PE24103 “R2D2” PMBus Support

1 TABLE OF CONTENTS

2 TABLE OF FIGURES

3 TABLE OF TABLES

4 GLOSSARY & LIST OF ABBREVIATIONS

Table 1 List of Abbreviations Used in This Document

Abbreviation	Meaning	Description
ATB	Analog Test Bus	Defining an analog test bus is a common way of allowing production test, or bench debug, to measure or monitor internal analog nodes inside a chip.
ATPG	Automatic Test Pattern Generation	Commonly used for production testing of the digital parts of a chip to screen for manufacturing flaws.
BIST	Built-In Self-Test	On-chip circuitry which is used to help with some form of test. The testing may support mission-mode functions or bench/production test (or both).
CDC	Clock Domain Crossing	As part of the design process digital blocks with more than a single clock source would normally run a series of CDC checks to help make sure that the clock-to-clock logic has been analyzed and designed correctly.
CYA		Usually referring to a fallback option to help recover an unexpected state.
DC-DC	Direct Current to Direct Current	A DC-to-DC converter is an electronic circuit or electromechanical device that converts a source of Direct Current from one voltage level to another. It is a type of electric power converter. PE24103 is a DC-DC convertor.
DTB	Digital Test Bus	A digital test bus is often used to allow internal nodes within the heart of the digital block(s) to be passed to a chip pin in certain circumstances. Monitoring certain key internal nodes may help with production testing or bench debug.
eFUSE	Electrical fuse	eFUSE usually refers to a range of possible electrical circuit implementations. The key characteristic of the eFUSE is that the circuit is manufactured in one logic state and may be electrically programmed (on a bit-by-bit basic) to the opposite state. eFUSE may therefore be used for trim/configuration information.
GF	Global Foundries	External silicon wafer manufacturer.
GPIO	General Purpose Input Output	Typically used to refer to chip pins (and associated input/output circuits) that are capable of supporting digital operation.
HV	High Voltage	“High” is a relative term depending on the application. In the context of PE24103 “high” means circuitry connected to VIN which can range up to 8V nominal.

Table 1 List of Abbreviations Used in This Document

Abbreviation	Meaning	Description
I/F	Interface	Sometimes used as shorthand for a group of signals. For example the interface between the digital and bias blocks may consist of a number of independent signals. SystemVerilog has a formal definition for an interface which may be used in some digital designs.
I/O	Input/Output	Shorthand for the input/output signals of a block or the chip.
I/P	Input	Referring usually to a signal going into a block.
I^2C	Inter Integrated Circuit	I^2C is a commonly used serial data interface which runs through 2 pins (one for clock and one for data).
IDDQ	Quiescent supply current	IDDQ testing is often used for digital production testing in association with scan based test techniques. IDDQ testing requires that the digital supply current can be isolated and measured with high accuracy, but can offer good fault coverage for only a handful of test measurements.
IOS	Initial Outline Specification	
LDO	Low Dropout (Regulator)	An LDO is a linear voltage regulator that can regulate the output voltage even when the supply voltage is very close to the output voltage.
LEC	Logical Equivalence Checking	LEC is commonly used as part of the digital design flow to increase confidence that the various versions (RTL or gate level versions for example) of a digital design are equivalent.
NVM	Non-Volatile Memory	"Permanent" memory which retains value when power is cycled. Used on PE24103 for analog trim and device configuration.
O/P	Output	Referring usually to a signal coming out of a block.
OTP	One-Time Programmable	OTP is normally taken to mean a specific type of digital memory which is typically electrically programmed and erased with UV. It is one-time programmable because UV light cannot usually reach the surface of the die once the device is packaged.
P&R	Place and Route	Another way of referring to the digital block. Typically refers to the fact that the layout of the digital is auto-routed using layout tools rather than being laid out by hand.
PMBus	Power Management Bus	The PMBus specification defines a protocol to manage power convertors and/or a power system via communication over a digital communication bus.
POR	Power-On Reset	Many designs contain a dedicated POR block which ensures that the digital logic starts from a known state.
PSRR	Power Supply Rejection Ratio	PSRR is a common figure of merit used for IC design and describes the capability of an electronic circuit to isolate any input power supply variations from its output signal.

Table 1 List of Abbreviations Used in This Document

Abbreviation	Meaning	Description
PWM	Pulse-Width Modulation	PWM can be used to modulate the information (or power) content of a signal by varying the width of the individual pulses.
RC	Resistor-Capacitor	Simple, first order delays (or low pass filters) can be generated using a resistor and capacitor in series. This type of simple circuit is often referred to as an RC delay.
RTL	Register Transfer Logic	Usually refers to one (or more) high level languages which are used to describe the functionality of a digital block. Usually RTL refers to the part of the digital design which will become hardware on the die (rather than code used just for testing). SystemVerilog can be used for RTL.
SMBus	System Management Bus	The System Management Bus is a 2-wire, digital, serial interface through which various system components can communicate. It is based on the principles of operation of the I^2C bus. It is also used as the physical layer for PMBus.
SPI	Serial Peripheral Interface	SPI refers to a generic family of commonly used serial data interfaces. There are many minor variations of SPI, but in general SPI will use 3 or 4 pins (one for select, one for clock, one for data into the device and one for data out from the device).
STA	Static Timing Analysis	STA is used in digital design flows to help verify the correctness of a digital design. It is most useful to verify clocktree and correct physical design and provides a way of verifying a digital design without having to generate complex simulation waveforms.
TBD	To Be Decided	Used in this document to refer to various specification or functionality which has yet to be fully defined.
UV	Ultra Violet	In the context of this document used as in “UV light” which is commonly used as part of the semiconductor manufacturing flow. It is commonly used to create a default ‘erased’ state in some forms of NVM.

5 PMBus COMMANDS

5.1 Supported Commands

The list of PMBus commands proposed for PE24103 is based in part on the Application Profile for a DC-DC Point of Load (POL) device as published by the System Management Interface Forum. This document is not a specification in itself, merely a subset of commands commonly used for DC-DC convertors similar to PE24103. Table 2 lists the commonly used commands from the POL profile and whether they are supported by PE24103 or not.

Table 2 Common PMBus Commands For DC-DC POL Devices

Command Code	Command Name	# of Data Bytes per command	Supported for PE24103?	Does PE24103 Use PAGING With This Command?
00h	PAGE	1	Yes	No
01h	OPERATION	1	Yes	Yes
02h	ON_OFF_CONFIG	1	Yes	No
03h	CLEAR_FAULTS	0	Yes	Yes
11h	STORE_DEFAULT_ALL*	0	**No**	N/A
12h	RESTORE_DEFAULT_ALL*	0	**No**	N/A
15h	STORE_USER_ALL	0	Yes	No
16h	RESTORE_USER_ALL	0	Yes	No
19h	CAPABILITY	1	Yes	No
20h	VOUT_MODE	1	Yes	Yes
21h	VOUT_COMMAND	2	Yes	Yes
25h	VOUT_MARGIN_HIGH	2	Yes	Yes
26h	VOUT_MARGIN_LOW	2	Yes	Yes
35h	VIN_ON	2	Yes	No
36h	VIN_OFF	2	Yes	No
40h	VOUT_OV_FAULT_LIMIT	2	Yes	Yes
44h	VOUT_UV_FAULT_LIMIT	2	Yes	Yes
46h	IOUT_OC_FAULT_LIMIT	2	Yes	Yes
51h	OT_WARN_LIMIT	2	**No**	N/A
60h	TON_DELAY	2	Yes	Yes
61h	TON_RISE	2	Yes	Yes
64h	TOFF_DELAY	2	Yes	Yes
65h	TOFF_FALL	2	Yes	Yes
78h	STATUS_BYTE	1	Yes	Yes
79h	STATUS_WORD	2	Yes	Yes
7Ah	STATUS_VOUT	1	Yes	Yes
7Dh	STATUS_TEMPERATURE	1	Yes	No
7Eh	STATUS_CML	1	Yes	No
8Bh	READ_VOUT	2	Yes	Yes
8Ch	READ_IOUT	2	Yes	Yes
8Dh	READ_TEMPERATURE_1	2	Yes	No

Table 2 Common PMBus Commands For DC-DC POL Devices

Command Code	Command Name	# of Data Bytes per command	Supported for PE24103?	Does PE24103 Use PAGING With This Command?
98h	PMBUS_REVISION	1	Yes	No
99h	MFR_ID	Variable	Yes	No
ADh	IC_DEVICE_ID	Variable	Yes	No

* *It is clear that the User will be able to store the PMBus configuration into NVM of PE24103 and to restore that configuration on demand. However, there is probably not enough NVM space to store the factory default setting as well. If there were a factory default setting, then what would STORE_DEFAULT_ALL do?*

In addition to the commands described in Table 2, there are some additional PMBus commands which are supported by PE24103. These additional commands are listed in Table 3 along with a brief description of the command.

Table 3 Additional PMBus Commands Supported by PE24103

Command Code	Does PE24103 Use PAGING With This Command?	Command Name	# of Data Bytes per command	Comment
05h	Depends on which command code is used.	PAGE_PLUS_WRITE	Variable	PE24103 supports PAGES so this combined "meta-command" makes sense.
06h	Depends on which command code is used.	PAGE_PLUS_READ	Variable	PE24103 supports PAGES so this combined "meta-command" makes sense.
1Bh	Depends on which STATUS_xxx command code is used.	SMBALERT_MASK	2	PE24103 supports an SMBALERT pin so the masks for that pin make sense.
27h	Yes	VOUT_TRANSITION_RATE	2	As the PE24103 already has to support controlled slew rates at start-up and shut-down, the same circuits can also be used to support a controlled ramp rate when switching the output voltage from one level to another.
33h	No	FREQUENCY_SWITCH	2	PRD supports 3 possible frequencies for the BUCK outputs. This command allows selection of the 3 (fixed in hardware) switching frequencies. Warning, watch for implementation of this. For various other time related things we'd prefer not to alter the oscillator frequency at source…

Table 3 Additional PMBus Commands Supported by PE24103

Command Code	Does PE24103 Use PAGING With This Command?	Command Name	# of Data Bytes per command	Comment
4Bh	Yes	IOUT_UC_FAULT_LIMIT	2	PE24103 already measures IOUT and compares with IOUT_OC, so comparing with IOUT_UC is a small step.
4Fh	No	OT_FAULT_LIMIT	2	POL spec has the WARN limit already, but for consistency with the other fault flags we choose to support the over-temperature FAULT flag and not the WARN flag.
55h	No	VIN_OV_FAULT_LIMIT	2	PE24103 will measure VIN. Based on Cobra/PE25200 experience, it may be better to shutdown at some VIN overvoltage levels rather than attempt to operate through them.
59h	No	VIN_UV_FAULT_LIMIT	2	PE24103 will measure VIN. There should be support for either VIN_UV_FAULT or VIN_UV_WARN.
5Bh	No	IIN_OC_FAULT_LIMIT	2	PE24103 will measure IIN. There should be support for either IIN_OC_FAULT or IIN_OC_WARN.
5Eh	Yes	POWER_GOOD_ON	2	As VOUT will be measured via the ADC, having a variable set-point for PGOOD is not difficult.
5Fh	Yes	POWER_GOOD_OFF	2	As VOUT will be measured via the ADC, having a variable set-point for PGOOD is not difficult.
7Bh	Yes	STATUS_IOUT	1	PE24103 will measure IOUT for each output. Supporting a STATUS_IOUT value should not involve much overhead.
7Ch	No	STATUS_INPUT	1	PE24103 will measure VIN and IIN. Supporting a STATUS_INPUT value should not involve much overhead.
7Fh	No	STATUS_OTHER	1	PE24103 supports an SMBALERT pin and there is one bit in the STATUS_OTHER register which could be helpful for PE24103.
80h	No	STATUS_MFR_SPECIFIC	1	We want PE24103 to be able to (as a minimum) give the user information about the charge pump status. The manufacturer specific STATUS command will allow this.
88h	No	READ_VIN	2	PE24103 will measure VIN. Supporting a user read of that ADC result should not involve much overhead.
89h	No	READ_IIN	2	PE24103 will measure IIN. Supporting a user read of that ADC result should not involve much overhead.

Table 3 Additional PMBus Commands Supported by PE24103

Command Code	Does PE24103 Use PAGING With This Command?	Command Name	# of Data Bytes per command	Comment
A0h	No	MFR_VIN_MIN	2	This read-only value (coming from NVM) would provide the user data on the capability of PE24103. Expected value comes from the PRD (currently VIN UVLO = 9.0V).
A1h	No	MFR_VIN_MAX	2	This read-only value (coming from NVM) would provide the user data on the capability of PE24103. Expected value comes from the PRD (currently VIN range is up to 15.5V).
A2h	No	MFR_IIN_MAX	2	This read-only value (coming from NVM) would provide the user data on the capability of PE24103. Expected value Is not defined in the PRD (implied value from 4 x 4A x 2V output power = 32W, at ~80% efficiency means 4.2A input at 9.6V).
A4h	No	MFR_VOUT_MIN	2	This read-only value (coming from NVM) would provide the user data on the capability of PE24103. Expected value comes from the PRD (currently 0.6V).
A5h	No	MFR_VOUT_MAX	2	This read-only value (coming from NVM) would provide the user data on the capability of PE24103. Expected value comes from the PRD (currently 2.0V).
A6h	No	MFR_IOUT_MAX	2	This read-only value (coming from NVM) would provide the user data on the capability of PE24103. Expected value comes from the PRD (line regulation spec'd at up to 4A).
C4h	No	MFR_SPECIFIC_PGOOD	2	Unique command for PE24103 to allow definition of what the Power OK pin should mean for each of the 4 outputs. Provides some flexibility to logically AND/OR the VOUT status for various channels into the single POK pin.
C5h	No	MFR_SPECIFIC_MATRIX	2	Unique command for PE24103 to allow definition of how the board level BUCK outputs are combined to form higher capacity outputs. This is used to setup the compensation network in the analog part of PE24103.
C6h	Yes	MFR_SPECIFIC_IOUTFILT	2	Unique command for PE24103 to allow definition of how the ADC results for IOUT measurements are filtered (if at all).
C7h	Yes	MFR_SPECIFIC_OP_BEH	1	Unique command for PE24103 to allow control of various aspects of VOUT behavior not covered elsewhere.

Table 3 Additional PMBus Commands Supported by PE24103

Command Code	Does PE24103 Use PAGING With This Command?	Command Name	# of Data Bytes per command	Comment
C8h	No	MFR_SPECIFIC_MASK	2	Unique command to allow User control of how faults may affect the SMBAlert output. This supplements the existing standard PMBus mechanisms.
C9h	No	MFR_SPECIFIC_DCR_TC	2	Unique command to allow a temperature coefficient value to be added for the external inductor at each VOUT. This value affects the reported IOUT values.
CAh	No	MFR_SPECIFIC_RSENSE_TC	2	Unique command to allow a temperature coefficient value to be added for the external Rsense resistor (if fitted) at VIN. This value affects the reported IIN values.
CBh	Yes	MFR_IOUT_CAL_GAIN	2	Unique command for PE24103 to allow a manufacturer specific version of the IOUT_CAL_GAIN command.
CCh	No	MFR_IIN_CAL_GAIN	2	Unique command for PE24103 to allow a user to define the value of the IIN sense resistor.
CDh	No	MFR_CFG_PMBUS	1	Unique command for PE24103 to allow a user to define/change the PMBus device address value.

5.1.1 PAGE (Command Code 00h)

PE24103 supports the PAGE command and uses the PAGE value to allow control of the four switching outputs to be managed through a single PMBus device address. In general each of the four VOUT outputs (VOUT1, 2, 3 and 4) are associated with the corresponding PAGE number (1, 2, 3 or 4).

PE24103 also supports the pre-defined broadcast PAGE value of 0xFF which affects the values in all PAGES at the same time. Note that attempting to read a value using the broadcast PAGE value will result in indeterminate results (depending on the value being read) and is not recommended.

If a PAGE command is written using a PAGE value other than 0x01, 0x02, 0x03, 0x04 or 0xFF then the PAGE value will revert to 0x00 (which is unused in PE24103).

