUT} OV fault status.
[4:2]	Reserved	R	Reserved.
1	CML	R/W	Communication, memory, or logic event.
0	Reserved	R	Reserved.

STATUS_WORD

The STATUS_WORD command returns two bytes of information with a summary of the unit's fault condition.

Table 33. Register 0x79—STATUS_WORD

Bits	Bit Name	R/W	Description
15	VOUT	R/W	Logic OR of STATUS_VOUT, Bits[7:0].
[14:6]	Reserved	R	Reserved.
5	VOUT_OV_FAULT	R/W	V_{OUT} OV fault status.
[4:2]	Reserved	R	Reserved.
1	CML	R/W	Communication, memory, logic.
0	Reserved	R	Reserved.

STATUS_VOUT

The STATUS_VOUT command obtains the status of the rail comparators.

Table 34. Register 0x7A—STATUS_VOUT

Bits	Bit Name	R/W	Description
7	VOUT_OV_FAULT	R	V_{OUT} OV fault status.
6	VOUT_OV_WARNING	R	V_{OUT} OV warning status.
5	VOUT_UV_WARNING	R	V_{OUT} UV warning status.
4	VOUT_UV_FAULT	R	V_{OUT} UV fault status.
[3:0]	Reserved	R	Reserved.

STATUS_CML

The STATUS_CML command returns one data byte with contents as described in Table 35.

Table 35. Register 0x7E—STATUS_CML

Bits	Bit Name	R/W	Description
7	INVALID_COMMAND	R	Invalid or unsupported command received.
6	Reserved	R	Reserved.
5	PEC_ERROR	R	PEC failed.
4	MEMORY_FAULT_DETECTED	R	Memory fault detected.
[3:0]	Reserved	R	Reserved.

STATUS_MFR_SPECIFIC

The STATUS_MFR_SPECIFIC command returns one data byte with contents as described in Table 36.

Table 36. Register 0x80—STATUS_MFR_SPECFIC

Bits	Bit Name	R/W	Description
[7:6]	Reserved	R	Reserved.
5	ALL_CRC_FAULT	R	0 means all CRC checks passed, 1 means all CRC checks failed.
4	Reserved	R	Reserved.
3	RUNNING_REFRESH	R	0 means refresh is complete, 1 means refresh is running.
2	PART_LOCKED	R	0 means device is unlocked, 1 means device is locked.
1	PART_DATA_COMPATIBLE	R	0 means settings and sequence data is compatible, 1 means settings and sequence data are not compatible.
0	SILICON_COMPATIBLE	R	0 means silicon version check passed, 1 means silicon version check failed.

READ_VOUT

The READ_VOUT command returns the actual, measured output voltage, $V = Y \times 2^N$. Exponent N is set using VOUT_MODE[4:0].

Table 37. Register 0x8B—READ_VOUT

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R	Unsigned Mantissa Y used in output voltage related commands in linear data format ($V = Y \times 2^N$).

PMBUS_REVISION

The PMBUS_REVISION command returns the PMBus version information. The ADM1266 is compliant with PMBus Revision 1.2. Reading this command results in a value of 0x22.

Table 38. Register 0x98—PMBUS_REVISION

Bits	Bit Name	R/W	Description
[7:4]	Part 1 revision	R	Compliant to PMBus Part 1 specification: 0010 = Revision 1.2.
[3:0]	Part 2 revision	R	Compliant to PMBus Part 2 specification: 0010 = Revision 1.2.

MFR_ID

The MFR_ID command either sets or reads the manufacturer ID. MFR_ID is typically set only once, at the time of manufacture. The maximum length of the ID is 32 bytes.

Table 39. Register 0x99—MFR_ID (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	ID length	Block W	The length of the manufacturer ID data to read back. Maximum = 32 bytes.
0	Data length	Block R	The length of the manufacturer ID data that the ADM1266 returns.
[64:1]	Data	Block R	Manufacturer ID data.

Table 40. Register 0x99—MFR_ID (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	The length of the manufacturer ID data to write. Maximum = 32 bytes.
[64:1]	Data	Block W	Manufacturer ID data.

MFR_MODEL

The MFR_MODEL command either sets or reads the manufacturer model number. MFR_MODEL is typically set only once, at the time of manufacture. The maximum length of the model number is 32 bytes.

Table 41. Register 0x9A—MFR_MODEL (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	ID length	Block W	The length in bytes of manufacturer model number to read back. Maximum = 32 bytes.
0	Data length	Block R	The length of the manufacturer model number that the ADM1266 returns.
[64:1]	Data	Block R	Manufacturer model data.

Table 42. Register 0x9A—MFR_MODEL (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	The length in bytes of manufacturer model number to write. Maximum = 32 bytes.
[64:1]	Data	Block W	Manufacturer model data.

MFR_REVISION

The MFR_REVISION command either sets or reads the manufacturer revision number. MFR_REVISION is typically set only once, at the time of manufacture. The maximum length of the revision number is 8 bytes.

Table 43. Register 0x9B—MFR_REVISION (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	ID length	Block W	The length of manufacturer revision number to read back. Maximum = 8 bytes.
0	Data length	Block R	The length of the manufacturer revision number that the ADM1266 returns
[64:1]	Data	Block R	Manufacturer revision data

Table 44. Register 0x9B—MFR_REVISION (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	The length of manufacturer revision number to write. Maximum = 8 bytes.
[64:1]	Data	Block W	Manufacturer's revision data

MFR_LOCATION

The MFR_LOCATION command either sets or reads the manufacturing location of the device. MFR_LOCATION is typically set only once, at the time of manufacture. The maximum length of the location is 48 bytes.

Table 45. Register 0x9C—MFR_LOCATION (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	ID length	Block W	The length of manufacturer location data to read back. Maximum = 48 bytes.
0	Data length	Block R	The length of the manufacturer location data that the ADM1266 returns.
[64:1]	Data	Block R	Manufacturer location data.

Table 46. Register 0x9C—MFR_LOCATION (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	The length of manufacturer location data to write. Maximum = 48 bytes.
[64:1]	Data	Block W	Manufacturer location data.

MFR_DATE

The MFR_DATE command either sets or reads the date the device was manufactured. MFR_DATE is typically set only once, at the time of manufacture. The maximum length of the date is 16 bytes.

Table 47. Register 0x9D—MFR_DATE (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	ID length	Block W	The length of manufacturer date to read back. Maximum = 16 bytes.
0	Data length	Block R	The length of the manufacturer date that the ADM1266 returns.
[64:1]	Data	Block R	Manufacturer date.

Table 48. Register 0x9D—MFR_DATE (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	The length of manufacturer date to write. Maximum = 16 bytes.
[64:1]	Data	Block W	Manufacturer date.

MFR_SERIAL

The MFR_SERIAL command either sets or reads the manufacturer serial number of the device. MFR_LOCATION is typically set only once, at the time of manufacture. The maximum length of the serial number is 32 bytes.

Table 49. Register 0x9E—MFR_SERIAL (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	ID length	Block W	The length of manufacturer serial data to read back. Maximum = 32 bytes.
0	Data length	Block R	The length of the manufacturer serial data that the ADM1266 returns.
[64:1]	Data	Block R	Manufacturer serial data.

Table 50. Register 0x9E—MFR_SERIAL (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	The length of manufacturer serial data to write. Maximum = 32 bytes.
[64:1]	Data	Block W	Manufacturer serial data

IC_DEVICE

The IC_DEVICE command returns the ID and device number of the ADM1266. The default values are 0x41, 0x12, and 0x66.