PE24103 supports 5 page values in total which are used to control various configurations of the switching outputs. Figure 1 shows the configurations that PE24103 can support. Each of the 4 switching outputs can be combined with other outputs to create higher power output voltages if needed. The 4 outputs can also be driven independently. When combining two (or more) outputs into a single supply, it is important that the voltage related commands (to turn ON/OFF, to ramp at a certain rate, to set a certain voltage etc.) occur to both outputs at the same time, and this is handled automatically inside PE24103.

The MFR_SPECIFIC_MATRIX command is also compulsory when combining two or more of the switching outputs into a single, combined, output.

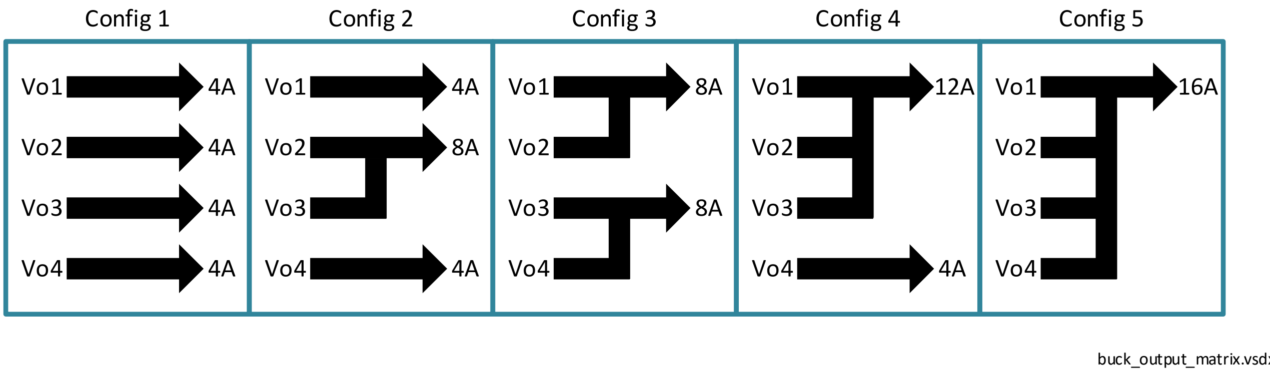


Figure 1 PE24103 Switching Output Configurations

The relationship between the supported PAGE values and the configurations in Figure 1 is fully described in Table 4. Note that PAGE 0 is not used by PE24103. There are two general rules which underpin the choice of the correct PAGE value to use for a particular command in a particular configuration:

- For a combined output, VOUT related commands use the lower numbered output and PAGE value for control inputs or to readback values.
- For a combined output, IOUT related commands use the unique PAGE value which is associated with each output of PE24103.

Table 4 PE24103 PAGE Number vs Configuration & Switching Output

Configuration	PAGE Number	Switching Output Number	Comment
1	1	1	Four independent outputs each accessed by a unique PAGE value for both read and write.
	2	2	
	3	3	
	4	4	
2	1	1	For configuration 2 use PAGE number 2 to control switching outputs 2 and 3 at the same time with the same voltage values. Switching outputs 1 and 4 are accessed using PAGEs 1 and 4.
	2	2 & 3	
	4	4	
3	1	1 & 2	For configuration 3 use PAGE number 1 to control switching outputs 1 and 2 at the same time with the same voltage values. Use PAGE number 3 to control switching outputs 3 and 4 at the same time with the same voltage values.
	3	3 & 4	
4	1	1, 2 & 3	For configuration 4 use PAGE number 1 to control switching outputs 1, 2 and 3 at the same time with the same voltage values. Switching output 4 is accessed using PAGE 4.
	4	4	

Table 4 PE24103 PAGE Number vs Configuration & Switching Output

Configuration	PAGE Number	Switching Output Number	Comment
5	1 or 255	1, 2, 3 & 4	For configuration 5 use PAGE number 1 or the pre-defined broadcast PAGE value of 255 to control all the switching outputs at the same time with the same voltage values.

The PAGE structure described in Table 4 is the simplest scheme in terms of knowing which PAGE value to use to control which VOUTx output. The main drawback of this simple scheme becomes clear when considering the reporting, and setting, of IOUT related currents.

Each individual output of PE24103 is designed to support a nominal load current of up to 4A. In configuration 2, combining VOUT3 with VOUT2 allows a shared load current of up to 8A to be supported. When considering combined outputs, it is important to understand that the IOUT related reporting and fault limit setting remains based on a single channel normal load of 4A.

Considering configuration 2 (where VOUT3 is combined with VOUT2), all output voltage related controls will come from PAGE2. To read IOUT, the READ_IOUT command should be used with both PAGE2 & PAGE3 and the values summed externally. Setting IOUT overcurrent uses the IOUT_OC_FAULT_LIMIT value for PAGE2 and PAGE3 (each scaled for the nominal output range of 0 to 4A for an individual output). Inside PE24103, the IOUT2 and IOUT3 currents are individually measured and each is compared with its own unique current limit and readback value.

The PAGE values vs configuration and (PAGED) commands are summarized in Table 5. Note that the shaded cells in Table 5 indicate those current based commands which continue to use all four of the normal PAGE values regardless of the configuration.

Table 5 PE24103 Configurations, PAGES & Commands

PAGED CMD	Config					Comment
	1	2	3	4	5	
OPERATION	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
CLEAR_FAULTS	1/2/3/4	1/2/4	1/3	1/4	1 or 255	PAGE value to use depends on whether clearing VOUT or IOUT faults (or all).
VOUT_MODE	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
VOUT_COMMAND	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
VOUT_MARGIN_HIGH	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
VOUT_MARGIN_LOW	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
VOUT_OV_FAULT_LIMIT	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
VOUT_UV_FAULT_LIMIT	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
IOUT_OC_FAULT_LIMIT	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	
TON_DELAY	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
TON_RISE	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
TOFF_DELAY	1/2/3/4	1/2/4	1/3	1/4	1 or 255	

Table 5 PE24103 Configurations, PAGES & Commands

PAGED CMD	Config					Comment
	1	2	3	4	5	
TOFF_FALL	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
STATUS_BYTE	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
STATUS_WORD	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
STATUS_VOUT	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
READ_VOUT	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
READ_IOUT	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	
SMBALERT_MASK	1/2/3/4	1/2/4	1/3	1/4	1 or 255	PAGE value to use depends on whether setting/reading VOUT or IOUT status register MASKs.
VOUT_TRANSITION_RATE	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
IOUT_UC_FAULT_LIMIT	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	
POWER_GOOD_ON	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
POWER_GOOD_OFF	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
STATUS_IOUT	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	
MFR_SPECIFIC_OP_BEH	1/2/3/4	1/2/4	1/3	1/4	1 or 255	
MFR_IOUT_CAL_GAIN	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	1/2/3/4	

The value set by the PAGE command is not stored (or restored) from NVM by the STORE_USER_ALL (or RESTORE_USER_ALL) command.

The PAGE command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

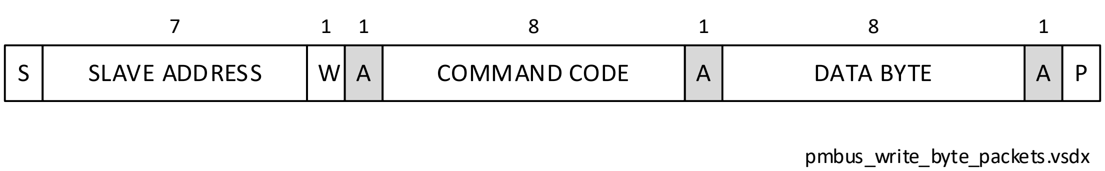


Figure 2 PE24103 Write BYTE Command Example

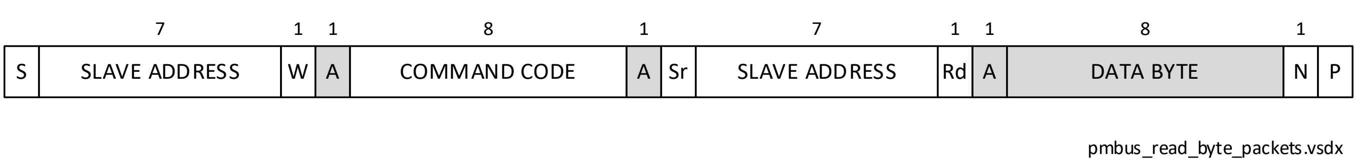


Figure 3 PE24103 Read BYTE Command Example

5.1.2 OPERATION (Command Code 01h)

The OPERATION command is used to configure the operational state of PE24103. The OPERATION command is used to:

- Turn PE24103 outputs on and off with commands sent over the PMBus.
- Select the voltage margin state of PE24103 (margin off, margin high, margin low).
- Select whether fault conditions caused by margining are ignored or acted upon.
- Select whether PE24103 powers down immediately or follows the programmed TOFF_DELAY and TOFF_FALL commands when commanded to turn off an output.

The data byte of the OPERATION command is summarized in Table 6.

Table 6 OPERATION Command Data Byte

Bit Number	Description
7	ON/OFF State
6	Turn Off Behavior
5:4	Voltage Command Source
3:2	Margin Fault Response
1	Transition Control
0	Reserved

Note that PE24103 does not support AVSBus control, so not all bits of the OPERATION command can be set.

5.1.2.1 *OPERATION Command Bit [7]*

Bit [7] controls whether the PE24103 output is on or off.

If bit [7] is cleared (equals 0), then the output is off.

If bit [7] is set (equals 1), then the output is on.

5.1.2.2 *OPERATION Command Bit [6]*

Bit [6] controls the power down behavior.

If bit [7] is set (equals 1), then bit [6] is ignored.

If bit [7] is cleared (equals 0), then:

- If Bit [6] is cleared (equals 0), then the output is turned off immediately and any power down sequencing commands are ignored;
- Else if Bit [6] is set (equals 1), then the device powers down following the values set by the TOFF_DELAY and TOFF_FALL commands.

5.1.2.3 *OPERATION Command Bits [5:4]*

If the PMBus device output is on (bit [7] = 1), then bits [5:4] control the basic source of the output voltage command.

If bits [5:4] equal 2'b00, then the nominal output voltage is set by the PMBus VOUT_COMMAND data.

If bits [5:4] equal 2’b01, then the nominal output voltage is set by the PMBus VOUT_MARGIN_LOW data.

If bits [5:4] equal 2’b10, then the nominal output voltage is set by the PMBus VOUT_MARGIN_HIGH data.

If bits [5:4] equal 11b, then the nominal output voltage is set by the AVSBus (AVS_VOUT_COMMAND). Note that PE24103 does not support AVSBus control, so setting bits 5:4 to 2’b11 is not allowed.

5.1.2.4 OPERATION Command Bits [3:2]

If the PMBus device output is on (bit [7] = 1), then bits [3:2] select whether a fault is generated if a margin command causes the output voltage to go beyond a limit set by the VOUT_OV_FAULT_LIMIT or VOUT_UV_FAULT_LIMIT commands.

In some cases, during system qualification testing for example, it may be desirable to program the output voltages well beyond the normal operating limits. In this case, having PE24103 shut down its output due to a fault condition defeats the purpose of the test. This setting allows a system engineer to prevent the activation of the fault detection circuitry during margin testing.

If bits [3:2] equal 2’b01, then faults caused by selecting VOUT_MARGIN_HIGH or VOUT_MARGIN_LOW as the nominal output voltage source are ignored.

If bits [3:2] equal 2’b10, then faults caused by selecting VOUT_MARGIN_HIGH or VOUT_MARGIN_LOW as the nominal output voltage source are acted upon as normal.

5.1.2.5 OPERATION Command Bit [1]

Bit [1] controls how the nominal output voltage command is updated, or not, when control is passed from the AVSBus to the PMBus. As PE24103 does not support AVSBus control, this bit is set to 1’b0.

5.1.2.6 OPERATION Command Bit [0]

This bit is reserved for future use.

5.1.2.7 OPERATION Command Invalid Data

If PE24103 receives an OPERATION command data byte that attempts to configure or operate the device in an unsupported manner then the device shall treat this as invalid data and declare a communications fault through the STATUS_CML register.

Examples include, but are not limited to, attempting to enable the AVSBus when the AVSBus is not supported, attempting to margin high or low when margining is not supported, or attempting to set bits [3:2] to 2’b00 or 2’b11.

5.1.2.8 OPERATION Command and PAGES

The OPERATION command is a PAGED command so that the four outputs supported by PE24103 may be individually turned on or off independently of the other outputs. The contents of the OPERATION command data byte are also stored in NVM using the STORE_USER_ALL command and can be restored using the RESTORE_USER_ALL command.

The OPERATION command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.3 ON_OFF_CONFIG (Command Code 02h)

The ON_OFF_CONFIG command has a single data byte associated with it. PE24103 does not support a CONTROL pin, so as a result, not all bits of the ON_OFF_CONFIG register value can be set by the user. The 8-bits of data which are part of the ON_OFF_CONFIG are defined for PE24103 in Table 7.

Table 7 PE24103 ON_OFF_CONFIG Command Data Byte

Bits	Purpose	PE24103 Bit Value	Meaning
7:5		3'b000	Bit reserved for future use. Writing to these bits will have no effect. Reading these bits will always return 3'b000.
4	Sets the default to either operate any time power is present or for the on/off to be controlled by serial bus commands	User set. When 1'b0	PE24103 powers up any time power is present and bits 3:0 of this register have no effect.
		User set. When 1'b1	PE24103 does not power up until commanded to do by the OPERATION command (and bits 3:0 of this register).
3	Controls how PE24103 responds to commands received on the serial bus. This bit ignored unless ON_OFF_CONFIG[4] is set.	Illegal (1'b0)	PE24103 ignores the on/off portion of the OPERATION command.
		1'b1 - fixed	For PE24103 to start, the on/off portion of the OPERATION command must instruct the device to run.
2	Controls how PE24103 responds to the CONTROL pin	1'b0 - fixed	PE24103 does not support a CONTROL pin.
1	Polarity of the CONTROL pin	1'b0 - fixed	PE24103 does not support a CONTROL pin.
0	How to respond to the CONTROL pin when turning PE24103 off.	1'b0 - fixed	PE24103 does not support a CONTROL pin.

As noted in Table 7, the ON_OFF_CONFIG command for PE24103 only has two active bits and so the legal values for the ON_OFF_CONFIG data byte are:

8'b000_0_1_000 PE24103 powers up automatically when power is present

8'b000_1_1_000 PE24103 waits for the on/off portion of the OPERATION command to turn on

The value set by the ON_OFF_CONFIG command is not PAGED (just one value for the whole device. The value is stored in NVM by the STORE_USER_ALL command and restored from NVM by the RESTORE_USER_ALL command.

The ON_OFF_CONFIG command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.4 CLEAR_FAULTS (Command Code 03h)

The CLEAR_FAULTS command is used to clear any fault bits that have been set for the active PAGE value. This command clears all bits in all status registers for the selected PAGE simultaneously. To clear all faults on all pages, the broadcast PAGE value of 0xFF can be used. At the same time, PE24103 negates (clears, releases) its SMBALERT signal output when the CLEAR_FAULTS command is received.

The CLEAR_FAULTS command does not cause an output that has latched off for a fault condition to turn back on. Outputs that have shut down for a fault condition are restarted only by OPERATION commands which first turn the output OFF and then ON again. A reset of the digital block will also clear all faults.

If a fault is still present when the associated fault bit is cleared, the fault bit shall immediately be set again and the host notified by the usual means.

This command is write only. There is no data byte for this command.

The CLEAR_FAULTS command is not stored (or restored) from NVM by the STORE_USER_ALL (or RESTORE_USER_ALL) command.

The CLEAR_FAULTS command has no data bytes as argument and uses the Send BYTE protocol as shown in Figure 4.

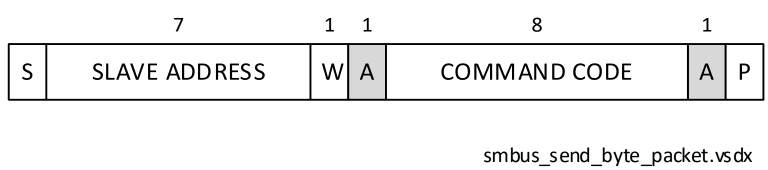


Figure 4 PE24103 Send BYTE Command Example

5.1.5 PAGE_PLUS_WRITE (command Code 05h)

The PAGE_PLUS_WRITE command is used to set the page within PE24103, send a command, and send the data for the command in one packet.