Table 51. Register 0xAD—IC_DEVICE_ID

Byte	Byte Name	R/W	Description
0	Data length	Block R	Parameter data length, fixed to 3.
[3:1]	Data	Block R	Return the IC ID and device number: 0x41, 0x12, and 0x66 in bootloader mode. Return the IC ID and device number: 0x42, 0x12, and 0x66 in normal mode.

IC_DEVICE_REV

The IC_DEVICE_REV command returns the ADM1266 firmware, bootloader, and chip revision.

Table 52. Register 0xAE—IC_DEVICE_REV in Normal Mode

Byte	Byte Name	R/W	Description
0	Data length	Block R	Parameter data length, fixed to 8.
[3:1]	Firmware revision	Block R	ADM1266 firmware revision, for example: 0x01, 0x08, 0x07 means Version 1.8.7.
[6:4]	Bootloader revision	Block R	ADM1266 bootloader revision, for example: 0x00, 0x00, 0x07 means Version 0.0.7.
[8:7]	Chip revision	Block R	ADM1266 chip revision, for example: 41 and 30 are ASCII B and 0, respectively.

Table 53. Register 0xAE—IC_DEVICE_REV in Bootloader Mode

Byte	Byte Name	R/W	Description
0	Data length	Block R	Parameter data length, fixed to 8.
[3:1]	Bootloader revision	Block R	ADM1266 bootloader revision, for example: 0x01, 0x08, 0x07 means Version 1.8.7.
[6:4]	Reserved	Block R	Reserved
[8:7]	Chip revision	Block R	ADM1266 chip revision, for example: 41 and 30 are ASCII B and 0, respectively.

VOUT_OV_HYST_LIMIT

The VOUT_OV_HYST_LIMIT command either sets or reads the overvoltage hysteresis (in volts) measured at the sense/output pin that causes an overvoltage fault condition. The exponent N is set using VOUT_MODE[4:0].

Table 54. Register 0xD0—VOUT_OV_HYST_LIMIT

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R/W	Unsigned Mantissa Y used in output voltage related commands in linear data format ($V = Y \times 2^N$).

VOUT_UV_HYST_LIMIT

The VOUT_UV_HYST_LIMIT command either sets or reads the undervoltage hysteresis (in volts) measured at the sense/output pin that causes an undervoltage fault condition. The Exponent N is set using VOUT_MODE[4:0].

Table 55. Register 0xD1—VOUT_UV_HYST_LIMIT

Bits	Bit Name	R/W	Description
[15:0]	Mantissa Y	R/W	Unsigned Mantissa Y used in output voltage related commands in linear data format ($V = Y \times 2^N$).

Vx_CONFIGURATION

This command either writes or reads the VHx/VPx configuration for the device.

Table 56. Register 0xD2—VH_CONFIGURATION (for Page Command (0x00) Values of 0 to 3)

Bits	Bit Name	R/W	Description
[15:13]	VH_RANGE	R/W	VHx input range select. 000 = disconnected. 001 = disconnected. 010 = 6.0 V to 15.0 V. 011 = 3.0 V to 7.5 V. 100 = 1.5 V to 3.75 V. 101 = 0.75 V to 1.875 V. 110 = direct range (0.4 V to 1.0 V). 111 = reserved.
12	Reserved	R	Reserved.
[11:8]	VH_UV_FILTER	R/W	VHx UV glitch filter setting. Pulses smaller than this width are suppressed. Sampling delays may add up to 400 ns of additional delay. 0000 = reserved. 0001 = reserved. 0010 = 2.0 µs. 0011 = 4.0 µs. 0100 = 5.0 µs. 0101 = 6.0 µs. 0110 = 7.5 µs. 0111 = 8.0 µs. 1000 = 10.0 µs. 1001 = 20.0 µs. 1010 = 40.0 µs. 1011 = 50.0 µs. 1100 = 60.0 µs. 1101 = 75.0 µs. 1110 = 80.0 µs. 1111 = 100.0 µs.
7	Reserved	R/W	Reserved.
6	VH_UV_ENABLE	R/W	VHx UV comparator enable. 0 = disable UV comparator. 1 = enable UV comparator.
[5:2]	VH_OV_FILTER	R/W	VHx OV glitch filter setting. Pulses smaller than this width are suppressed. Sampling delays may add up to 400 ns of additional delay. 0000 = reserved. 0001 = reserved. 0010 = 2.0 µs. 0011 = 4.0 µs. 0100 = 5.0 µs. 0101 = 6.0 µs. 0110 = 7.5 µs. 0111 = 8.0 µs. 1000 = 10.0 µs. 1001 = 20.0 µs. 1010 = 40.0 µs. 1011 = 50.0 µs. 1100 = 60.0 µs. 1101 = 75.0 µs. 1110 = 80.0 µs. 1111 = 100.0 µs.

Bits	Bit Name	R/W	Description
1	Reserved	R/W	Reserved.
0	VH_OV_ENABLE	R/W	VHx OV comparator enable. 0 = disable OV comparator. 1 = enable OV comparator.

Table 57. Register 0xD2—VP_CONFIGURATION (for Page Command (0x00) Values of 4 to 16)

Bits	Bit Name	R/W	Description
[15:13]	VP_RANGE	R/W	VPx input range select. 000 = disconnected. 001 = disconnected. 010 = disconnected. 011 = 2.2 V to 5 V. 100 = 1.5 V to 3.75 V. 101 = 0.75 V to 1.875 V. 110 = direct range (0.4 V to 1.0 V). 111 = reserved.
12	VP_DIFF_EN	R/W	VPx differential mode select. 0 = disable differential voltage mode. 1 = enable differential mode (ignored on even pins).
[11:8]	VP_UV_FILTER	R/W	VPx UV glitch filter setting. Pulses smaller than this width are suppressed. Sampling delays may add up to 400 ns of additional delay. 0000 = reserved. 0001 = reserved. 0010 = 2.0 µs. 0011 = 4.0 µs. 0100 = 5.0 µs. 0101 = 6.0 µs. 0110 = 7.5 µs. 0111 = 8.0 µs. 1000 = 10.0 µs. 1001 = 20.0 µs. 1010 = 40.0 µs. 1011 = 50.0 µs. 1100 = 60.0 µs. 1101 = 75.0 µs. 1110 = 80.0 µs. 1111 = 100.0 µs.
7	Reserved	R/W	Reserved.
6	VP_UV_ENABLE	R/W	VPx UV comparator enable. 0 = disable UV comparator. 1 = enable UV comparator.

Bits	Bit Name	R/W	Description
[5:2]	VP_OV_FILTER	R/W	VPx OV glitch filter setting. Pulses smaller than this width are suppressed. Sampling delays may add up to 400 ns of additional delay. 0000 = reserved. 0001 = reserved. 0010 = 2.0 µs. 0011 = 4.0 µs. 0100 = 5.0 µs. 0101 = 6.0 µs. 0110 = 7.5 µs. 0111 = 8.0 µs. 1000 = 10.0 µs. 1001 = 20.0 µs. 1010 = 40.0 µs. 1011 = 50.0 µs. 1100 = 60.0 µs. 1101 = 75.0 µs. 1110 = 80.0 µs. 1111 = 100.0 µs.
1	Reserved	R/W	Reserved
0	VP_OV_ENABLE	R/W	VPx OV comparator enable. 0 = disable OV comparator. 1 = enable OV comparator.

BLACKBOX_CONFIGURATION

The BLACKBOX_CONFIGURATION command either sets or reads the cyclic black box record configuration.

Table 58. Register 0xD3—BLACKBOX_CONFIGURATION

Bits	Bit Name	R/W	Description
[15:1]	Reserved	R	Reserved.
0	CYCLIC_RECORD	R/W	Cyclic record mode. 0 = disabled. 1 = enabled

PDIO_CONFIGURATION

This command either block writes or reads the PDIOx configuration.

Table 59. Register 0xD4—PDIO_CONFIGURATION (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	PDIO index parameter	Block W	00000 = PDIO1. 00001 = PDIO 2. 00010 = PDIO 3. 00011 = PDIO 4. 00100 = PDIO5. 00101 = PDIO6. 00110 = PDIO7. 00111 = PDIO8. 01000 = PDIO9. 01001 = PDIO10. 01010 = PDIO11. 01011 = PDIO12.

Byte	Byte Name	R/W	Description
			01100 = PDIO13. 01101 = PDIO14. 01110 = PDIO15. 01111 = PDIO16. Setting the byte to 0xFF means the device reads back data for all PDIOs.
0	Data length	Block R	The length of the PDIO configuration data that the ADM1266 returns. Set to 2 when PDIO index parameter < 16, and 32 when PDIO index parameter is 0xFF.
[N:1]	Data	Block R	Configuration data for PDIOx.

Table 60. Register 0xD4—PDIO_CONFIGURATION (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write. For a write, the data length is from 3 to 33.
1	Starting index	Block W	00000 = PDIO1. 00001 = PDIO2. 00010 = PDIO3. 00011 = PDIO4. 00100 = PDIO5. 00101 = PDIO6. 00110 = PDIO7. 00111 = PDIO8. 01000 = PDIO9. 01001 = PDIO10. 01010 = PDIO11. 01011 = PDIO12. 01100 = PDIO13. 01101 = PDIO14. 01110 = PDIO15. 01111 = PDIO16. This value together with data length determine which PDIO is configured. For example; if starting index is 5 and data length is 7, PDIO6, PDIO7, and PDIO8 are configured.
[33:2]	Data	Block W	Data for PDIOx configuration.

Table 61. Two Bytes of Data for Each PDIOx Configuration

Bits	Bit Name	R/W	Description
[15:13]	PDIO_PIN_CFG	R/W	Operating mode for PDIOx pin. 000 = disabled. 001 = output. 010 = input. 011 = input/output. 100 = disabled. 101 = invalid. 110 = invalid. 111 = invalid.
[12:9]	PDIO_GLITCH_FILT	R/W	Input glitch filter setting. Pulses smaller than this width are suppressed. 0000 = 500 ns. 0001 = 1.0 µs. 0010 = 2.0 µs. 0011 = 4.0 µs. 0100 = 5.0 µs. 0101 = 6.0 µs. 0110 = 7.5 µs.

Bits	Bit Name	R/W	Description
			0111 = 8.0 µs. 1000 = 10.0 µs. 1001 = 20.0 µs. 1010 = 40.0 µs. 1011 = 50.0 µs. 1100 = 60.0 µs. 1101 = 75.0 µs. 1110 = 80.0 µs. 1111 = 100.0 µs.
[8:3]	Reserved	R/W	Reserved.
[2:0]	PDIO_OUTPUT_CFG	R/W	Output configuration. Sets the configuration of the PDIOx output drivers. 000 = 20 kΩ pull-down resistor. The resistor is enabled even during power-up. 001 = 20 kΩ pull-up resistor to AVDD. The resistor is enabled even during power-up. 010 = open source with 20 kΩ pull-down resistor. 011 = open drain with 20 kΩ pull-up resistor to AVDD. 100 = open source (requires external pull-down resistor). 101 = open drain (requires external pull-up resistor). 110 = push/pull output driver. 111 = high-Z.

DAC_CONFIGURATION

This command either block writes or reads the DAC configuration.

Table 62. Register 0xD5—DAC_CONFIGURATION (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	DAC index parameter	Block W	00000 = DAC1. 00001 = DAC2. 00010 = DAC3. 00011 = DAC4. 00100 = DAC5. 00101 = DAC6. 00110 = DAC7. 00111 = DAC8. 01000 = DAC9. Setting this byte to 0xFF means the device reads back data for all DACs.
0	Data length	Block R	The length of the DAC configuration data that ADM1266 returns. Set to 2 when DAC index parameter < 9, 18 when DAC index parameter is 0xFF.
[18:1]	Data	Block R	Data for DAC configuration.

Table 63. Register 0xD5—DAC_CONFIGURATION (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write. For a write, this value is from 3 to 19.
1	Starting Index	Block W	00000 = DAC1. 00001 = DAC2. 00010 = DAC3. 00011 = DAC4. 00100 = DAC5. 00101 = DAC6. 00110 = DAC7. 00111 = DAC8. 01000 = DAC9. This value together with data length determine which DAC is configured. For example; if starting index is 5 and data length is 7, DAC6, DAC7, and DAC8 are configured.
[19:2]	Data	Block W	Data for DAC configuration.

Table 64. Two Bytes of Data for Each DAC Configuration

Bits	Bit Name	R/W	Description
[15:11]	Reserved	R	Reserved.
[10:6]	DAC_MAPPING	R/W	These bits map the input pin that sets the DAC voltage for closed-loop margining. 00000 = open. 00001 = VH1. 00010 = VH2. 00011 = VH3. 00100 = VH4. 00101 = VP1. 00110 = VP2. 00111 = VP3. 01000 = VP4. 01001 = VP5. 01010 = VP6. 01011 = VP7. 01100 = VP8. 01101 = VP9. 01110 = VP10. 01111 = VP11. 10000 = VP12. 10001 = VP13
5	DAC_CLOSED_LOOP	R/W	This sets the two different closed-loop behaviors. 0 = closed loop is on continuously. 1 = closed loop regulates to the setpoint and stops.
[4:2]	Reserved	R/W	Reserved.
[1:0]	MARGIN_MODE	R/W	Margin mode. 00 = off. 01 = open loop. 10 = closed loop. 11 = reserved.

SEQUENCE_CONFIGURATION

This command either block writes or reads the sequence configuration.

Table 65. Register 0xD6—SEQUENCE_CONFIGURATION (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 3.
1	Data length	Block W	Data length.
2	Offset address (low)	Block W	The low 8-bit offset address of total configuration data.
3	Offset address (high)	Block W	The high 8-bit offset address of total configuration data.
0	Data length	Block R	Readback sequence configuration data length, maximum value of N = 252. N must be a multiple value of 4.
[252:1]	Data	Block R	Sequence configuration data.

Table 66. Register 0xD6—SEQUENCE_CONFIGURATION (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write.
1	Offset address (low)	Block W	The low 8-bit offset address of total configuration data.
2	Offset address (high)	Block W	The high 8-bit offset address of total configuration data.
[250:3]	Data	Block W	Data for sequence configuration, maximum value of N = 250. N − 3 + 1 must be a multiple value of 4.

SYSTEM_CONFIGURATION

This command either block writes or reads the system configuration.

Table 67. Register 0xD7—SYSTEM_CONFIGURATION (for Block WR)

Byte	Bit Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 3.
1	Data length	Block W	Data length.
2	Offset address (low)	Block W	The low 8-bit offset address of total configuration data.
3	Offset address (high)	Block W	The high 8-bit offset address of total configuration data.
0	Data length	Block R	Readback system configuration data length, maximum value of N = 252. N must be a multiple value of 4.
[252:1]	Data	Block R	System configuration data.

Table 68. Register 0xD7—SYSTEM_CONFIGURATION (for Block W)

Byte	Bit Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write.
1	Offset address (low)	Block W	The low 8-bit offset address of total configuration data.
2	Offset address (high)	Block W	The high 8-bit offset address of total configuration data.
[250:3]	Data	Block W	Data for system configuration, maximum value of N = 250. N − 3 + 1 must be a multiple value of 4.

GO_COMMAND

This command triggers various functions.