The PAGE_PLUS_WRITE command uses the WRITE BLOCK protocol.

An example of the PAGE_PLUS command being used to send a command that has two data bytes to be written is shown in Figure 5.

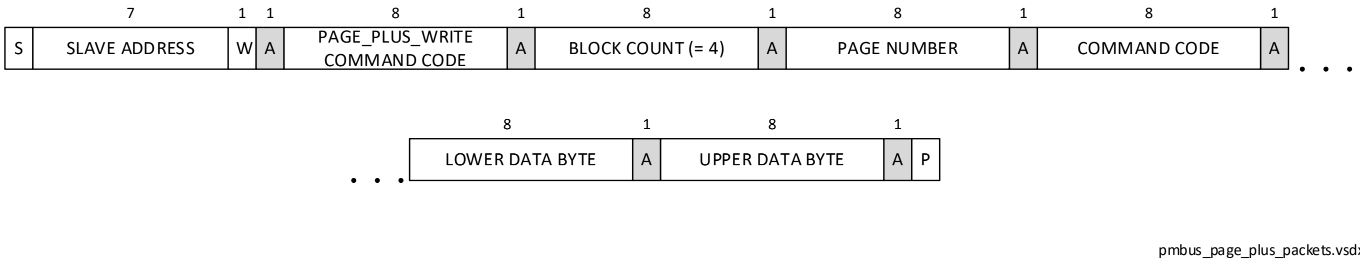


Figure 5 PE24103 PAGE_PLUS_WRITE Command Example

Note that the PAGE_PLUS_WRITE (and PAGE_PLUS_READ) commands are equivalent to a PAGE command followed by another command. The PAGE value stored on-chip is permanently changed by the PAGE_PLUS_WRITE/READ command.

It is possible in future revisions of the PMBus standard that an alternate implementation will be required where only a PAGE command permanently changes the PAGE value. A PAGE_PLUS_xxx command would only alter the PAGE value temporarily (for the duration of that command packet).

In versions of PE24103 after ES2, there is an MTP bit setting to allow support of the future implementation of the PAGE_PLUS_xxx commands.

5.1.6 PAGE_PLUS_READ (Command Code 06h)

The PAGE_PLUS_READ command is used to set the page within PE24103, send a command, and read the data returned by the command in one packet.

The PAGE_PLUS_READ command uses the BLOCK WRITE – BLOCK READ PROCESS CALL protocol.

An example of the PAGE_PLUS command being used to send a command that has two data bytes to be read is shown in Figure 6.

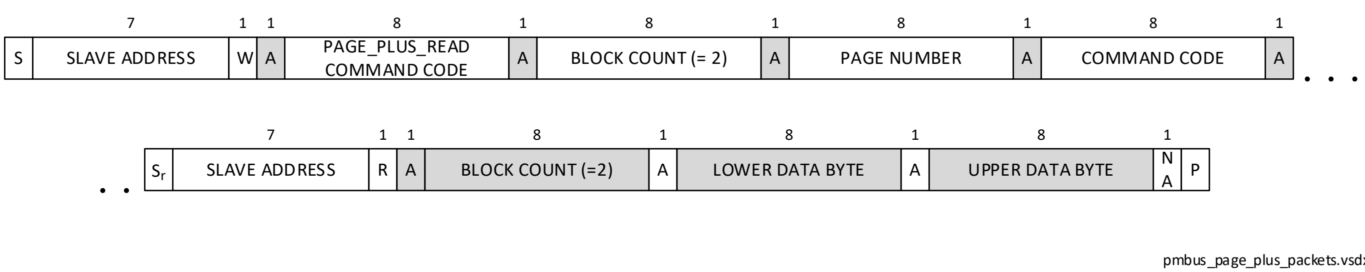


Figure 6 PE24103 PAGE_PLUS_READ Command Example

Note that the PAGE_PLUS_READ (and PAGE_PLUS_WRITE) commands are equivalent to a PAGE command followed by another command. The PAGE value stored on-chip is permanently changed by the PAGE_PLUS_READ/WRITE command.

It is possible in future revisions of the PMBus standard that an alternate implementation will be required where only a PAGE command permanently changes the PAGE value. A PAGE_PLUS_xxx command would only alter the PAGE value temporarily (for the duration of that command).

In versions of PE24103 after ES2, there is an MTP bit setting to allow support of the future implementation of the PAGE_PLUS_xxx commands.

5.1.7 STORE_USER_ALL (Command Code 15h)

The STORE_USER_ALL command instructs PE24103 to copy the entire contents of the Operating Memory to the matching locations in the on-chip non-volatile memory (NVM). Any items in Operating Memory that do not have matching locations in the NVM are ignored.

It is permitted to use the STORE_USER_ALL command while PE24103 is operating, however, due to the time taken to erase, and then program, the non-volatile memory, the processing of telemetry values (ADC conversion results) may be held up while the STORE_USER_ALL command is being processed. It is also not recommended that any of the values in the Operating memory are updated by PMBus command while the STORE_USER_ALL command is being processed. In general it is recommended that the PE24103 outputs be turned off before sending the STORE_USER_ALL command.

This command has no data bytes.

This command is write only.

The STORE_USER_ALL command has no data bytes as argument and uses the Send BYTE protocol as shown in Figure 4.

Table 8 shows a summary of the PMBus commands which have data stored (or restored) by the (RE)STORE_USER_ALL command for the command set that PE24103 supports. There are approximately 173 bytes of non-volatile memory used/listed in the table.

Table 8 PE24103 PMBus Commands & STORE_USER_ALL Data Amounts

Command Code	Command Name	# of Data Bytes per command	Does PE24103 Use PAGING With This Command?	How Many Bytes of NVM are Written To Support This Command By STORE_USER_ALL
00h	PAGE	1	No	0
01h	OPERATION	1	Yes	4
02h	ON_OFF_CONFIG	1	No	1
03h	CLEAR_FAULTS	0	Yes	0
05h	PAGE_PLUS_WRITE	Variable	N/A	0
06h	PAGE_PLUS_READ	Variable	N/A	0
15h	STORE_USER_ALL	0	No	0
16h	RESTORE_USER_ALL	0	No	0
19h	CAPABILITY	1	No	0
1Bh	SMBALERT_MASK	2	Variable	0
20h	VOUT_MODE	1	Yes	8

Table 8 PE24103 PMBus Commands & STORE_USER_ALL Data Amounts

Command Code	Command Name	# of Data Bytes per command	Does PE24103 Use PAGING With This Command?	How Many Bytes of NVM are Written To Support This Command By STORE_USER_ALL
21h	VOUT_COMMAND	2	Yes	8
25h	VOUT_MARGIN_HIGH	2	Yes	8
26h	VOUT_MARGIN_LOW	2	Yes	8
27h	VOUT_TRANSITION_RATE	2	Yes	8
33h	FREQUENCY_SWITCH	2	No	2
35h	VIN_ON	2	No	2
36h	VIN_OFF	2	No	2
40h	VOUT_OV_FAULT_LIMIT	2	Yes	8
44h	VOUT_UV_FAULT_LIMIT	2	Yes	8
46h	IOUT_OC_FAULT_LIMIT	2	Yes	8
4Bh	IOUT_UC_FAULT_LIMIT	2	Yes	8
4Fh	OT_FAULT_LIMIT	2	No	2
55h	VIN_OV_FAULT_LIMIT	2	No	2
59h	VIN_UV_FAULT_LIMIT	2	No	2
5Bh	IIN_OC_FAULT_LIMIT	2	No	2
5Eh	POWER_GOOD_ON	2	Yes	8
5Fh	POWER_GOOD_OFF	2	Yes	8
60h	TON_DELAY	2	Yes	8
61h	TON_RISE	2	Yes	8
64h	TOFF_DELAY	2	Yes	8
65h	TOFF_FALL	2	Yes	8
78h	STATUS_BYTE	1	Yes	0
79h	STATUS_WORD	2	Yes	2 (MASK register)
7Ah	STATUS_VOUT	1	Yes	4 (MASK registers)
7Bh	STATUS_IOUT	1	Yes	4 (MASK registers)
7Ch	STATUS_INPUT	1	No	1 (MASK register)
7Dh	STATUS_TEMPERATURE	1	No	1 (MASK register)
7Eh	STATUS_CML	1	No	1 (MASK register)
7Fh	STATUS_OTHER	1	No	1 (MASK register)
80h	STATUS_MFR_SPECIFIC	1	No	1 (MASK register)
88h	READ_VIN	2	No	0
89h	READ_IIN	2	No	0
8Bh	READ_VOUT	2	Yes	0
8Ch	READ_IOUT	2	Yes	0
8Dh	READ_TEMPERATURE_1	2	No	0
98h	PMBUS_REVISION	1	No	0
99h	MFR_ID	1	No	0
A0h	MFR_VIN_MIN	2	No	0
A1h	MFR_VIN_MAX	2	No	0
A2h	MFR_IIN_MAX	2	No	0
A4h	MFR_VOUT_MIN	2	No	0

Table 8 PE24103 PMBus Commands & STORE_USER_ALL Data Amounts

Command Code	Command Name	# of Data Bytes per command	Does PE24103 Use PAGING With This Command?	How Many Bytes of NVM are Written To Support This Command By STORE_USER_ALL
A5h	MFR_VOUT_MAX	2	No	0
A6h	MFR_IOUT_MAX	2	No	0
ADh	IC_DEVICE_ID	1	No	0
C4h	MFR_SPECIFIC_PGOOD	2	No	2
C5h	MFR_SPECIFIC_MATRIX	2	No	2
C6h	MFR_SPECIFIC_IOUTFILT	2	Yes	8
C7h	MFR_SPECIFIC_OP_BEH	1	Yes	4
C8h	MFR_SPECIFIC_MASK	2	No	2
C9h	MFR_SPECIFIC_DCR_TC	2	No	2
CAh	MFR_SPECIFIC_RSENSE_TC	2	No	2
CBh	MFR_IOUT_CAL_GAIN	2	Yes	8
CCh	MFR_IIN_CAL_GAIN	2	No	2
CDh	MFR_CFG_PMBUS	1	No	1

5.1.8 RESTORE_USER_ALL (Command Code 16h)

The RESTORE_USER_ALL command instructs PE24103 to copy the entire contents of the non-volatile memory (NVM) to the matching locations in the Operating Memory. The values in the Operating Memory are overwritten by the value retrieved from NVM. Any items in NVM that do not have matching locations in the Operating Memory are ignored.

It is permitted to use the RESTORE_USER_ALL command while the device is operating, but this is strongly discouraged. The data from the NVM will be restored to the operating memory serially, and so the interim configuration of the device while data is being copied is not controlled and data dependent. Unplanned configurations could be applied momentarily. The safest option for PE24103 is to disable all outputs before using the RESTORE_USER_ALL command. The data for the OPERATION command will be the last data to be restored from NVM to operating memory and will therefore (if so defined) re-enable outputs once copied.

This command has no data bytes.

This command is write only.

The RESTORE_USER_ALL command has no data bytes as argument and uses the Send BYTE protocol as shown in Figure 4.

5.1.9 CAPABILITY (Command Code 19h)

The CAPABILITY command is read only and supplies a single byte of data. The 8-bits of data which are supplied in response to the CAPABILITY command for PE24103 are defined in Table 9.

Table 9 PE24103 CAPABILITY Command Response Byte

Bits	Description	PE24103 Value	Meaning
7	Packet Error Checking	1'b0	Packet Error Checking is not supported
6:5	Maximum Bus Speed	2'b10	PE24103 supports a PMBus speed of up to 1MHz.
4	SMBAlert	1'b1	PE24103 does support an SMBAlert pin and the SMBus Alert response protocol.
3	Numeric Format	1'b0	PE24103 supports numeric data using LINEAR11, ULINEAR16 and SLINEAR16 data formats rather than IEEE half precision floating point.
2	AVSBus support	1'b0	PE24103 does not support AVSBus.
1:0	Reserved	2'b00	Reserved bits.

The value returned by the CAPABILITY command is not affected by the PAGE number and cannot be changed by the user so is not affected by the STORE_USER_ALL or RESTORE_USER_ALL commands.

The CAPABILITY command has a single byte as argument and uses the Read BYTE protocol as shown in Figure 3.

For versions of PE24103 after ES2, attempting to write using the (read only) CAPABILITY command will trigger a fault bit in the STATUS_CML register.

5.1.10 SMBALERT_MASK (Command Code 1Bh)

The SMBALERT_MASK command may be used to prevent a fault condition from asserting the SMBALERT pin on PE24103.

The command format used to block a status bit (or bits) from triggering SMBALERT is shown in Figure 7. The bits in the mask byte align with the bits in the corresponding status register. For example if the STATUS_TEMPERATURE command code were sent with the mask byte 0x80, then PE24103 will not assert SMBALERT in the event of an over-temperature fault. The underlying status register and bit are unchanged by the mask value.

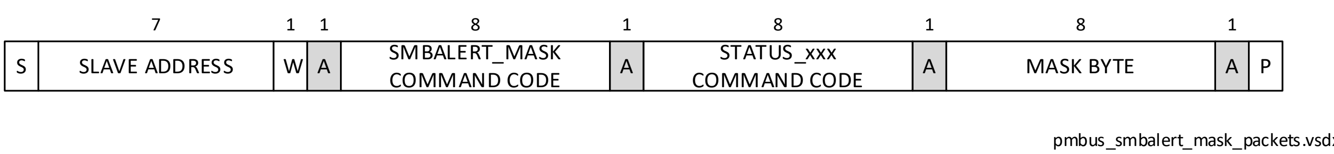


Figure 7 PE24103 SMBALERT_MASK Write Sequence

The command format used by to read back the current SMBALERT_MASK value for a particular status register is shown in Figure 8.

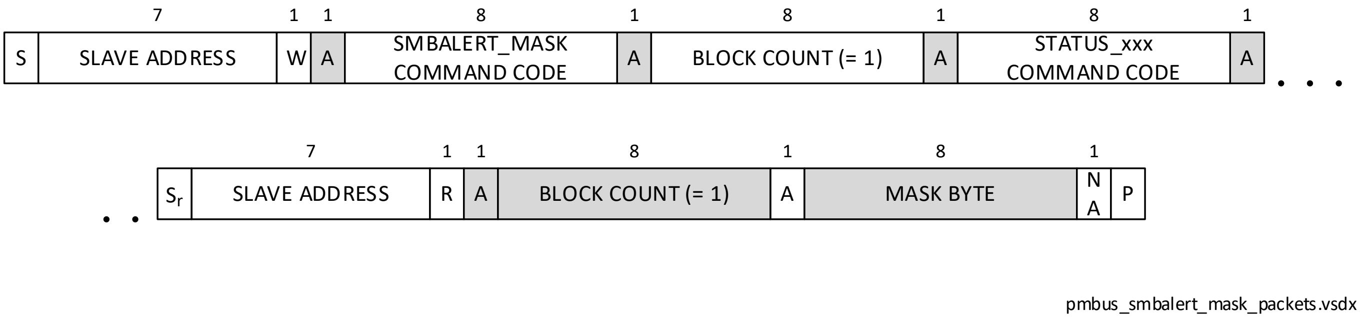


Figure 8 PE24103 SMBALERT_MASK Read Sequence

The STATUS_VOUT and STATUS_IOUT commands are PAGED. As a result when using the SMBALERT_MASK command with the STATUS_VOUT or STATUS_IOUT command codes, the active PAGE number is also taken into account. There is a mask register for each PAGED version of the STATUS_VOUT (and STATUS_IOUT) register. Using the broadcast PAGE value (0xFF) prior to a SMBALERT_MASK write using the STATUS_VOUT command code would block all the versions of STATUS_VOUT from triggering the SMBALERT pin.

Other status registers (STATUS_TEMPERATURE for example) are not PAGED and so the current PAGE value is not important when using the SMBALERT_MASK code with the STATUS_TEMPERATURE etc. command code.