Table 69. Register 0xD8—GO_COMMAND

Bits	Bit Name	R/W	Description
[15:5]	Reserved	R	Reserved.
4	Seamless reset	W	Writing 1 to this bit enables seamless reset of the sequence by jumping to the PGOOD state. Writing 0 to this bit disables seamless reset of the sequence by jumping to State 0.
3	SEQUENCE_MODE	W	Writing 1 to this bit enables sequence debug mode. Writing 0 to this bit enables sequence normal mode.
2	Hardware reset	W	Writing 1 to this bit resets the CPU.
1	SEQUENCE_RESET	W	Writing 1 to this bit resets the sequence. Writing 0 to this bit does not reset the sequence.
0	Run/stop	W	Writing 1 to this bit stops the sequence. Writing 0 to this bit runs the sequence.

READ_STATE

The READ_STATE command returns the current value of the state bit that the sequencer is executing.

Table 70. Register 0xD9—READ_STATE

Bits	Bit Name	R/W	Description
[15:0]	State	R	Current state number that the sequencer is executing.

VOUT_MARGIN_LOOP

The VOUT_MARGIN_LOOP command either sets or reads the gain (R1/R3), which is used to calculate the V_{FB} (feedback voltage) according to the DAC output and V_{OUT}. For the relationship of V_{FB}, V_{OUT}, and the output of DAC, see Figure 12.

Table 71. Register 0xDA—VOUT_MARGIN_LOOP

Bits	Bit Name	R/W	Description
[15:11]	Exponent N	R/W	Twos complement Exponent N used in linear data format ($X = Y \times 2^N$).
[10:0]	Mantissa Y	R/W	Twos complement Mantissa Y used in linear data format ($X = Y \times 2^N$).

MARGIN_CONFIGURATION

The MARGIN_CONFIGURATION command either sets or reads the ramp step in margining.

Table 72. Register 0xDB—MARGIN_CONFIGURATION

Bits	Bit Name	R/W	Description
[15:12]	Reserved	R	Reserved.
[11:8]	RAMP_INTERVAL	R/W	Ramp interval time, interval: 0.1 ms.
[7:0]	RAMP_STEP	R/W	Number of DAC codes to increment for each step.

BREAKPOINTS

This command either block writes or reads the breakpoints for the sequence states.

Table 73. Register 0xDC—BREAKPOINTS (Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Number of bytes of parameter data, fix to 1.
1	Data length	Block W	The length of breakpoints data to read back. Maximum = 128 bytes.
0	Data length	Block R	The length of the breakpoints data that the ADM1266 returns.
[128:1]	Data	Block R	Bit 0 of Byte 1 sets the breakpoint for State 1, Bit 1 of Byte 1 sets the breakpoint for State 2, incrementing up to Bit 7 of Byte 128 sets the breakpoint for State 1024. Maximum value of N = 128.

Table 74. Register 0xDC—BREAKPOINTS (Block W)

Byte	Bit Name	R/W	Description
0	Data length	Block W	The length of breakpoints data to write. Maximum = 64 bytes.
[128:1]	Data	Block W	Bit 0 of Byte 1 sets the breakpoint for State 1, Bit 1 of Byte 1 sets the breakpoint for State 2, incrementing up to Bit 7 of Byte 64 sets the breakpoint for State 1024. Maximum value of N = 128.

ICB_CONFIGURATION

This command either writes or reads the IDB configurations.

Table 75. Register 0xDD—ICB_CONFIGURATION (Block R/W)

Byte	Bit Name	R/W	Description
0	Data length	Block R/W	Number of bytes of data for this block read/write, fixed to 8.
[8:1]	Data	Block R/W	Data for ICB configuration.

READ_BLACKBOX

This command reads back the black box record or erases the black box memory.

Table 76. Register 0xDE—READ_BLACKBOX (for Block WR)

Byte	Bit Name	R/W	Description
0	Parameter length	Block W	Number of bytes of data for this block write, fixed to 1.
1	Index	Block W	Black box record index.
0	Data length	Block R	The length of the black box data that the ADM1266 returns.
[64:1]	Data	Block R	Data for one black box record.

Table 77. Register 0xDE—READ_BLACKBOX (for Block W)

Byte	Bit Name	R/W	Description
0	Parameter data length	Block W	Number of bytes of data for this block write. This byte is fixed to 2 to erase the black box memory.
[2:1]	Parameter	Block W	To erase, set Byte 1 as 0xFE and Byte 2 as 0x00.

Table 78. Black Box Data Format

Byte	Field	Description
[1:0]	ID	Each black box record has a unique ID but that ID is the same for all black box records across multiple devices.
2	Empty	0 = used, 1 = empty.
	Reserved	Reserved.
	Page	Page index that current record is saved in.
	JUMP_TYPE	The jump (transition from one state to another) was due to sequence action, receipt of jump message.
3	ACTION_INDEX	Black box action index.
4	RULE_INDEX	Black box action rule index.
5	VHx_OV_STATUS	Overvoltage status of the VHx pins. Mapping of the VHx pins are shown in Table 79.
	VHx_UV_STATUS	Undervoltage status of the VHx pins. Mapping of the VHx pins are shown in Table 79.
[7:6]	CURRENT_STATE	The state in which the black box write was triggered.
[9:8]	LAST_STATE	The state from which the black box write was entered.
[11:10]	VP_OV_STATUS	Overvoltage status of the VPx pins. Mapping of the VPx pins are shown in Table 80.
[13:12]	VP_UV_STATUS	Undervoltage status of the VPx pins. Mapping of the VPx pins are shown in Table 80.
[15:14]	GPIO_IN_STATUS	Input status of GPIOx pins. Mapping of the GPIOx pins are shown in Table 81.
[17:16]	GPIO_OUT_STATUS	Output status of GPIOx pins. Mapping of the GPIOx pins are shown in Table 81.
[19:18]	PDIO_IN_STATUS	Input status of PDIOx pins. Mapping of the PDIOx pins are shown in Table 82.
[21:20]	PDIO_OUT_STATUS	Output status of PDIOx pins. Mapping of the PDIOx pins are shown in Table 82.

Byte	Field	Description
[23:22]	POWERUP_COUNTER	Number of times the device is power cycled (powered on and off).
[31:24]	TIME_STAMP	The time when the black box record was triggered.
[62:32]	Reserved	Reserved.
63	CRC	Cyclic redundancy check for black box data integrity.

Table 79. VHx_OV_STATUS and VHx_UV_STATUS Mapping

Bit	Field	Description
0	VH1_OV	VH1 OV fault status
1	VH2_OV	VH2 OV fault status
2	VH3_OV	VH3 OV fault status
3	VH4_OV	VH4 OV fault status
4	VH1_UV	VH1 UV fault status
5	VH2_UV	VH2 UV fault status
6	VH3_UV	VH3 UV fault status
7	VH4_UV	VH4 UV fault status

Table 80. VPx_OV_STATUS and VPx_UV_STATUS Mapping

Bit	Field	Description
0	VP1_OV/UV	VP1 OV/UV fault status
1	VP2_OV/UV	VP2 OV/UV fault status
2	VP3_OV/UV	VP3 OV/UV fault status
3	VP4_OV/UV	VP4 OV/UV fault status
4	VP5_OV/UV	VP5 OV/UV fault status
5	VP6_OV/UV	VP6 OV/UV fault status
6	VP7_OV/UV	VP7 OV/UV fault status
7	VP8_OV/UV	VP8 OV/UV fault status
8	VP9_OV/UV	VP9 OV/UV fault status
9	VP10_OV/UV	VP10 OV/UV fault status
10	VP11_OV/UV	VP11 OV/UV fault status
11	VP12_OV/UV	VP12 OV/UV fault status
12	VP13_OV/UV	VP13 OV/UV fault status
[13:15]	Reserved	Reserved

Table 81. GPIO_IN_STATUS and GPIO_OUT_STATUS Mapping

Bit	Field	Description
0	GPIO1_IN/OUT_STATUS	GPIO1 input/output status
1	GPIO2_IN/OUT_STATUS	GPIO2 input/output status
2	GPIO3_IN/OUT_STATUS	GPIO3 input/output status
[3:5]	Reserved	Reserved
6	GPIO8_IN/OUT_STATUS	GPIO8 input/output status
7	GPIO9_IN/OUT_STATUS	GPIO9 input/output status
8	GPIO4_IN/OUT_STATUS	GPIO4 input/output status
9	GPIO5_IN/OUT_STATUS	GPIO5 input/output status
10	GPIO6_IN/OUT_STATUS	GPIO6 input/output status
11	GPIO7_IN/OUT_STATUS	GPIO7 input/output status
[12:15]	Reserved	Reserved

Table 82. PDIO_IN_STATUS and PDIO_OUT_STATUS Mapping

Bit	Field	Description
0	PDIO1_IN/OUT_STATUS	PDIO1 input/output status
1	PDIO2_IN/OUT_STATUS	PDIO2 input/output status
2	PDIO3_IN/OUT_STATUS	PDIO3 input/output status
3	PDIO4_IN/OUT_STATUS	PDIO4 input/output status
4	PDIO5_IN/OUT_STATUS	PDIO5 input/output status
5	PDIO6_IN/OUT_STATUS	PDIO6 input/output status
6	PDIO7_IN/OUT_STATUS	PDIO7 input/output status
7	PDIO8_IN/OUT_STATUS	PDIO8 input/output status
8	PDIO9_IN/OUT_STATUS	PDIO9 input/output status
9	PDIO10_IN/OUT_STATUS	PDIO10 input/output status
10	PDIO11_IN/OUT_STATUS	PDIO11 input/output status
11	PDIO12_IN/OUT_STATUS	PDIO12 input/output status
12	PDIO13_IN/OUT_STATUS	PDIO13 input/output status
13	PDIO14_IN/OUT_STATUS	PDIO14 input/output status
14	PDIO15_IN/OUT_STATUS	PDIO15 input/output status
15	PDIO16_IN/OUT_STATUS	PDIO16 input/output status

SET_RTC

This command reads/writes the timestamp from/to the device by the GUI.

Table 83. Register 0xDF—SET_RTC

Byte	Bit Name	R/W	Description
0	Data length	Block R/W	Size of data for this block read/write, fixed to 6.
[7:1]	Data	Block R/W	6-byte timestamp message. Each LSB represents $1/(2^{16})$ sec if using all 6 bytes. For LSB in 1 sec size, set Byte 1 and Byte 2 to zero and the rest of the time in Byte[3:7].

LOGIC_CONFIGURATION

This command block reads/writes the logic configuration for the device.

Table 84. Register 0xE0—LOGIC_CONFIGURATION (for Block WR)

Byte	Bit Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 3.
1	Data length	Block W	Data length.
2	Offset address (low)	Block W	The low 8-bit offset address of total logic data.
3	Offset address (high)	Block W	The high 8-bit offset address of total logic data.
0	Data length	Block R	Readback logic data length, maximum value of N = 252. N must be a multiple value of 4.
[252:1]	Data	Block R	Logic data.

Table 85. Register 0xE0—LOGIC_CONFIGURATION (for Block W)

Byte	Bit Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write.
1	Offset address (low)	Block W	The low 8-bit offset address of total logic data.
2	Offset address (high)	Block W	The high 8-bit offset address of total logic data.
[250:3]	Data	Block W	Data for logic data, max value of N = 250. N − 3 + 1 must be a multiple value of 4.

GPIO_CONFIGURATION

This command block reads/writes the GPIO configuration.

Table 86. Register 0xE1—GPIO_CONFIGURATION (for Block RW)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	GPIO index parameter	Block W	GPIO index. Refer to Table 89.
0	Data length	Block R	The length of the GPIO configuration data that the ADM1266 returns, fixed to 2.
1	Data	Block R	Data for GPIO configuration.
2	Data	Block R	Reserved.

Table 87. Register 0xE1—GPIO_SYNC_CONFIGURATION (for Block W)

Byte	Bit Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write. For a write, its value is fixed to 2.
1	Index	Block W	GPIO_SYNC index. Refer to Table 89.
2	Data	Block W	Data for GPIO configuration.

Table 88. Data for Each GPIO_SYNC Configuration

Bits	Bit Name	R/W	Description
[7:4]	Reserved	R	Reserved.
4	Out mode	R/W	0 is push/pull, 1 is open drain
3	Output enable	R/W	0 is disable, 1 is enable
2	Input enable	R/W	0 is disable, 1 is enable
[1:0]	GPIO functions	R/W	GPIO functions, 00 is high-Z, 11 is GPIO

Table 89 shows the mapping between the internal GPIO index and the external GPIOx pins.

Table 89. GPIO Mapping

External GPIOx_SYNC Pin	GPIO Index
GPIO1	0
GPIO2	1
GPIO3	2
GPIO4	8
GPIO5	9
GPIO6	10
GPIO7	11
GPIO8	6
GPIO9	7
SYNC	5

USER_DATA

This command block reads/writes the user data.

Table 90. Register 0xE3—USER_DATA (for Block WR)

Byte	Bit Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 3.
1	Data length	Block W	Data length.
2	Offset address (low)	Block W	The low 8-bit offset address of total user data.
3	Offset address (high)	Block W	The high 8-bit offset address of total user data.
0	Data length	Block R	Read back user data length, maximum value of N = 252. N must be a multiple value of 4.
[N:1]	Data	Block R	User data.

Table 91. Register 0xE3— USER_DATA (for Block W)

Byte	Bit Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write.
1	Offset address (low)	Block W	The low 8-bit offset address of total user data.
2	Offset address (high)	Block W	The high 8-bit offset address of total user data.
[N:3]	Data	Block W	Data for user data, maximum value of N = 250.

POWERUP_COUNTER

This command reads the power-up counter from the device by the GUI.

Table 92. Register 0xE4—POWERUP_COUNTER

Byte	Bit Name	R/W	Description
0	Data length	Block W	The length of the power-up counter data that the ADM1266 returns, fixed to 2
[2:1]	Counter	Block W	Counter value

VOUT_RESISTOR

This command block reads/writes the resistor divider information.

Table 93. Register 0xE5—VOUT_RESISTOR (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 1.
1	Resistor index parameter	Block W	The index of the resistor is set. When the index is 0xFF, it reads back all resistor configurations. 0 = R4, 1 = R5, 2 = R1, 3 = R2, 4 = R3 (see Figure 12).
0	Data length	Block R	Readback configuration data length. Set to 3 when resistor index parameter < 5, 15 when resistor index parameter is 0xFF.
[15:1]	Data	Block R	Information for resistor.

Table 94. Register 0xE5—VOUT_RESISTOR (for Block W)

Byte	Bit Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write.
1	Resistor index parameter	Block W	The index of the resistor.
[5:2]	Data	Block W	Data for resistor, maximum value of N = 5.

Table 95. Three Bytes of Data for Each Resistor

Bits	Bit Name	R/W	Description
[23:16]	Exponent	R/W	Twos complement Exponent N used in linear data format ($X = Y \times 2^N$).
15	Reserved	R/W	Reserved.
[14:0]	Mantissa	R/W	Twos complement Mantissa Y used in linear data format ($X = Y \times 2^N$).

BLACKBOX_INFORMATION

This command reads the black box record counter and logic index.