5.1.11 VOUT_MODE (Command Code 20h)

The VOUT_MODE command is read-write, although for PE24103 only bit[7] is writable by the user. This command is PAGED (although bits 6:0 are fixed and are therefore the same for all PAGES). VOUT_MODE controls/defines how output voltage related commands should be interpreted.

The 8-bits of data which are supplied in response to the VOUT_MODE command are defined for PE24103 in Table 10.

Table 10 PE24103 VOUT_MODE Command Response Byte

Bits	Description	PE24103 Value	Meaning
7	Selects absolute (1’b0) or relative (1’b1) mode	User set	For 6 VOUT related commands this bit determines how the data value sent for those commands should be treated.
6:5	Data format	2’b00	PE24103 Voltage related commands using a ULINEAR16 numeric data format with an exponent value set by bits 4:0.

Table 10 PE24103 VOUT_MODE Command Response Byte

Bits	Description	PE24103 Value	Meaning
4:0	Exponent	-11 (5'b1_0101)	Using an exponent value of -11 means that voltage related commands can support an (unsigned) data range of 0.0V to +31.99V with a resolution of roughly 500uV per bit.

The values written using the VOUT_MODE command are PAGED. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The relative/absolute bit (VOUT_MODE[7]) affects six commands which are supported for PE24103:

- VOUT_MARGIN_HIGH
- VOUT_MARGIN_LOW
- VOUT_OV_FAULT_LIMIT
- VOUT_UV_FAULT_LIMIT
- POWER_GOOD_ON
- POWER_GOOD_OFF

Note that PE24103 does support a unique relative/absolute mode setting for each of the four outputs supported by the device. This may be in conflict with the PMBus standard (section 8.5 of Part II of the Rev 1.3.1 PMBus standard says "*It is not permitted to mix Absolute and Relative value mode within the same PMBus device.*")

Care should be taken if VOUT_MODE[7] is changed, care should also be taken when using combined PAGES (for example the broadcast PAGE number of 255 especially). Consider a broadcast write to change the VOUT_OV_FAULT_LIMIT value for all 4 outputs. If the VOUT_MODE[7] values for each of the outputs were not the same, then the same VOUT_OV_FAULT_LIMIT value could be treated as a relative value for some outputs and an absolute value for other outputs. This is probably undesirable.

The values used for VOUT_MARGIN_HIGH, VOUT_MARGIN_LOW, VOUT_OV_FAULT_LIMIT, VOUT_UV_FAULT_LIMIT, POWER_GOOD_ON and POWER_GOOD_OFF will be recalculated (in relative mode) when the individual command value is changed. All 6 of the values will be recalculated (in relative mode) when the VOUT_COMMAND value is written. The values will NOT be recalculated when VOUT_MODE[7] changes.

The VOUT_MODE command is a PAGED command with a unique value for each switching output. The contents of the VOUT_MODE command data byte are stored in NVM using the STORE_USER_ALL command and can be restored using the RESTORE_USER_ALL command.

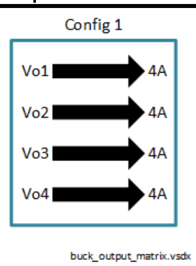
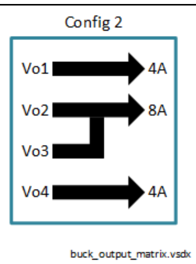
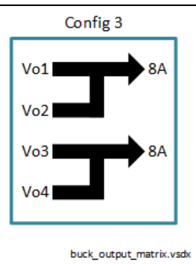
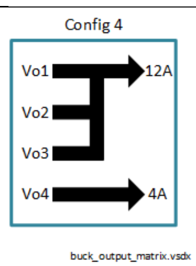
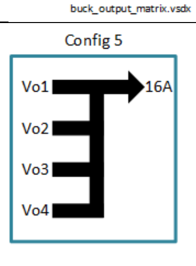
The VOUT_MODE command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.12 VOUT_COMMAND (Command Code 21h)

The VOUT_COMMAND command is read/write and sets the value of the output voltage, in volts, for one, or more, of the PE24103 switching outputs. This command is PAGED.

The two data bytes for VOUT_COMMAND use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

Table 11 PE24103 VOUT_COMMAND Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use VOUT_COMMAND with PAGES 1 to 4 to set (or read back) the commanded VOUT values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use VOUT_COMMAND for PAGES 1 & 4 to set (or read back) the commanded VOUT values for outputs 1 and 4 (respectively). Use VOUT_COMMAND for PAGE 2 to set (or read back) the commanded VOUT for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use VOUT_COMMAND for PAGE 1 to set (or read back) the commanded VOUT for shared outputs 1 and 2. Use VOUT_COMMAND for PAGE 3 to set (or read back) the commanded VOUT for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use VOUT_COMMAND for PAGE 1 to set (or read back) the commanded VOUT for shared outputs 1, 2 and 3. Use VOUT_COMMAND for PAGE 4 to set (or read back) the commanded VOUT for output 4.
5		All outputs are combined.	Use VOUT_COMMAND for PAGE 1 to set (or read back) the commanded VOUT for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

The values written using the VOUT_COMMAND are PAGED as noted in Table 11. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The VOUT_COMMAND command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

	7	1	1	8	1	8	1	8	1	
S	SLAVE ADDRESS	W	A	COMMAND CODE	A	LOWER DATA BYTE	A	UPPER DATA BYTE	A	P

pmbus_write_word_packets.vsdx

Figure 9 PE24103 Write WORD Command Example

	7	1	1	8	1
S	SLAVE ADDRESS	W	A	COMMAND CODE	A

. . .

	7	1	1	8	1	8	1	
Sr	SLAVE ADDRESS	Rd	A	LOWER DATA BYTE	A	UPPER DATA BYTE	N	P

pmbus_read_word_packets.vsdx

Figure 10 PE24103 Read WORD Command Example

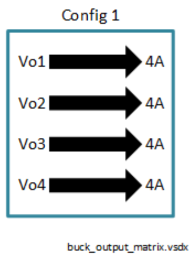
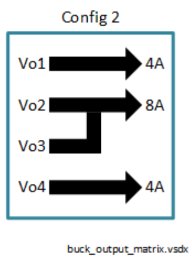
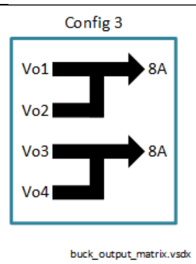
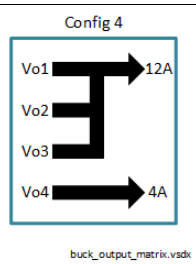
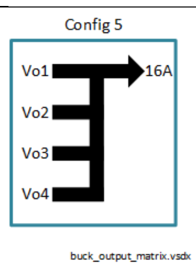
5.1.13 VOUT_MARGIN_HIGH (Command Code 25h)

The VOUT_MARGIN_HIGH command is read/write and sets the margin high value of the output voltage for one, or more, of the PE24103 switching outputs. This command is PAGED. The VOUT_MARGIN_HIGH value may be used to set the target output voltage for a PE24103 switching output depending on bits in the OPERATION command.

The two data bytes for VOUT_MARGIN_HIGH use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. PE24103 internally stores the VOUT_MARGIN_HIGH value using a 10-bit data format corresponding to bits 13:4 of the written value. When reading back a VOUT_MARGIN_HIGH value bits 15:14 and 3:0 will always read as zeros. The 10-bit internal data format allows the unsigned VOUT_MARGIN_HIGH value to represent voltages in the range of 0.0V to +7.99V with a resolution of approximately 8mV when in absolute mode. Alternatively when in relative mode (see the VOUT_MODE command) the value for VOUT_MARGIN_HIGH will be treated as a scaling factor relative to the value for the VOUT_COMMAND. The scaling factor is in the range of 0.0 to 7.99 with an LSB of approximately 0.8%.

Note: section 5.4.2 of Part II of the PMBus spec (revision 1.3.1) specifically notes that the received value may be stored with fewer bits than the data format allows, and that reading a value which does not completely match the input value due to bits which are not stored is not considered an error.

Table 12 PE24103 VOUT_MARGIN_HIGH Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use VOUT_MARGIN_HIGH with PAGES 1 to 4 to set (or read back) the commanded VOUT margin high values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use VOUT_MARGIN_HIGH for PAGES 1 & 4 to set (or read back) the commanded VOUT margin high value for outputs 1 & 4 (respectively). Use VOUT_MARGIN_HIGH for PAGE 2 to set (or read back) the commanded VOUT margin high value for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use VOUT_MARGIN_HIGH for PAGE 1 to set (or read back) the commanded VOUT margin high value for shared outputs 1 and 2. Use VOUT_MARGIN_HIGH for PAGE 3 to set (or read back) the commanded VOUT margin high value for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use VOUT_MARGIN_HIGH for PAGE 1 to set (or read back) the commanded VOUT margin high value for shared outputs 1, 2 and 3. Use VOUT_MARGIN_HIGH for PAGE 4 to set (or read back) the commanded VOUT margin high value for output 4.
5		All outputs are combined.	Use VOUT_MARGIN_HIGH for PAGE 1 to set (or read back) the commanded VOUT margin high value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

The values written using the VOUT_MARGIN_HIGH are PAGED as noted in Table 12. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The VOUT_MARGIN_HIGH command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.14 VOUT_MARGIN_LOW (Command Code 26h)

The VOUT_MARGIN_LOW command is read/write and sets the margin low value of the output voltage for one, or more, of the PE24103 switching outputs. This command is PAGED. The VOUT_MARGIN_LOW value may be used to set the target output voltage for a PE24103 switching output depending on bits in the OPERATION command.

The two data bytes for VOUT_MARGIN_LOW use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. PE24103 internally stores the VOUT_MARGIN_LOW value using a 10-bit data format corresponding to bits 13:4 of the written value. When reading back a VOUT_MARGIN_LOW value bits 15:14 and 3:0 will always read as zeros. The 10-bit internal data format allows the unsigned VOUT_MARGIN_LOW value to represent voltages in the range of 0.0V to +7.99V with a resolution of approximately 8mV when in absolute mode. Alternatively when in relative mode (see the VOUT_MODE command) the value for VOUT_MARGIN_LOW will be treated as a scaling factor relative to the value for the VOUT_COMMAND. The scaling factor is in the range of 0.0 to 7.99 with an LSB of approximately 0.8%.

Note: section 5.4.2 of Part II of the PMBus spec (revision 1.3.1) specifically notes that the received value may be stored with fewer bits than the data format allows, and that reading a value which does not completely match the input value due to bits which are not stored is not considered an error.

Table 13 PE24103 VOUT_MARGIN_LOW Command vs Configuration and PAGE Number

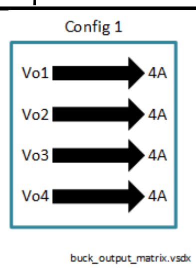
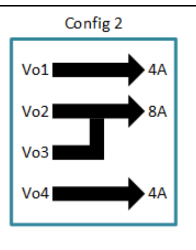
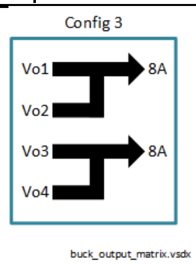
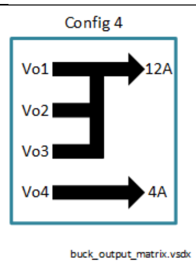
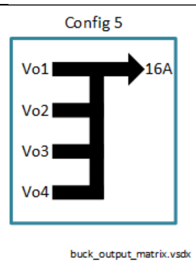
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use VOUT_MARGIN_LOW with PAGES 1 to 4 to set (or read back) the commanded VOUT margin low values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use VOUT_MARGIN_LOW for PAGES 1 & 4 to set (or read back) the commanded VOUT margin low value for outputs 1 & 4 (respectively). Use VOUT_MARGIN_LOW for PAGE 2 to set (or read back) the commanded VOUT margin low value for shared outputs 2 and 3.

Table 13 PE24103 VOUT_MARGIN_LOW Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use VOUT_MARGIN_LOW for PAGE 1 to set (or read back) the commanded VOUT margin low value for shared outputs 1 and 2. Use VOUT_MARGIN_LOW for PAGE 3 to set (or read back) the commanded VOUT margin low value for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use VOUT_MARGIN_LOW for PAGE 1 to set (or read back) the commanded VOUT margin low value for shared outputs 1, 2 and 3. Use VOUT_MARGIN_LOW for PAGE 4 to set (or read back) the commanded VOUT margin low value for output 4.
5		All outputs are combined.	Use VOUT_MARGIN_LOW for PAGE 1 to set (or read back) the commanded VOUT margin low value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

The values written using the VOUT_MARGIN_LOW are PAGED as noted in Table 13. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The VOUT_MARGIN_LOW command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.15 VOUT_TRANSITION_RATE (Command Code 27h)

When one of the switching outputs of PE24103 receives either a VOUT_COMMAND or OPERATION (Margin High, Margin Low, Margin off) command that causes the target output voltage to change the VOUT_TRANSITION_RATE command sets the rate in mV/us at which the output should attempt to change voltage. The VOUT_TRANSITION_RATE does not apply when an output is commanded to turn on or turn off.

The two data bytes for the VOUT_TRANSITION_RATE command are defined using a LINEAR11 data format as follows:

Exponent = -9 (5'b1_0111)

Mantissa which is expressed as sign_x.y_yyyy_yyyy where x are the integer bits (in units of mV/us) and y are the fractional bits.

The LINEAR11 format specified allows ramp rates from 1.9uV/us (or approx. 2V/second) to 2mV/us (~2V/ms).

A value of full scale (16'b10111_0_1111111_111 = 0xBBFF) means that PE24103 should ramp the output voltage up as fast as it can. A value of 0 for the VOUT_TRANSITION_RATE value will result in the output voltage failing to change at all (and is effectively an error condition as a result).

Note that although the LINEAR11 data format does allow for negative data values, a negative number would have no meaning in the context of the VOUT_TRANSITION_RATE command.

For ES1/ES2 silicon if a negative value is written to the VOUT_TRANSITION_RATE command, the sign bit will be ignored. This may lead to unexpected results as small negative values could be interpreted as large positive values when the sign bit is ignored. The VOUT_TRANSITION_RATE value will always read as positive.

For silicon after ES2, writing a negative value to the VOUT_TRANSITION_RATE command will trigger the bad data bit in the STATUS_CML register and will be treated the same as a value of all 0s. In effect this prevents the VOUT values from ramping between one voltage and another. Reading back a negative VOUT_TRANSITION_RATE value will return a value of all 0s.

Table 14 PE24103 VOUT_TRANSITION_RATE Command vs Configuration and PAGE Number

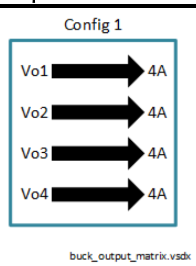
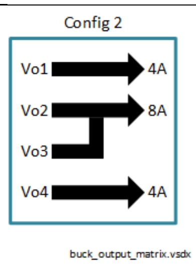
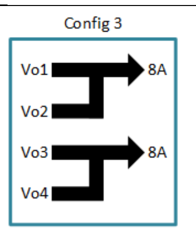
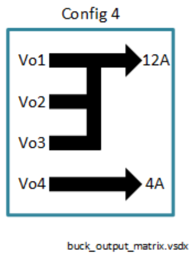
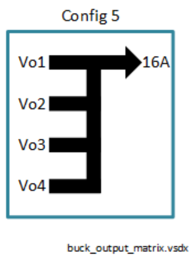
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use VOUT_TRANSITION_RATE with PAGES 1 to 4 to set (or read back) the commanded ramp rate values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use VOUT_TRANSITION_RATE for PAGES 1 & 4 to set (or read back) the commanded ramp rate value for outputs 1 & 4 (respectively). Use VOUT_TRANSITION_RATE for PAGE 2 to set (or read back) the commanded ramp rate value for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use VOUT_TRANSITION_RATE for PAGE 1 to set (or read back) the commanded ramp rate value for shared outputs 1 and 2. Use VOUT_TRANSITION_RATE for PAGE 3 to set (or read back) the commanded ramp rate value for shared outputs 3 and 4.