Table 96. Register 0xE6—BLACKBOX_INFORMATION

Byte	Byte Name	R/W	Description
0	Data length	Block R	The length of the black box data that the ADM1266 returns, fixed to 4.
[2:1]	Black box ID	Block R	Latest black box ID.
3	Logic index	Block R	The latest black box record logic index.
4	Record count	Block R	The value of black box record count.

ALL_STATUS_VOUT

The ALL_STATUS_VOUT command returns all rails comparator status.

Table 97. Register 0xE7—ALL_STATUS_VOUT

Byte	Bit Name	R/W	Description
0	Data length	Block R	Readback status data length, fixed to 17.
1	STATUS_VH1	Block R	VH1 status V_{OUT}.
2	STATUS_VH2	Block R	VH2 status V_{OUT}.
3	STATUS_VH3	Block R	VH3 status V_{OUT}.
4	STATUS_VH4	Block R	VH4 status V_{OUT}.
5	STATUS_VP1	Block R	VP1 status V_{OUT}.
6	STATUS_VP2	Block R	VP2 status V_{OUT}.
7	STATUS_VP3	Block R	VP3 status V_{OUT}.
8	STATUS_VP4	Block R	VP4 status V_{OUT}.
9	STATUS_VP5	Block R	VP5 status V_{OUT}.
10	STATUS_VP6	Block R	VP6 status V_{OUT}.
11	STATUS_VP7	Block R	VP7 status V_{OUT}.
12	STATUS_VP8	Block R	VP8 status V_{OUT}.
13	STATUS_VP9	Block R	VP9 status V_{OUT}.
14	STATUS_VP10	Block R	VP10 status V_{OUT}.
15	STATUS_VP11	Block R	VP11 status V_{OUT}.
16	STATUS_VP12	Block R	VP12 status V_{OUT}.
17	STATUS_VP13	Block R	VP13 status V_{OUT}.

ALL_READ_VOUT

The ALL_READ_VOUT command returns all the output voltage value (V) in linear data format ($V = Y \times 2^N$). Exponent N is set using VOUT_MODE[4:0].

Table 98. Register 0xE8—ALL_READ_VOUT

Byte	Bit Name	R/W	Description
0	Data Length	Block R	Read back data length, fixed to 51.
[2:1]	MANTISSA_VH1	Block R	Mantissa of VH1.
[4:3]	MANTISSA_VH2	Block R	Mantissa of VH2.
[6:5]	MANTISSA_VH3	Block R	Mantissa of VH3.
[8:7]	MANTISSA_VH4	Block R	Mantissa of VH4.
[10:9]	MANTISSA_VP1	Block R	Mantissa of VP1.
[12:11]	MANTISSA_VP2	Block R	Mantissa of VP2.
[14:13]	MANTISSA_VP3	Block R	Mantissa of VP3.
[16:15]	MANTISSA_VP4	Block R	Mantissa of VP4.
[18:17]	Mantissa_VP5	Block R	Mantissa of VP5.
[20:19]	MANTISSA_VP6	Block R	Mantissa of VP6.
[22:21]	MANTISSA_VP7	Block R	Mantissa of VP7.

Byte	Bit Name	R/W	Description
[24:23]	MANTISSA_VP8	Block R	Mantissa of VP8.
[26:25]	MANTISSA_VP9	Block R	Mantissa of VP9.
[28:27]	MANTISSA_VP10	Block R	Mantissa of VP10.
[30:29]	MANTISSA_VP11	Block R	Mantissa of VP11.
[32:31]	MANTISSA_VP12	Block R	Mantissa of VP12.
[34:33]	MANTISSA_VP13	Block R	Mantissa of VP13.
35	VOUT_MODE_VH1	Block R	VOUT_MODE of VH1.
36	VOUT_MODE_VH2	Block R	VOUT_MODE of VH2.
37	VOUT_MODE_VH3	Block R	VOUT_MODE of VH3.
38	VOUT_MODE_VH4	Block R	VOUT_MODE of VH4.
39	VOUT_MODE_VP1	Block R	VOUT_MODE of VP1.
40	VOUT_MODE_VP2	Block R	VOUT_MODE of VP2.
41	VOUT_MODE_VP3	Block R	VOUT_MODE of VP3.
42	VOUT_MODE_VP4	Block R	VOUT_MODE of VP4.
43	VOUT_MODE_VP5	Block R	VOUT_MODE of VP5.
44	VOUT_MODE_VP6	Block R	VOUT_MODE of VP6.
45	VOUT_MODE_VP7	Block R	VOUT_MODE of VP7.
46	VOUT_MODE_VP8	Block R	VOUT_MODE of VP8.
47	VOUT_MODE_VP9	Block R	VOUT_MODE of VP9.
48	VOUT_MODE_VP10	Block R	VOUT_MODE of VP10.
49	VOUT_MODE_VP11	Block R	VOUT_MODE of VP11.
50	VOUT_MODE_VP12	Block R	VOUT_MODE of VP12.
51	VOUT_MODE_VP13	Block R	VOUT_MODE of VP13.

PDIO_STATUS

This command block reads the status of the PDIOs.

Table 99. Register 0xE9—PDIO_STATUS (for Block R)

Byte	Byte Name	R/W	Description
0	Data length	Block R	Number of bytes of data for this block read, fixed to 2.
[2:1]	PDIO status	Block R	Input or output status of all the PDIOs. Refer to Table 100.

Table 100. Two Bytes of Data for PDIO Status

Bits	Bit Name	R/W	Description
15	PDIO16_STATUS	R	PDIO16 pin status.
14	PDIO15_STATUS	R	PDIO15 pin status.
13	PDIO14_STATUS	R	PDIO14 pin status.
12	PDIO13_STATUS	R	PDIO13 pin status.
11	PDIO12_STATUS	R	PDIO12 pin status.
10	PDIO11_STATUS	R	PDIO11 pin status.
9	PDIO10_STATUS	R	PDIO10 pin status.
8	PDIO9_STATUS	R	PDIO9 pin status.
7	PDIO8_STATUS	R	PDIO8 pin status.
6	PDIO7_STATUS	R	PDIO7 pin status.
5	PDIO6_STATUS	R	PDIO6 pin status.
4	PDIO5_STATUS	R	PDIO5 pin status.
3	PDIO4_STATUS	R	PDIO4 pin status.
2	PDIO3_STATUS	R	PDIO3 pin status.
1	PDIO2_STATUS	R	PDIO2 pin status.
0	PDIO1_STATUS	R	PDIO1 pin status.

GPIO_STATUS

This command block reads the status of the GPIOs.

Table 101. Register 0xEA—GPIO_STATUS (for BLOCK R)

Byte	Byte Name	R/W	Description
0	Data length	Block R	Number of bytes of data for this block read, fixed to 2.
[2:1]	GPIO status	Block R	Input or output status of all the GPIOs. Refer to Table 102.

Table 102. Two Bytes of Data for GPIO_STATUS

Bits	Bit Name	R/W	Description
13	Reserved	R	Reserved.
12	Reserved	R	Reserved.
11	GPIO7_STATUS	R	GPIO7 pin status.
10	GPIO6_STATUS	R	GPIO6 pin status.
9	GPIO5_STATUS	R	GPIO5 pin status.
8	GPIO4_STATUS	R	GPIO4 pin status.
7	GPIO9_STATUS	R	GPIO9 pin status.
6	GPIO8_STATUS	R	GPIO8 pin status.
5	Reserved	R	Reserved.
4	Reserved	R	Reserved.
3	Reserved	R	Reserved.
2	GPIO3_STATUS	R	GPIO3 pin status.
1	GPIO2_STATUS	R	GPIO2 pin status.
0	GPIO1_STATUS	R	GPIO1 pin status.