Table 14 PE24103 VOUT_TRANSITION_RATE Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use VOUT_TRANSITION_RATE for PAGE 1 to set (or read back) the commanded ramp rate value for shared outputs 1, 2 and 3. Use VOUT_TRANSITION_RATE for PAGE 4 to set (or read back) the commanded ramp rate value for output 4.
5		All outputs are combined.	Use VOUT_TRANSITION_RATE for PAGE 1 to set (or read back) the commanded ramp rate value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

The values written using the VOUT_TRANSITION_RATE command are PAGED as noted in Table 14. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The VOUT_TRANSITION_RATE command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.16 FREQUENCY_SWITCH (Command Code 33h)

The PMBus FREQUENCY_SWITCH command allows the switching frequency of PE24103 to be set (in units of kHz) via a 2-byte value which is specified (for PE24103) using the LINEAR11 data format as follows:

Exponent = +1 (5'b0_0001)

Mantissa which is expressed as sign_xx_xxxx_xxxx where x are the integer bits with units of 2kHz/LSB. This allows a range of values from 0Hz to 2046kHz to be expressed.

Note that there are only 7 legal values for the FREQUENCY_SWITCH command. Using a value other than those listed in Table 15 will result in disabling of the switching outputs.

The values are formed by combining the target frequency (in kHz) divided by two with the exponent value for this LINEAR11 value. For example to set a switching frequency of 600kHz, the value to write to FREQUENCY_SWITCH would be:

- Fixed exponent of 5'b0000_1
- Sign bit should always be positive: 1'b0
- Value is 600kHz/2 or 300 written as a 10-bit value: 10'b01_0010_1100
- Combining those values we get: 16'b00001_0_0100101100 or 0x092C

Technically a negative value can be written using the FREQUENCY_SWITCH command, although a negative frequency value does not make sense for PE24103.

For ES1/ES2 silicon if a negative value is written to the FREQUENCY_SWITCH command, the sign bit will be ignored and the magnitude bits will be compared against the 7 legal values expected. The FREQUENCY_SWITCH value will always read as positive.

For silicon after ES2, writing a negative value to the FREQUENCY_SWITCH command will trigger the bad data bit in the STATUS_CML register and will be treated the same as any other illegal value (so the switching outputs will be disabled). Reading back a negative FREQUENCY_SWITCH value will return a value of all 0s.

Table 15 PE24103 Possible Power Output Switching Frequencies

On-Chip Oscillator Frequency	Integer Clock Division Ratio	Resulting Switching Frequency	PE24103 FREQUENCY_SWITCH Value in LINEAR11 data format
12MHz	20	600kHz	16'b00001_0_01_0010_1100 (0x092C)
	12	1.0Mhz	16'b00001_0_01_1111_0100 (0x09F4)
	10	1.2Mhz	16'b00001_0_10_0101_1000 (0x0A58)
	9	1.33MHz	16'b00001_0_10_1001_1011 (0x0A9B)
	8	1.5MHz	16'b00001_0_10_1110_1110 (0x0AEE)
	7	1.71MHz	16'b00001_0_11_0101_1001 (0x0B59)
	6	2.0MHz	16'b00001_0_11_1110_1000 (0x0BE8)

The FREQUENCY_SWITCH command is not PAGED, the same switching frequency value applies to all of the PE24103 power outputs. The value is stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The FREQUENCY_SWITCH command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.17 VIN_ON (Command Code 35h)

The VIN_ON command is read/write and sets the value of the input voltage, in volts, at which PE24103 should start power conversion.

The two data bytes for VIN_ON use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

Although there is nothing hard-coded in the PE24103 hardware, the following relative magnitudes are one way to set VIN related values so that all the possible thresholds play a part in the operation of VIN:

$$MFR\ VIN\ MAX > VIN\ OV > \{VIN\ ON|VIN\ UV\} > VIN\ OFF > MFR\ VIN\ MIN$$

In addition the MFR_VIN_MIN value should be greater than, or equal to, the real hardware limit for VIN_UVLO (at which point PE24103 may no longer respond to PMBus commands at all).

Figure 11 shows a sketch of how the VIN related thresholds may be set in a typical application.

WARNING: Figure 11 includes internal thresholds which the user will not be able to set or control. If the content is useful, a simplified version will be needed for public use.

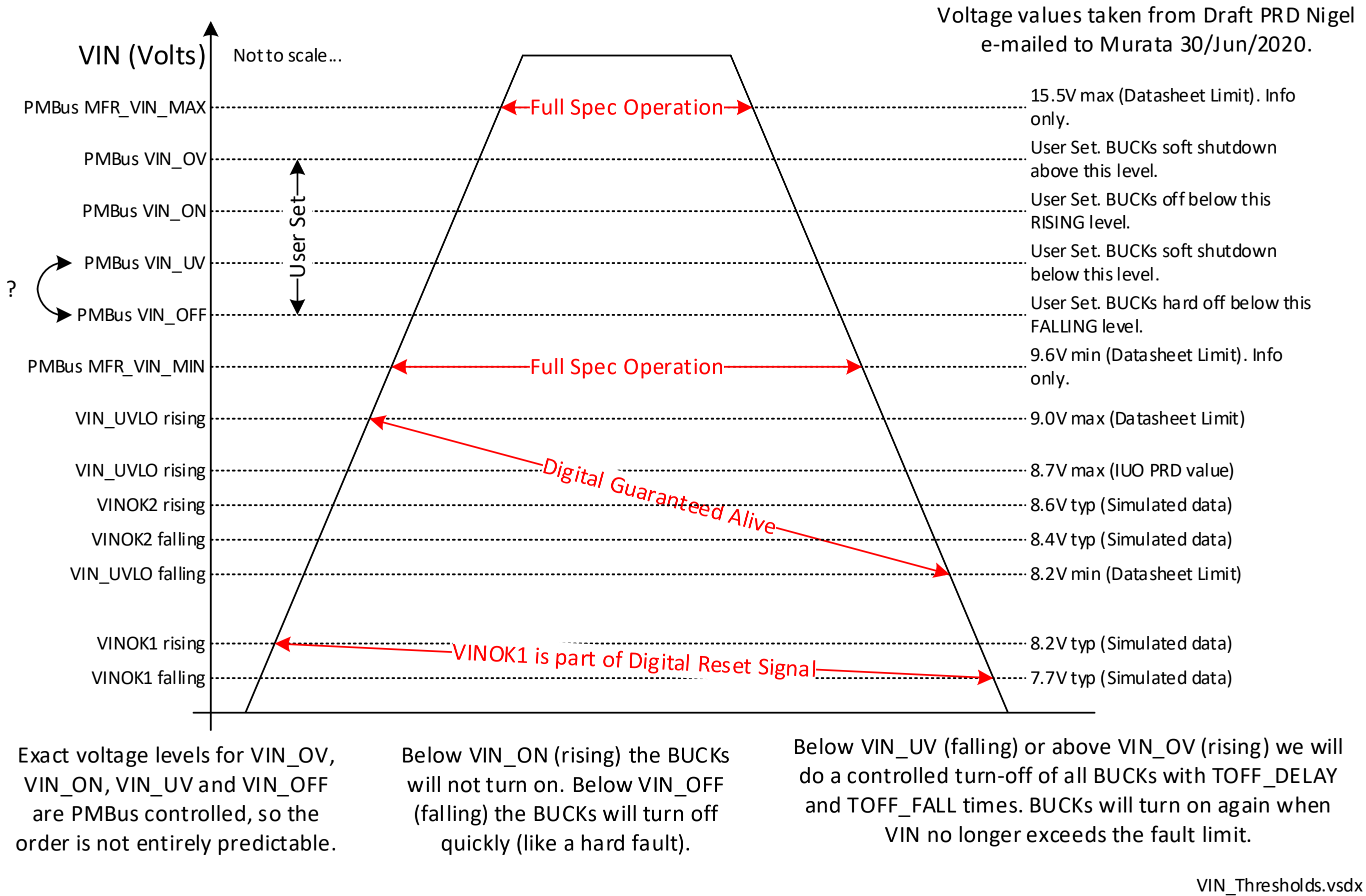


Figure 11 PE24103 VIN Related Thresholds

The VIN_ON value is not affected by the PAGE number. The value is stored in NVM and will be saved by using the STORE_USER_ALL and restored with the RESTORE_USER_ALL command.

The VIN_ON command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.18 VIN_OFF (Command Code 36h)

The VIN_OFF command is read/write and sets the value of the input voltage, in volts, at which PE24103, once full power operation has started, should stop power conversion.

The two data bytes for VIN_OFF use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

Although there is nothing hard-coded in the PE24103 hardware, the following relative magnitudes are one way to set VIN related values so that all the possible thresholds play a part in the operation of VIN:

$$MFR\ VIN\ MAX > VIN\ OV > \{VIN\ ON|VIN\ UV\} > VIN\ OFF > MFR\ VIN\ MIN$$

In addition the MFR_VIN_MIN value should be greater than, or equal to, the real hardware limit for VIN_UVLO (at which point PE24103 may no longer respond to PMBus commands at all).

Figure 11 shows a sketch of how the VIN related thresholds may be set in a typical application.

The VIN_OFF value is not affected by the PAGE number. The value is stored in NVM and will be saved by using the STORE_USER_ALL and restored with the RESTORE_USER_ALL command.

The VIN_OFF command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

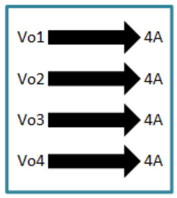
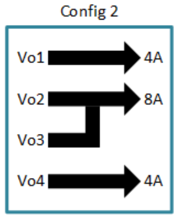
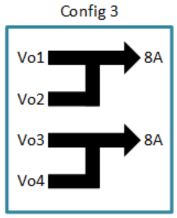
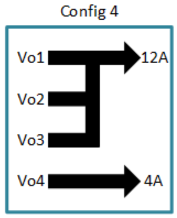
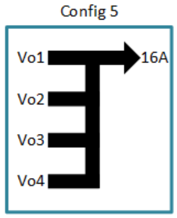
5.1.19 VOUT_OV_FAULT_LIMIT (Command Code 40h)

The VOUT_OV_FAULT_LIMIT command is read/write and sets the value of the output voltage that causes an output overvoltage fault.

The two data bytes for the VOUT_OV_FAULT_LIMIT use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit when in absolute mode. Alternatively when in relative mode (see the VOUT_MODE command) the value for VOUT_OV_FAULT_LIMIT will be treated as a scaling factor relative to the value for the VOUT_COMMAND. The scaling factor is in the range of 0.0 to 31.99 with an LSB of approximately 0.05%.

The VOUT_OV_FAULT_LIMIT command is PAGED so that each output (or combination of outputs) can be controlled individually. Table 16 has more details on how the VOUT_OV_FAULT_LIMIT is applied for the various configurations that PE24103 can support.

Table 16 PE24103 VOUT_OV_FAULT_LIMIT Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
1	Config 1  buck_output_matrix.vsdx	Four independent outputs.	Use VOUT_OV_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded VOUT fault limit voltage value for outputs 1 to 4 (respectively).
2	Config 2  buck_output_matrix.vsdx	Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use VOUT_OV_FAULT_LIMIT for PAGES 1 & 4 to set (or read back) the commanded VOUT fault limit voltage value for outputs 1 & 4 (respectively). Use VOUT_OV_FAULT_LIMIT for PAGE 2 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 2 and 3.
3	Config 3  buck_output_matrix.vsdx	Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use VOUT_OV_FAULT_LIMIT for PAGE 1 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 1 and 2. Use VOUT_OV_FAULT_LIMIT for PAGE 3 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 3 and 4.
4	Config 4  buck_output_matrix.vsdx	Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use VOUT_OV_FAULT_LIMIT for PAGE 1 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 1, 2 and 3. Use VOUT_OV_FAULT_LIMIT for PAGE 4 to set (or read back) the commanded VOUT fault limit voltage value for output 4.
5	Config 5  buck_output_matrix.vsdx	All outputs are combined.	Use VOUT_OV_FAULT_LIMIT for PAGE 1 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

Although there is nothing hard-coded in the PE24103 hardware, the following relative magnitudes are one interpretation of how the various VOUT related voltage levels could be set, so that all the values play a part in the device operation:

MFR_VOUT_MAX >=

VOUT_OV_FAULT_LIMIT >=

POWER_GOOD_ON >=

POWER_GOOD_OFF >=

VOUT_UV_FAULT_LIMIT >=

MFR_VOUT_MIN

The values written using VOUT_OV_FAULT_LIMIT are PAGED. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The VOUT_OV_FAULT_LIMIT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.20 VOUT_UV_FAULT_LIMIT (Command Code 44h)

The VOUT_UV_FAULT_LIMIT command is read/write and sets the value of the output voltage that causes an output undervoltage fault.

The two data bytes for the VOUT_UV_FAULT_LIMIT use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit when in absolute mode. Alternatively when in relative mode (see the VOUT_MODE command) the value for VOUT_UV_FAULT_LIMIT will be treated as a scaling factor relative to the value for the VOUT_COMMAND. The scaling factor is in the range of 0.0 to 31.99 with an LSB of approximately 0.05%.

The VOUT_UV_FAULT_LIMIT command is PAGED so that each output (or combination of outputs) can be controlled individually. Table 17 has more details on how the VOUT_UV_FAULT_LIMIT is applied for the various configurations that PE24103 can support.

Table 17 PE24103 VOUT_UV_FAULT_LIMIT Command vs Configuration and PAGE Number

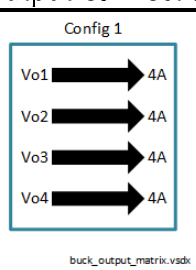
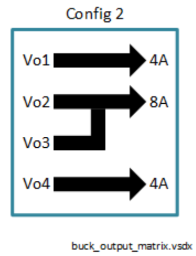
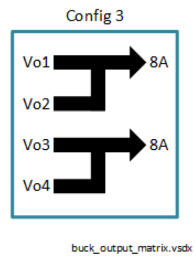
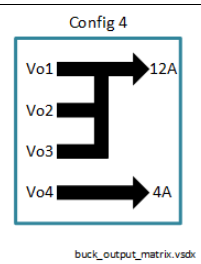
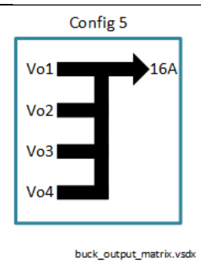
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use VOUT_UV_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded VOUT fault limit voltage value for outputs 1 to 4 (respectively).

Table 17 PE24103 VOUT_UV_FAULT_LIMIT Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use VOUT_UV_FAULT_LIMIT for PAGES 1 & 4 to set (or read back) the commanded VOUT fault limit voltage value for outputs 1 & 4 (respectively). Use VOUT_UV_FAULT_LIMIT for PAGE 2 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use VOUT_UV_FAULT_LIMIT for PAGE 1 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 1 and 2. Use VOUT_UV_FAULT_LIMIT for PAGE 3 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use VOUT_UV_FAULT_LIMIT for PAGE 1 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 1, 2 and 3. Use VOUT_UV_FAULT_LIMIT for PAGE 4 to set (or read back) the commanded VOUT fault limit voltage value for output 4.
5		All outputs are combined.	Use VOUT_UV_FAULT_LIMIT for PAGE 1 to set (or read back) the commanded VOUT fault limit voltage value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

Although there is nothing hard-coded in the PE24103 hardware, the following relative magnitudes are one interpretation of how the various VOUT related voltage levels could be set, so that all the values play a part in the device operation:

MFR_VOUT_MAX >=

VOUT_OV_FAULT_LIMIT >=

POWER_GOOD_ON >=

POWER_GOOD_OFF >=

VOUT_UV_FAULT_LIMIT >=

MFR_VOUT_MIN

The values written using VOUT_UV_FAULT_LIMIT are PAGED. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The VOUT_UV_FAULT_LIMIT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.21 IOUT_OC_FAULT_LIMIT (Command Code 46h)

The IOUT_OC_FAULT_LIMIT command is read/write and sets the value of the output current, in amps, that causes an output overcurrent fault.

The two data bytes for the IOUT_OC_FAULT_LIMIT use a LINEAR11 data format as follows:

Exponent = -5 (5'b1_1011)

Mantissa which is expressed as sign_xxxxx_yyyyy where x are the integer bits and yyyyy are the fractional bits. This allows a range of values from -32.0A to +31.97A to be expressed with a resolution of around 31mA.

The values set using the IOUT_OC_FAULT_LIMIT command apply to each individual switching output and are independent of whether certain outputs may be tied together or not. The value set using the IOUT_OC_FAULT_LIMIT command should be appropriate for the load current from a single switching stage and passing through a single inductor (the current is measured using the differential voltage across the inductor).

Table 18 shows that, regardless of the board level configuration, each switching output has its own current limit, set using its unique PAGE value. Although there is nothing in the PE24103 hardware to enforce this, it is expected that when 2, or more, switching outputs are combined together, the total current load will be shared equally between the connected outputs and so the over-current limit should also be set to the same value for the connected outputs.

Table 18 PE24103 IOUT_OC_FAULT_LIMIT Command vs Configuration and PAGE Number

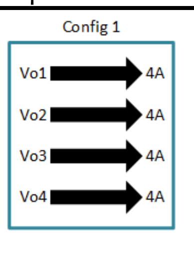
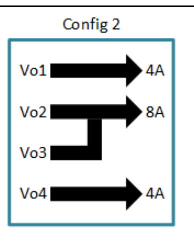
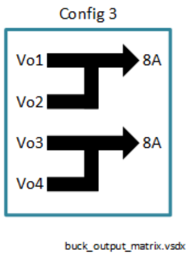
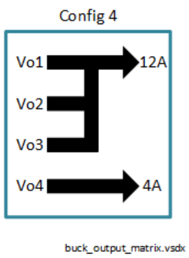
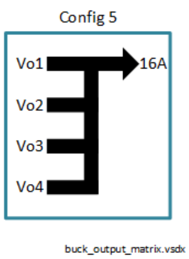
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use IOUT_OC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use IOUT_OC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively).

Table 18 PE24103 IOUT_OC_FAULT_LIMIT Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use IOUT_OC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively).
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use IOUT_OC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively).
5		All outputs are combined.	Use IOUT_OC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively). PAGE 255 may also be used for write commands and will have the same effect assuming that (as expected) all IOUT_OC_FAULT_LIMIT numbers are set to the same value.

The values written using IOUT_OC_FAULT_LIMIT are PAGED. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The IOUT_OC_FAULT_LIMIT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.22 IOUT_UC_FAULT_LIMIT (Command Code 4Bh)

The IOUT_UC_FAULT_LIMIT command is read/write and sets the value of the output current, in amps, that causes an output undercurrent fault.

The two data bytes for the IOUT_UC_FAULT_LIMIT use a LINEAR11 data format as follows:

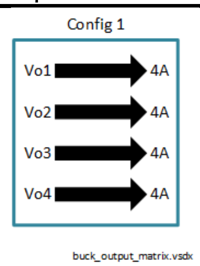
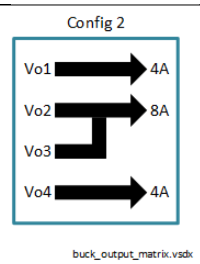
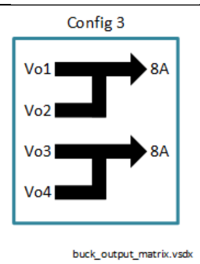
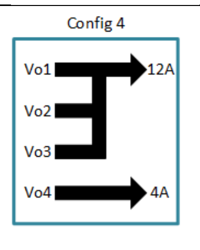
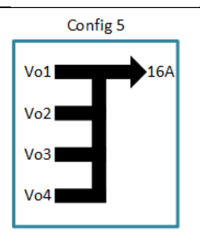
Exponent = -5 (5'b1_1011)

Mantissa which is expressed as sign_xxxxx_yyyyy where x are the integer bits and yyyyy are the fractional bits. This allows a range of values from -32.0A to +31.97A to be expressed with a resolution of around 31mA.

The values set using the IOUT_UC_FAULT_LIMIT command apply to each individual switching output and are independent of whether certain outputs may be tied together or not. The value set using the IOUT_UC_FAULT_LIMIT command should be appropriate for the load current from a single switching stage and passing through a single inductor (the current is measured using the differential voltage across the inductor).

Table 19 shows that, regardless of the board level configuration, each switching output has its own current limit, set using its unique PAGE value. Although there is nothing in the PE24103 hardware to enforce this, it is expected that when 2, or more, switching outputs are combined together, the total current load will be shared equally between the connected outputs and so the under-current limit should also be set to the same value for the connected outputs.

Table 19 PE24103 IOUT_UC_FAULT_LIMIT Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use IOUT_UC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use IOUT_UC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively).
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use IOUT_UC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively).
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use IOUT_UC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively).
5		All outputs are combined.	Use IOUT_UC_FAULT_LIMIT with PAGES 1 to 4 to set (or read back) the commanded IOUT fault limit current value for outputs 1 to 4 (respectively). PAGE 255 may also be used for write commands and will have the same effect assuming that (as expected) all IOUT_UC_FAULT_LIMIT numbers are set to the same value.

The values written using IOUT_UC_FAULT_LIMIT are PAGED. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The IOUT_UC_FAULT_LIMIT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.23 OT_FAULT_LIMIT (Command Code 4Fh)

The OT_FAULT_LIMIT command is read/write and sets the nominal value of the internal die temperature (in Celsius), that causes an over-temperature fault.

The data format for the OT_FAULT_LIMIT uses a LINEAR11 data format as follows:

Exponent = -2 (5'b1_1110)

Mantissa which is expressed as sign_xxxx_xxxx.yy where x are the integer bits and y are the fractional bits. This allows a range of values from -256.0C to +255.75C to be expressed.

Note that although the temperature fault limit can be set with a resolution of 0.25C, there is no implication that the underlying temperature sensor is calibrated to that level of accuracy.

The OT_FAULT_LIMIT is not affected by the PAGE number. The value is stored in NVM and will be saved by using the STORE_USER_ALL and restored with the RESTORE_USER_ALL command.

The OT_FAULT_LIMIT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.24 VIN_OV_FAULT_LIMIT (Command Code 55h)

The VIN_OV_FAULT_LIMIT command is read/write and sets the value of the input voltage, in volts, that causes an input overvoltage fault.

The two data bytes for the VIN_OV_FAULT_LIMIT use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

Although there is nothing hard-coded in the PE24103 hardware, the following relative magnitudes are one way to set VIN related values so that all the possible thresholds play a part in the operation of VIN:

$$MFR\ VIN\ MAX > VIN\ OV > \{VIN\ ON|VIN\ UV\} > VIN\ OFF > MFR\ VIN\ MIN$$

Figure 11 shows a sketch of how the VIN related thresholds may be set in a typical application.

The VIN_OV_FAULT_LIMIT is not affected by the PAGE number. The value is stored in NVM and will be saved by using the STORE_USER_ALL and restored with the RESTORE_USER_ALL command.

The VIN_OV_FAULT_LIMIT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.25 VIN_UV_FAULT_LIMIT (Command Code 59h)

The VIN_UV_FAULT_LIMIT command is read/write and sets the value of the input voltage, in volts, that causes an input undervoltage fault.

The two data bytes for the VIN_UV_FAULT_LIMIT use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

Although there is nothing hard-coded in the PE24103 hardware, the following relative magnitudes are one way to set VIN related values so that all the possible thresholds play a part in the operation of VIN:

$$MFR\ VIN\ MAX > VIN\ OV > \{VIN\ ON|VIN\ UV\} > VIN\ OFF > MFR\ VIN\ MIN$$

Figure 11 shows a sketch of how the VIN related thresholds may be set in a typical application.

The VIN_UV_FAULT_LIMIT is not affected by the PAGE number. The value is stored in NVM and will be saved by using the STORE_USER_ALL and restored with the RESTORE_USER_ALL command.

The VIN_UV_FAULT_LIMIT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.26 IIN_OC_FAULT_LIMIT (Command Code 5Bh)

The IIN_OC_FAULT_LIMIT command is read/write and sets the value of the input current, in amps, that causes an input overcurrent fault.

The two data bytes for the IIN_OC_FAULT_LIMIT command use a LINEAR11 data format as follows:

Exponent = -5 (5’b1_1011)

Mantissa which is expressed as sign_xxxxx_yyyyy where x are the integer bits and yyyyy are the fractional bits. This allows a range of values from -32.0A to +31.97A to be expressed with a resolution of around 31mA.

The IIN_OC_FAULT_LIMIT is not affected by the PAGE number. The value is stored in NVM and will be saved by using the STORE_USER_ALL and restored with the RESTORE_USER_ALL command.

The IIN_OC_FAULT_LIMIT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.27 POWER_GOOD_ON (Command Code 5Eh)

The POWER_GOOD_ON command is read/write and sets the output voltage when a nominal “power is good” state has been reached for one, or more, of the PE24103 switching outputs. When the voltage exceeds the value set by the POWER_GOOD_ON command that output will be considered to have reached a “good” state. PE24103 has a single power good output pin and the effect of a single PE24103 switching output reaching a “good” status on the common power good output pin can be user controlled (with some limitations) by the MFR_SPECIFIC_PGOOD command.

This command is PAGED.

The two data bytes for POWER_GOOD_ON use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit when in absolute mode. Alternatively when in relative mode (see the VOUT_MODE command) the value for POWER_GOOD_ON will be treated as a scaling factor relative to the value for the VOUT_COMMAND. The scaling factor is in the range of 0.0 to 31.99 with an LSB of approximately 0.05%.

Table 20 PE24103 POWER_GOOD_ON Command vs Configuration and PAGE Number

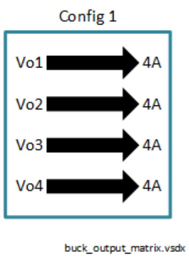
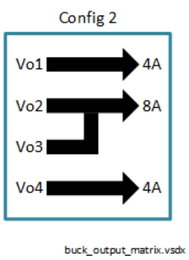
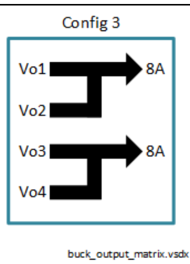
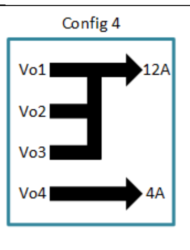
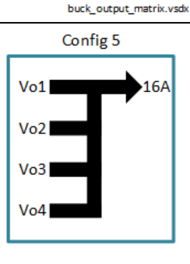
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use POWER_GOOD_ON with PAGES 1 to 4 to set (or read back) the commanded VOUT high "good" threshold values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use POWER_GOOD_ON for PAGES 1 & 4 to set (or read back) the commanded VOUT high "good" threshold value for outputs 1 & 4 (respectively). Use POWER_GOOD_ON for PAGE 2 to set (or read back) the commanded VOUT high "good" threshold value for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use POWER_GOOD_ON for PAGE 1 to set (or read back) the commanded VOUT high "good" threshold value for shared outputs 1 and 2. Use POWER_GOOD_ON for PAGE 3 to set (or read back) the commanded VOUT high "good" threshold value for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use POWER_GOOD_ON for PAGE 1 to set (or read back) the commanded VOUT high "good" threshold value for shared outputs 1, 2 and 3. Use POWER_GOOD_ON for PAGE 4 to set (or read back) the commanded VOUT high "good" threshold value for output 4.
5		All outputs are combined.	Use POWER_GOOD_ON for PAGE 1 to set (or read back) the commanded VOUT high "good" threshold value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

Table 20 PE24103 POWER_GOOD_ON Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used

The values written using the POWER_GOOD_ON command are PAGED as noted in Table 20. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The POWER_GOOD_ON command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.28 POWER_GOOD_OFF (Command Code 5Fh)

The POWER_GOOD_OFF command is read/write and sets the output voltage when a nominal "power is bad" state has been reached for one, or more, of the PE24103 switching outputs. When the voltage drops below the value set by the POWER_GOOD_OFF command that output will be considered to have reached a "bad" state. PE24103 has a single power good output pin and the effect of a single PE24103 switching output reaching a "bad" status on the common power good output pin can be user controlled (with some limitations) by the MFR_SPECIFIC_PGOOD command.

This command is PAGED.

The two data bytes for POWER_GOOD_OFF use a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit when in absolute mode. Alternatively when in relative mode (see the VOUT_MODE command) the value for POWER_GOOD_OFF will be treated as a scaling factor relative to the value for the VOUT_COMMAND. The scaling factor is in the range of 0.0 to 31.99 with an LSB of approximately 0.05%.

Table 21 PE24103 POWER_GOOD_OFF Command vs Configuration and PAGE Number

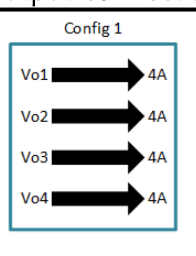
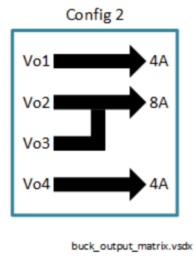
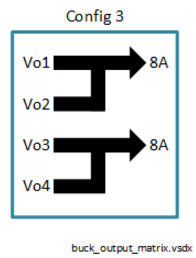
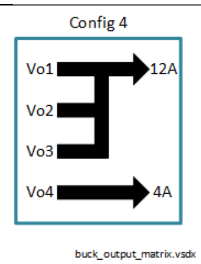
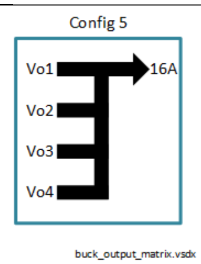
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use POWER_GOOD_OFF with PAGES 1 to 4 to set (or read back) the commanded VOUT high "bad" threshold values for outputs 1 to 4 (respectively).

Table 21 PE24103 POWER_GOOD_OFF Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use POWER_GOOD_OFF for PAGES 1 & 4 to set (or read back) the commanded VOUT high "bad" threshold value for outputs 1 & 4 (respectively). Use POWER_GOOD_OFF for PAGE 2 to set (or read back) the commanded VOUT high "bad" threshold value for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use POWER_GOOD_OFF for PAGE 1 to set (or read back) the commanded VOUT high "bad" threshold value for shared outputs 1 and 2. Use POWER_GOOD_OFF for PAGE 3 to set (or read back) the commanded VOUT high "bad" threshold value for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use POWER_GOOD_OFF for PAGE 1 to set (or read back) the commanded VOUT high "bad" threshold value for shared outputs 1, 2 and 3. Use POWER_GOOD_OFF for PAGE 4 to set (or read back) the commanded VOUT high "bad" threshold value for output 4.
5		All outputs are combined.	Use POWER_GOOD_OFF for PAGE 1 to set (or read back) the commanded VOUT high "bad" threshold value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

The values written using the POWER_GOOD_OFF command are PAGED as noted in Table 21. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The POWER_GOOD_OFF command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.29 TON_DELAY (Command Code 60h)

The TON_DELAY command sets the time, in milliseconds, from when a start condition is received (as programmed by the ON_OFF_CONFIG command) until the output voltage starts to rise.

A value of 0 milliseconds means that PE24103 should start to ramp the output voltage immediately.

The two data bytes for the TON_DELAY command are defined using a LINEAR11 data format as follows:

Exponent = 0 (5'b0_0000)

Mantissa which is expressed as sign_xx_xxxx_xxxx where x are the integer bits (in units of milliseconds) with no fractional bits. This allows a range of values from 0ms to +1023ms to be defined.

Note that although the LINEAR11 data format does allow for negative data values, a negative number would have no meaning in the context of the TON_DELAY command and will be treated as 0ms.

For ES1/ES2 silicon if a negative value is written to the TON_DELAY command, the value will be treated as 0ms. The sign bit will always read as positive (regardless of the value written). The mantissa bits will read back as written.

For silicon after ES2, writing a negative value to the TON_DELAY command will trigger the bad data bit in the STATUS_CML register and will be treated the same as a value of all 0s. Reading back a negative TON_DELAY value will return a value of all 0s.