DAC_CODE_CONFIGURATION

This command block reads/writes the DAC code in an open-loop margining configuration.

Table 103. Register 0xEB—DAC_CODE_CONFIGURATION (for Block WR)

Byte	Byte Name	R/W	Description
0	Parameter length	Block W	Parameter data length, fixed to 2.
1	DAC index parameter	Block W	0000 = DAC1. 0001 = DAC2. 0010 = DAC3. 0011 = DAC4. 0100 = DAC5. 0101 = DAC6. 0110 = DAC7. 0111 = DAC8. 1000 = DAC9.
2	Data length	Block W	The length of DAC code configuration data to read back from the ADM1266.
0	Data length	Block R	The length of DAC code configuration data the ADM1266 returns.
1	Code parameter	Block R	Refer to Table 105.
[17:2]	DAC code	Block R	DAC code. Maximum = 16 DAC codes, N = 17 maximum.

Table 104. Register 0xEB—DAC_CODE_CONFIGURATION (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	BLOCK W	Number of bytes of data to write to the ADM1266.
1	DAC index	BLOCK W	0000 = DAC1. 0001 = DAC2. 0010 = DAC3. 0011 = DAC4. 0100 = DAC5. 0101 = DAC6. 0110 = DAC7. 0111 = DAC8. 1000 = DAC9.
2	Code parameter	BLOCK W	Refer to Table 105.
[18:3]	DAC Code	BLOCK W	DAC code, maximum = 16 DAC codes, N = 18 maximum.

Table 105. One Byte of Data for Code Parameter

Bits	Bit Name	R/W	Description			
[7:4]	Code index	R/W	A maximum of 16 DAC codes can be programmed to the ADM1266. This index selects which code to load to the DAC.			
[3:1]	Range	R/W	DAC range.			
			Bits[3:1]	**Midcode Voltage (V)**	**Minimum Voltage Output (V)**	**Maximum Voltage Output (V)**
			0x00 = 3'b000	0.506	0.202	0.808
			0x01 = 3'b001	0.607	0.303	0.909
			0x02 = 3'b010	0.809	0.505	1.111
			0x03 = 3'b011	1.011	0.707	1.313
			0x04 = 3'b100	1.263	0.959	1.565
0	DAC enable	R/W	DAC enable/disable.			

RTS_CONFIGURATION

The RTS_CONFIGUARTION command either sets or reads the real time stamp (RTS) configuration.

Table 106. Register 0xEC—RTS_CONFIGURATION

Bits	Bit Name	R/W	Description
[15:2]	Reserved	R	Reserved.
1	RTS enable	R/W	0 is disable RTS, 1 is enable RTS.
0	XTAL enable	R/W	0 is external crystal oscillator disabled, 1 is external crystal oscillator enabled.

STATUS_MFR_SPECIFIC_2

The STATUS_MFR_SPECIFIC_2 command returns two bytes data with contents as shown in Table 107.

Table 107. Register 0xED—STATUS_MFR_SPECIFIC_2

Bits	Bit Name	R/W	Description
15	BKUP_PASSWORD_CRC_FAULT	R	0 means backup password CRC check passed, 1 means backup password CRC check failed.
14	BKUP_FIRMWARE_CRC_FAULT	R	0 means backup firmware CRC check passed, 1 means backup firmware CRC check failed.
13	BKUP_PROJECT_CRC_FAULT	R	0 means backup project CRC check passed, 1 means backup project CRC check failed.
12	BKUP_ABCONFIG_CRC_FAULT	R	0 means backup ABConfig CRC check passed, 1 means backup ABConfig CRC check failed.
11	MAIN_PASSWORD_CRC_FAULT	R	0 means main password CRC check passed, 1 means main password CRC check failed.
10	MAIN_FIRMWARE_CRC_FAULT	R	0 means main firmware CRC check passed, 1 means main firmware CRC check failed.
9	MAIN_PROJECT_CRC_FAULT	R	0 means main project CRC check passed, 1 means main project CRC check failed.
8	MAIN_ABCONFIG_CRC_FAULT	R	0 means main ABConfig CRC check passed, 1 means main ABConfig CRC check failed.
7	BKUP_IAP_CRC_FAULT	R	0 means backup IAP CRC check passed, 1 means the main IAP CRC check failed.
6	BKUP_MINI_IAP_CRC_FAULT	R	0 means backup mini IAP CRC check passed, 1 = main mini IAP CRC check failed.
5	MAIN_IAP_CRC_FAULT	R	0 means main IAP CRC check passed, 1 = main IAP CRC check failed.
4	MAIN_MINI_IAP_CRC_FAULT	R	0 means main mini IAP CRC check passed, 1 = main mini IAP CRC check failed.
3	AVDD_UVLO_FAULT	R	0 means no AVDD UVLO fault, 1 means AVDD UVLO fault occurred.
2	HARD_FAULT	R	0 means no hard fault, 1 means hard fault occurred.
1	AB_SYNC_FAULT	R	0 means sync pass, 1 means sync fail.
0	RUNNING_BACKUP_PROJECT	R	0 means main project, 1 means backup project.

REFRESH_CONFIGURATION

This command is used to set refresh configuration and get the refresh status.

Table 108. Register 0xF4—REFRESH_CONFIGURATION (for BLOCK R)

<table>
<tr><th>Byte</th><th>Byte Name</th><th>R/W</th><th colspan="3">Description</th></tr>
<tr><td>0</td><td>Parameter length</td><td>Block W</td><td colspan="3">Parameter data length—fixed to 1.</td></tr>
<tr><td>1</td><td>Parameter</td><td>Block W</td><td colspan="3">0x00: get auto-refresh interval.
0x01: get auto-refresh enabled/disabled status.
Others: get all status, including refresh enable/disable, autorefresh enable/disable, refresh times, recalculate CRC error times, autorefresh interval.</td></tr>
<tr><td>0</td><td>Data length</td><td>Block R</td><td colspan="3">Number of bytes of data for this block read (1 to 8).</td></tr>
<tr><td>[8:1]</td><td>Data</td><td>Block R</td><td colspan="3">Parameters:
0x00: get autorefresh interval, Bytes[2:1] are the autorefresh interval.
0x01: get autorefresh enabled/disabled status. If Byte 1 is 0, autorefresh is disabled.
Others: get all status, including refresh enable/disable, autorefresh enable/disable, refresh times, recalculate CRC error times, autorefresh interval, Bytes[8:1]:
<table>
<tr><th>Byte</th><th>Byte Name</th><th>Description</th></tr>
<tr><td>1</td><td>Refresh status</td><td>0: refresh is done.
1: refresh is running.</td></tr>
<tr><td>2</td><td>Autorefreshing</td><td>0: autorefresh is disabled.
1: autorefresh is enabled.</td></tr>
<tr><td>[4:3]</td><td>Refresh count</td><td>Refresh times.</td></tr>
<tr><td>[6:5]</td><td>Recalculate error count</td><td>Recalculate error times.</td></tr>
<tr><td>[8:7]</td><td>Autorefresh interval</td><td>Autorefresh interval, unit: day.</td></tr>
</table></td></tr>
</table>

Table 109. Register 0xF4—REFRESH_CONFIGURATION (for BLOCK W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write. The range is 2 to 3.
1	Configuration	Block W	0x00: set autorefresh interval, unit: day, Byte[3:2] is the interval. 0x01: enable/disable autorefresh. If Byte 2 is 0, disable autorefresh.
[3:2]	Data	Block W	Data

REFRESH_FLASH

This command is used to set refresh configuration .