Table 22 PE24103 TON_DELAY Command vs Configuration and PAGE Number

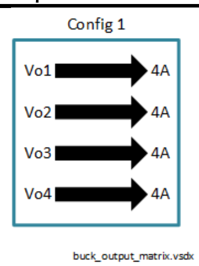
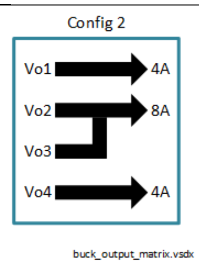
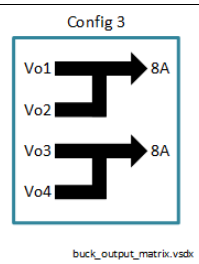
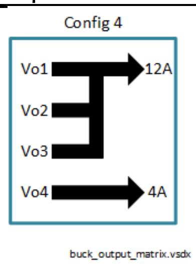
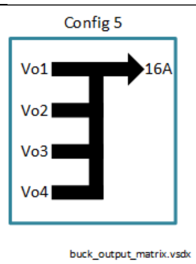
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use TON_DELAY with PAGES 1 to 4 to set (or read back) the commanded turn on delay time values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use TON_DELAY for PAGES 1 & 4 to set (or read back) the commanded turn on delay time value for outputs 1 & 4 (respectively). Use TON_DELAY for PAGE 2 to set (or read back) the commanded turn on delay time value for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use TON_DELAY for PAGE 1 to set (or read back) the commanded turn on delay time value for shared outputs 1 and 2. Use TON_DELAY for PAGE 3 to set (or read back) the commanded turn on delay time value for shared outputs 3 and 4.

Table 22 PE24103 TON_DELAY Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use TON_DELAY for PAGE 1 to set (or read back) the commanded turn on delay time value for shared outputs 1, 2 and 3. Use TON_DELAY for PAGE 4 to set (or read back) the commanded turn on delay time value for output 4.
5		All outputs are combined.	Use TON_DELAY for PAGE 1 to set (or read back) the commanded turn on delay time value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

The values written using the TON_DELAY command are PAGED as noted in Table 22. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The TON_DELAY command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.30 TON_RISE (Command Code 61h)

The TON_RISE command sets the time, in milliseconds, from the end of the turn-on delay time (as set by TON_DELAY) until the output voltage has reached the target voltage range.

A value of 0 milliseconds means that PE24103 should ramp the output voltage up as fast as it can.

The two data bytes for the TON_RISE command are defined using a LINEAR11 data format as follows:

Exponent = 0 (5'b0_0000)

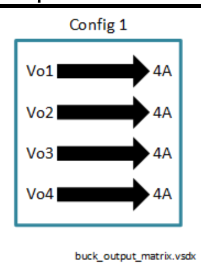
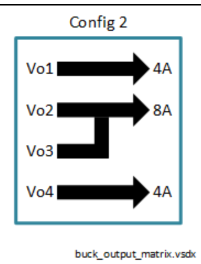
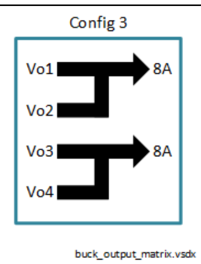
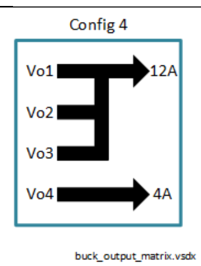
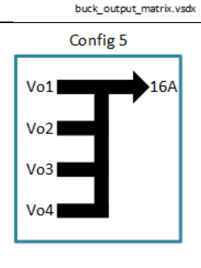
Mantissa which is expressed as sign_xx_xxxx_xxxx where x are the integer bits (in units of milliseconds) with no fractional bits. This allows a range of values from 0ms to +1023ms to be defined.

Note that although the LINEAR11 data format does allow for negative data values, a negative number would have no meaning in the context of the TON_RISE command.

For ES1/ES2 silicon if a negative value is written to the TON_RISE command, the sign bit will be ignored. This may lead to unexpected results as small negative values could be interpreted as large positive values when the sign bit is ignored. The TON_RISE value will always read as positive.

For silicon after ES2, writing a negative value to the TON_RISE command will trigger the bad data bit in the STATUS_CML register and will be treated the same as a value of all 0s which means the fastest rise time supported by PE24103. Reading back a negative TON_RISE value will return a value of all 0s.

Table 23 PE24103 TON_RISE Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use TON_RISE with PAGES 1 to 4 to set (or read back) the commanded turn on ramp time values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use TON_RISE for PAGES 1 & 4 to set (or read back) the commanded turn on ramp time value for outputs 1 & 4 (respectively). Use TON_RISE for PAGE 2 to set (or read back) the commanded turn on ramp time value for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use TON_RISE for PAGE 1 to set (or read back) the commanded turn on ramp time value for shared outputs 1 and 2. Use TON_RISE for PAGE 3 to set (or read back) the commanded turn on ramp time value for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use TON_RISE for PAGE 1 to set (or read back) the commanded turn on ramp time value for shared outputs 1, 2 and 3. Use TON_RISE for PAGE 4 to set (or read back) the commanded turn on ramp time value for output 4.
5		All outputs are combined.	Use TON_RISE for PAGE 1 to set (or read back) the commanded turn on ramp time value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

The values written using the TON_RISE command are PAGED as noted in Table 23. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The TON_RISE command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.31 TOFF_DELAY (Command Code 64h)

The TOFF_DELAY command sets the time, in milliseconds, from when a stop condition is received (as programmed by the ON_OFF_CONFIG command) until the output voltage is allowed to start falling.

A value of 0 milliseconds means that PE24103 should start to ramp the output voltage immediately.

The two data bytes for the TOFF_DELAY command are defined using a LINEAR11 data format as follows:

Exponent = 0 (5'b0_0000)

Mantissa which is expressed as sign_xx_xxxx_xxxx where x are the integer bits (in units of milliseconds) with no fractional bits. This allows a range of values from 0ms to +1023ms to be defined.

Note that although the LINEAR11 data format does allow for negative data values, a negative number would have no meaning in the context of the TOFF_DELAY command and will be treated as 0ms.

For ES1/ES2 silicon if a negative value is written to the TOFF_DELAY command, the value will be treated as 0ms. The sign bit will always read as positive (regardless of the value written). The mantissa bits will read back as written.

For silicon after ES2, writing a negative value to the TOFF_DELAY command will trigger the bad data bit in the STATUS_CML register and will be treated the same as a value of all 0s. Reading back a negative TOFF_DELAY value will return a value of all 0s.

Table 24 PE24103 TOFF_DELAY Command vs Configuration and PAGE Number

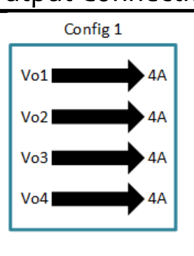
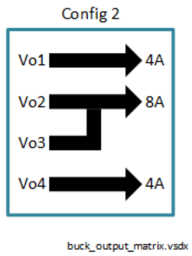
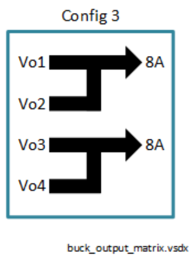
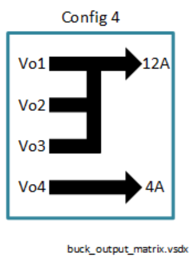
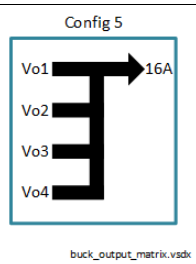
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use TOFF_DELAY with PAGES 1 to 4 to set (or read back) the commanded turn off delay time values for outputs 1 to 4 (respectively).

Table 24 PE24103 TOFF_DELAY Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use TOFF_DELAY for PAGES 1 & 4 to set (or read back) the commanded turn off delay time value for outputs 1 & 4 (respectively). Use TOFF_DELAY for PAGE 2 to set (or read back) the commanded turn off delay time value for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use TOFF_DELAY for PAGE 1 to set (or read back) the commanded turn off delay time value for shared outputs 1 and 2. Use TOFF_DELAY for PAGE 3 to set (or read back) the commanded turn off delay time value for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use TOFF_DELAY for PAGE 1 to set (or read back) the commanded turn off delay time value for shared outputs 1, 2 and 3. Use TOFF_DELAY for PAGE 4 to set (or read back) the commanded turn off delay time value for output 4.
5		All outputs are combined.	Use TOFF_DELAY for PAGE 1 to set (or read back) the commanded turn off delay time value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

The values written using the TOFF_DELAY command are PAGED as noted in Table 24. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The TOFF_DELAY command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.32 TOFF_FALL (Command Code 65h)

The TOFF_FALL command sets the time, in milliseconds, from the end of the turn-off delay time (as set by TOFF_DELAY) until the output voltage is commanded to zero. Note that this command requires that PE24103 can sink enough current to cause the output voltage to decrease at a controlled rate.

A value of 0 milliseconds means that PE24103 should ramp the output voltage down as fast as it can.

The two data bytes for the TOFF_FALL command are defined using a LINEAR11 data format as follows:

Exponent = 0 (5'b0_0000)

Mantissa which is expressed as sign_xx_xxxx_xxxx where x are the integer bits (in units of milliseconds) with no fractional bits. This allows a range of values from 0ms to +1023ms to be defined.

Note that although the LINEAR11 data format does allow for negative data values, a negative number would have no meaning in the context of the TOFF_FALL command.

For ES1/ES2 silicon if a negative value is written to the TOFF_FALL command, the sign bit will be ignored. This may lead to unexpected results as small negative values could be interpreted as large positive values when the sign bit is ignored. The TOFF_FALL value will always read as positive.

For silicon after ES2, writing a negative value to the TOFF_FALL command will trigger the bad data bit in the STATUS_CML register and will be treated the same as a value of all 0s which means the fastest fall time supported by PE24103. Reading back a negative TOFF_FALL value will return a value of all 0s.

Table 25 PE24103 TOFF_FALL Command vs Configuration and PAGE Number

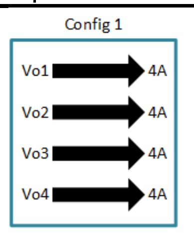
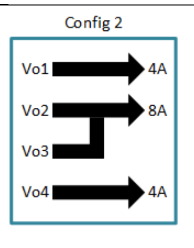
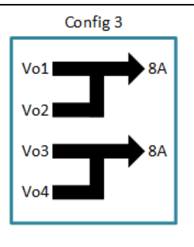
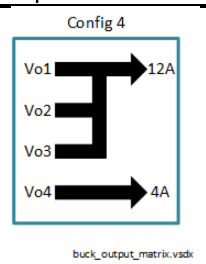
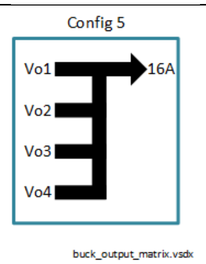
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Use TOFF_FALL with PAGES 1 to 4 to set (or read back) the commanded turn off ramp time values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Use TOFF_FALL for PAGES 1 & 4 to set (or read back) the commanded turn off ramp time value for outputs 1 & 4 (respectively). Use TOFF_FALL for PAGE 2 to set (or read back) the commanded turn off ramp time value for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Use TOFF_FALL for PAGE 1 to set (or read back) the commanded turn off ramp time value for shared outputs 1 and 2. Use TOFF_FALL for PAGE 3 to set (or read back) the commanded turn off ramp time value for shared outputs 3 and 4.

Table 25 PE24103 TOFF_FALL Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Use TOFF_FALL for PAGE 1 to set (or read back) the commanded turn off ramp time value for shared outputs 1, 2 and 3. Use TOFF_FALL for PAGE 4 to set (or read back) the commanded turn off ramp time value for output 4.
5		All outputs are combined.	Use TOFF_FALL for PAGE 1 to set (or read back) the commanded turn off ramp time value for shared outputs 1, 2, 3 and 4. PAGE 255 may also be used for write commands and will have the same effect.

The values written using the TOFF_FALL command are PAGED as noted in Table 25. The values are also stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The TOFF_FALL command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.33 STATUS_BYTE (Command Code 78h)

The STATUS_BYTE command is read/write and has a single data byte associated with the command. Both the STATUS_BYTE and STATUS_WORD commands provide summary information from the various other status registers that PE24103 supports. Figure 12 shows a summary of the PE24103 Status registers and how they are combined to generate the data for STATUS_BYTE and STATUS_WORD.

STATUS_BYTE is a PAGED command, although some of the bits reported using the STATUS_BYTE command are shared across all pages.

The bits of the STATUS_BYTE data byte are defined in Table 26.

Table 26 PE24103 STATUS_BYTE Command Data Byte Definition

Bit	Meaning	PE24103 Value	Description
7	BUSY		Common to all PAGEs
6	OFF		Unique value for each PAGE
5	VOUT overvoltage fault		Unique value for each PAGE
4	IOUT overcurrent fault		Unique value for each PAGE
3	VIN undervoltage fault		Common to all PAGEs

Table 26 PE24103 STATUS_BYTE Command Data Byte Definition

Bit	Meaning	PE24103 Value	Description
2	Temperature fault		Common to all PAGEs
1	Comms, memory or logic fault detected		Common to all PAGEs
0	None of the above		Unique value for each PAGE

Bits of the status byte are generally cleared by clearing the bits in the individual status registers which get summarized in the STATUS_BYTE value. There is one exception, the BUSY bit can be cleared by writing a data value of 0x80 to the STATUS_BYTE command.

The OFF bit is a live reflection of the PE24103 device status and is neither cleared, nor set, by faults. It continually reflects the device status for the respective VOUT1/2/3/4 power stage at all times.

The STATUS_BYTE data value will be returned to 0x00 (no faults – except for the OFF bit) by the CLEAR_FAULTS command or by all 4 outputs of PE24103 being turned off and then on again using appropriate OPERATION commands. The STATUS_BYTE value will also be cleared by any reset of the digital block in PE24103.

The STATUS_BYTE value is not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The STATUS_BYTE command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

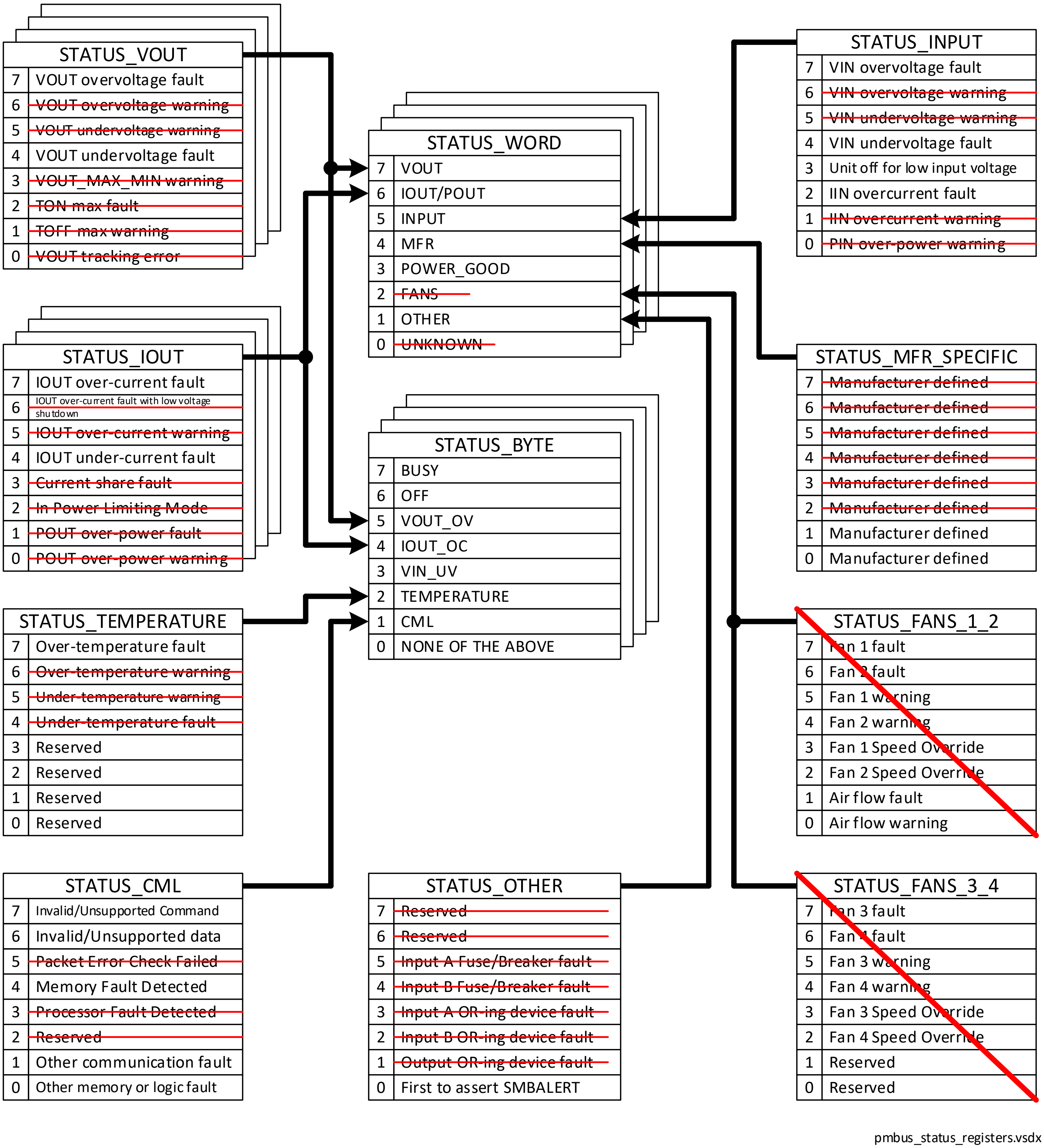


Figure 12 PE24103 Summary of the Status Registers

5.1.34 STATUS_WORD (Command Code 79h)

The STATUS_WORD command is read/write and has two data bytes associated with the command. The least significant byte of the STATUS_WORD is also accessible using the STATUS_BYTE command. Both the STATUS_BYTE and STATUS_WORD commands provide summary information from the various other status registers that PE24103 supports. Figure 12 shows a summary of the PE24103 Status registers and how they are combined to generate the data for STATUS_BYTE and STATUS_WORD.