Table 110. Register 0xF5—REFRESH_FLASH (for BLOCK W)

Byte	Byte Name	R/W	Description
[0]	Data Length	Block W	Number of bytes of data for this block write (fixed to 1).
[1]	Configuration	Block W	0x00: Project(including Reg, Sequence, User, System, Logic, Password, AB config) 0x01: Project + Firmware + IAP 0x02: Project + Firmware + IAP + mini IAP

HITLESS_TIMEOUT

This command is used to read or write the hitless timeout value.

Table 111. Register 0xF6—HITLESS_TIMEOUT

Byte	Byte Name	R/W	Description
0	Data length	Block R/W	Parameter data length, fixed to 2.
[2:1]	Timeout value	Block W	Timeout value; unit: sec.

VAR_VALUE

This command is used to read the variables value of sequence.

Table 112. Register 0xF7—VAR_VALUE

Byte	Byte Name	R/W	Description
0	Data length	Block W	Parameter data length; fixed to 1.
1	Index	Block W	Variables index, 0xFF for all values.
0	Data length	Block R	Size of data.
[N:1]	Value	Block R	Data. If the index is 0 to 3, N = 1. If the index is 0xFF, N = 4.

MEMORY_CONFIGURATION

This command reads/writes the main/backup memory configuration by the GUI.

Table 113. Register 0xF8—MEMORY_CONFIGURATION

Byte	Byte Name	R/W	Description
0	Data length	Block R/W	Size of data for this block read/write, fixed to 3.
[2:1]	Data	Block R/W	Main/backup memory configuration.
3	CRC	Block R/W	CRC-8 of main/backup memory configuration data.

MEMORY_RECALCULATE_CRC

This command recalculates both the main and backup memory CRC.

Table 114. Register 0xF9—MEMORY_RECALCULATE_CRC

Bits	Bit Name	R/W	Description
[15:0]	RECALCULATE_CRC	W	Write 100 (hexadecimal) to recalculate the CRC of all the sections of the memory

SWITCH_MEMORY

This command switches between the configure main memory and configure backup memory.

Table 115. Register 0xFA—SWITCH_MEMORY (for Block W)

Byte	Bit Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write, fixed to 1.
1	Memory index	Block W	Memory index. 0 for main memory, 1 for backup memory.

ERASE_MEMORY

This command erases the main memory or backup memory.

Table 116. Register 0xFB—ERASE_MEMORY (for Block W)

Byte	Bit Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write, fixed to 1.
1	Memory index	Block W	Memory index. 0 for main memory, 1 for backup memory.

UPDATE_FW

The command updates the firmware.

Table 117. Register 0xFC—UPDATE_FW

Bits	Bit Name	Type	Description
[15:0]	UPDATE_FW	W	Write 100 (hexadecimal) to jump to bootloader and start updating firmware

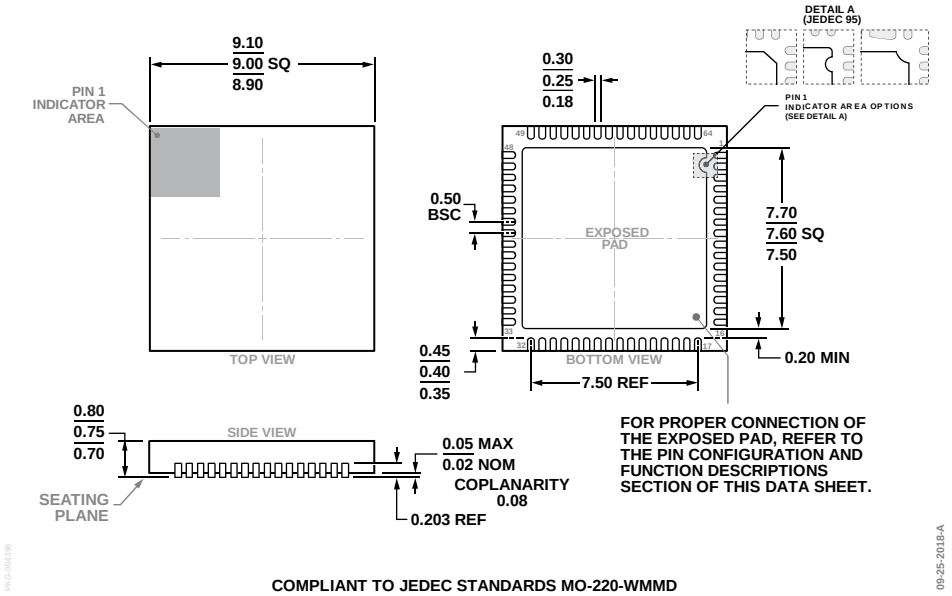
FW_PASSWORD

This command changes the password, locks/unlocks the device

Table 118. Register 0xFD—FW_PASSWORD (for Block W)

Byte	Byte Name	R/W	Description
0	Data length	Block W	Number of bytes of data for this block write, fixed to 17.
[1:16]	Password	Block W	16-byte password.
17	Command	Block W	Password command (in decimal code). 1 = change password. 2 = unlock device. 3 = lock device.

OUTLINE DIMENSIONS



COMPLIANT TO JEDEC STANDARDS MO-220-WMMD

Figure 30. 64-Lead Lead Frame Chip Scale Package [LFCSP]
9 mm × 9 mm Body and 0.75 mm Package Height
(CP-64-15)
Dimensions shown in millimeters

ORDERING GUIDE

Model[1]	Temperature Range	Package Description	Package Option
ADM1266ACPZ	−40°C to +105°C	64-Lead Lead Frame Chip Scale Package [LFCSP]	CP-64-15
ADM1266ACPZ-R7	−40°C to +105°C	64-Lead Lead Frame Chip Scale Package [LFCSP]	CP-64-15
ADM1266-EVALZ		Evaluation Board	

[1] Z = RoHS Compliant Part.

I²C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).

OCEAN CHIPS
Океан Электроники
Поставка электронных компонентов

Компания «Океан Электроники» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Поставка оригинальных импортных электронных компонентов напрямую с производств Америки, Европы и Азии, а так же с крупнейших складов мира;
- Широкая линейка поставок активных и пассивных импортных электронных компонентов (более 30 млн. наименований);
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Помощь Конструкторского Отдела и консультации квалифицированных инженеров;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Поставка электронных компонентов под контролем ВП;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- При необходимости вся продукция военного и аэрокосмического назначения проходит испытания и сертификацию в лаборатории (по согласованию с заказчиком);
- Поставка специализированных компонентов военного и аэрокосмического уровня качества (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Actel, Aeroflex, Peregrine, VPT, Syfer, Eurofarad, Texas Instruments, MS Kennedy, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Компания «Океан Электроники» является официальным дистрибьютором и эксклюзивным представителем в России одного из крупнейших производителей разъемов военного и аэрокосмического назначения «**JONHON**», а так же официальным дистрибьютором и эксклюзивным представителем в России производителя высокотехнологичных и надежных решений для передачи СВЧ сигналов «**FORSTAR**».

AVIC JONHON

«JONHON» (основан в 1970 г.)

Разъемы специального, военного и аэрокосмического назначения:

(Применяются в военной, авиационной, аэрокосмической, морской, железнодорожной, горно- и нефтедобывающей отраслях промышленности)

«FORSTAR» (основан в 1998 г.)

ВЧ соединители, коаксиальные кабели, кабельные сборки и микроволновые компоненты:

(Применяются в телекоммуникациях гражданского и специального назначения, в средствах связи, РЛС, а так же военной, авиационной и аэрокосмической отраслях промышленности).

FORSTAR®

Телефон: 8 (812) 309-75-97 (многоканальный)
Факс: 8 (812) 320-03-32
Электронная почта: ocean@oceanchips.ru
Web: http://oceanchips.ru/
Адрес: 198099, г. Санкт-Петербург, ул. Калинина, д. 2, корп. 4, лит. А