STATUS_WORD is a PAGED command, although some of the bits reported using the STATUS_WORD command are shared across all pages.

The bits of the upper byte of the STATUS_WORD value are defined in Table 27.

Table 27 PE24103 STATUS_WORD Command Upper Data Byte Definition

Bit	Meaning	PE24103 Value	Description
7	VOUT		Unique value for each PAGE
6	IOUT		Unique value for each PAGE
5	INPUT		Common to all PAGEs
4	MFR SPECIFIC		Common to all PAGEs
3	PG Status		Common to all PAGEs
2	FANS	Not supported by PE24103	
1	OTHER		Common to all PAGEs
0	UNKNOWN	Not supported by PE24103	

Bits of the upper status byte are generally cleared by clearing the bits in the individual status registers which get summarized in the STATUS_WORD value.

The PG Status bit is a live reflection of the PE24103 device status and is neither cleared, nor set, by faults. It continually reflects the state of the PE24103 Power Good pin at all times.

The STATUS_WORD data value will be returned to 0x00 (no faults – except for the OFF and PG Status bits) by the CLEAR_FAULTS command or by all 4 outputs of PE24103 being turned off and then on again using appropriate OPERATION commands. The STATUS_WORD value will also be cleared by any reset of the digital block in PE24103.

The STATUS_WORD value is not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The STATUS_WORD command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.35 STATUS_VOUT (Command Code 7Ah)

The STATUS_VOUT command is read/write and has a single data byte associated with the command. The bits of the data byte are defined in Table 28.

The STATUS_VOUT register is PAGED and the VOUT status of individual PE24103 outputs can be found by reading the STATUS_VOUT value from the appropriate PAGE number. Table 29 has more details.

Table 28 PE24103 STATUS_VOUT Command Data Byte Definition

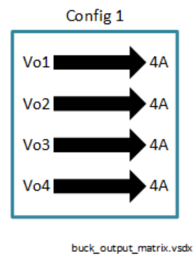
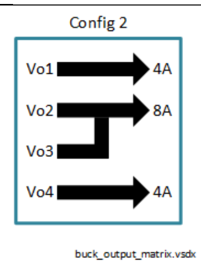
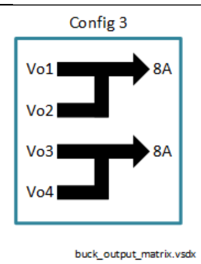
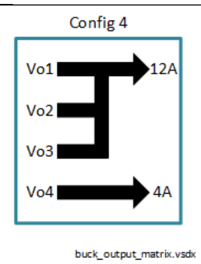
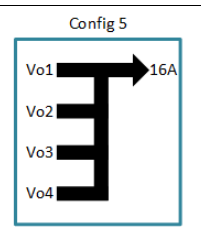
Bit	Meaning	PE24103 Value	Description
7	VOUT overvoltage fault flag		
6	VOUT overvoltage warning flag	Not supported by PE24103	
5	VOUT undervoltage warning flag	Not supported by PE24103	
4	VOUT undervoltage fault flag		
3	VOUT_MAX_MIN fault	Not supported by PE24103	
2	TON time exceeded fault flag	Not supported by PE24103	
1	TOFF time exceeded warning flag	Not supported by PE24103	
0	VOUT tracking error flag	Not supported by PE24103	

As with any of the other status related commands, individual bits of the STATUS_VOUT register can be cleared by writing to the STATUS_VOUT command for the appropriate PAGE. The data byte is a binary value. A 1 in any bit position indicates that bit is to be cleared, if set, and unchanged if not set. Examples of data bytes:

- 8'b0001_0000 indicates that bit [4] is to be cleared and all other bits are to be unchanged;
- 8'b0110_0010 indicates that bits [6], [5], and [1] are to be cleared and all other bits are to be unchanged;
- 8'b1111_1111, or 0xFF, indicates all bits are to be cleared.

The STATUS_VOUT data value for the active PAGE will be returned to 0x00 (no faults) by the CLEAR_FAULTS command or by the selected output of PE24103 being turned off and then on again using appropriate OPERATION commands. The STATUS_VOUT values will also be cleared by any reset of the digital block in PE24103.

Table 29 PE24103 STATUS_VOUT Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Read STATUS_VOUT for PAGES 1 to 4 to get the VOUT status for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Read STATUS_VOUT for PAGES 1 & 4 to get the VOUT status for outputs 1 & 4 (respectively). Read STATUS_VOUT for PAGE 2 for the common VOUT status for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Read STATUS_VOUT for PAGE 1 for the common VOUT status for shared outputs 1 and 2. Read STATUS_VOUT for PAGE 3 for the common VOUT status for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Read STATUS_VOUT for PAGE 1 for the common VOUT status for shared outputs 1, 2 and 3. Read STATUS_VOUT for PAGE 4 for the VOUT status for output 4.
5		All outputs are combined.	Read STATUS_VOUT for PAGE 1 for the common VOUT status for shared outputs 1, 2, 3 and 4.

The STATUS_VOUT values are PAGED as shown in Table 29. As with most of the PE24103 status bits the values are not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The STATUS_VOUT command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.36 STATUS_IOUT (Command Code 7Bh)

The STATUS_IOUT command is read/write and has a single data byte associated with the command. The bits of the data byte are defined in Table 30.

The STATUS_IOUT register is PAGED and the IOUT status of individual PE24103 outputs can be found by reading the STATUS_IOUT value from the appropriate PAGE number. Table 31 has more details.

Table 30 PE24103 STATUS_IOUT Command Data Byte Definition

Bit	Meaning	PE24103 Value	Description
7	IOUT overcurrent fault flag		
6	IOUT overcurrent and low voltage fault flag	Not supported by PE24103	
5	IOUT overcurrent warning flag	Not supported by PE24103	
4	IOUT undercurrent fault flag		
3	Current sharing fault	Not supported by PE24103	
2	Output in power limiting mode flag	Not supported by PE24103	
1	Output overpower fault flag	Not supported by PE24103	
0	Output overpower warning flag	Not supported by PE24103	

As with any of the other status related commands, individual bits of the STATUS_IOUT register can be cleared by writing to the STATUS_IOUT command for the appropriate PAGE. The data byte is a binary value. A 1 in any bit position indicates that bit is to be cleared, if set, and unchanged if not set. Examples of data bytes:

- 8’b0001_0000 indicates that bit [4] is to be cleared and all other bits are to be unchanged;
- 8’b0110_0010 indicates that bits [6], [5], and [1] are to be cleared and all other bits are to be unchanged;
- 8’b1111_1111, or 0xFF, indicates all bits are to be cleared.

The STATUS_IOUT data value for the active PAGE will be returned to 0x00 (no faults) by the CLEAR_FAULTS command or by the selected output of PE24103 being turned off and then on again using appropriate OPERATION commands. The STATUS_IOUT values will also be cleared by any reset of the digital block in PE24103.

Note that there are some minor differences in behaviour for over/under-current faults in the ES1/Es2 versions of PE24103 and later versions.

For silicon versions later than ES2, the matrix versions of the (internal) High Side Overcurrent (HSOC) and Low Side Overcurrent (LSOC) signals have been modified so that error flags which have already been included into the master PAGE value do not also remain in the slave page. This means that if VOUT2 is controlled by VOUT1 then HSOC/LSOC faults for BUCK2 will be routed to PAGE1 and BUCK1 control and will no longer set the OC fault flags in PAGE2 status registers. This means the fault is still captured as required, and can still shutdown the BUCKs as required, but is only flagged in one place. That makes the clean-up of the STATUS registers easier.

With ES1/ES2 silicon an over/under-current fault flag can set fault bits in both the master PAGE STATUS_IOUT value AND in the slave PAGE as well. This doesn't affect the fault behaviour, but does make user clean-up of the STATUS_IOUT registers harder and less obvious.

Table 31 PE24103 STATUS_IOUT Command vs Configuration and PAGE Number

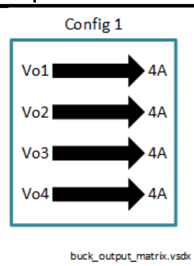
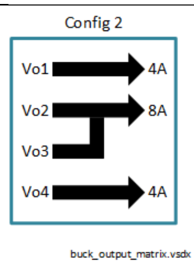
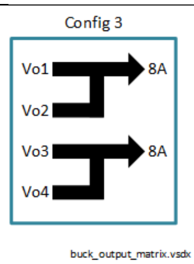
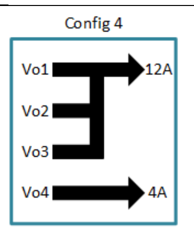
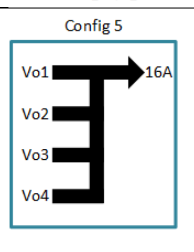
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	Read STATUS_IOUT for PAGES 1 to 4 to get the IOUT status for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	Read STATUS_IOUT for PAGES 1 & 4 to get the IOUT status for outputs 1 & 4 (respectively). Read STATUS_IOUT for PAGE 2 for the common IOUT status for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	Read STATUS_IOUT for PAGE 1 for the common IOUT status for shared outputs 1 and 2. Read STATUS_IOUT for PAGE 3 for the common IOUT status for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	Read STATUS_IOUT for PAGE 1 for the common IOUT status for shared outputs 1, 2 and 3. Read STATUS_IOUT for PAGE 4 for the IOUT status for output 4.
5		All outputs are combined.	Read STATUS_IOUT for PAGE 1 for the common IOUT status for shared outputs 1, 2, 3 and 4.

Table 31 PE24103 STATUS_IOUT Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used

The STATUS_IOUT values are PAGED as shown in Table 31. As with most of the PE24103 status bits the values are not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The STATUS_IOUT command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.37 STATUS_INPUT (Command Code 7Ch)

The STATUS_INPUT command is read/write and has a single data byte associated with the command. The bits of the data byte are defined in Table 32.

Table 32 PE24103 STATUS_INPUT Command Data Byte Definition

Bit	Meaning	PE24103 Value	Description
7	Input over-voltage fault flag		
6	Input over-voltage warning flag	Not supported by PE24103	
5	Input under-voltage warning flag	Not supported by PE24103	
4	Input under-voltage fault flag		
3	PE24103 off due to insufficient input voltage		
2	Input Overcurrent fault flag		
1	Input overcurrent warning flag	Not supported by PE24103	
0	Input over-power warning	Not supported by PE24103	

As with any of the other status related commands, individual bits of the STATUS_INPUT register can be cleared by writing to the STATUS_INPUT command. The data byte is a binary value. A 1 in any bit position indicates that bit is to be cleared, if set, and unchanged if not set. Examples of data bytes:

- 8'b0001_0000 indicates that bit [4] is to be cleared and all other bits are to be unchanged;
- 8'b0110_0010 indicates that bits [6], [5], and [1] are to be cleared and all other bits are to be unchanged;
- 8'b1111_1111, or 0xFF, indicates all bits are to be cleared.

The STATUS_INPUT data value will be returned to 0x00 (no faults) by the CLEAR_FAULTS command or by all 4 outputs of PE24103 being turned off and then on again using appropriate OPERATION commands. The STATUS_INPUT value will also be cleared by any reset of the digital block in PE24103.

The STATUS_INPUT value is not affected by the current PAGE. The value is also not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The STATUS_INPUT command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.38 STATUS_TEMPERATURE (Command Code 7Dh)

The STATUS_TEMPERATURE command is read/write and has a single data byte associated with the command. The bits of the data byte are defined in Table 33.

Table 33 PE24103 STATUS_TEMPERATURE Command Data Byte Definition

Bit	Meaning	PE24103 Value	Description
7	Over-temperature fault flag	Supported	
6	Over-temperature warning flag	Not supported by PE24103	
5	Under-temperature warning flag	Not supported by PE24103	
4	Under-temperature fault flag	Not supported by PE24103	
3:0	Reserved	Reserved	

As with any of the other status related commands, individual bits of the STATUS_TEMPERATURE register can be cleared by writing to the STATUS_TEMPERATURE command. The data byte is a binary value. A 1 in any bit position indicates that bit is to be cleared, if set, and unchanged if not set. Examples of data bytes:

- 8'b0001_0000 indicates that bit [4] is to be cleared and all other bits are to be unchanged;
- 8'b0110_0010 indicates that bits [6], [5], and [1] are to be cleared and all other bits are to be unchanged;
- 8'b1111_1111, or 0xFF, indicates all bits are to be cleared.

The STATUS_TEMPERATURE data value will be returned to 0x00 (no faults) by the CLEAR_FAULTS command or by all 4 outputs of PE24103 being turned off and then on again using appropriate OPERATION commands. The STATUS_TEMPERATURE value will also be cleared by any reset of the digital block in PE24103.

The STATUS_TEMPERATURE value is not affected by the current PAGE. The value is also not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The STATUS_TEMPERATURE command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.39 STATUS_CML (Command Code 7Eh)

The STATUS_CML command is read/write and has a single data byte associated with the command. The bits of the data byte are defined in Table 34.

Table 34 PE24103 STATUS_CML Command Data Byte Definition

Bit	Meaning	PE24103 Value	Description
7	Invalid or unsupported command received	Supported	
6	Invalid or unsupported data received	Supported	
5	Packet error check failed	Not supported by PE24103.	
4	Memory Fault Detected	Supported	
3	Processor Fault Detected	Not supported by PE24103.	
2	Reserved	Reserved	
1	A communication fault not covered by other fault bits has occurred.	Supported	
0	A memory or logic fault not covered by other fault bits has occurred.	Supported	

As with any of the other status related commands, individual bits of the STATUS_CML register can be cleared by writing to the STATUS_CML command. The data byte is a binary value. A 1 in any bit position indicates that bit is to be cleared, if set, and unchanged if not set. Examples of data bytes:

- 8'b0001_0000 indicates that bit [4] is to be cleared and all other bits are to be unchanged;
- 8'b0110_0010 indicates that bits [6], [5], and [1] are to be cleared and all other bits are to be unchanged;
- 8'b1111_1111, or 0xFF, indicates all bits are to be cleared.

The STATUS_CML data value will be returned to 0x00 (no faults) by the CLEAR_FAULTS command or by all 4 outputs of PE24103 being turned off and then on again using appropriate OPERATION commands. The STATUS_CML value will also be cleared by any reset of the digital block in PE24103.

The STATUS_CML value is not affected by the current PAGE. The value is also not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The STATUS_CML command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.40 STATUS_OTHER (Command Code 7Fh)

The STATUS_OTHER command is read/write and has a single data byte associated with the command. The bits of the data byte are defined in Table 35.

Table 35 PE24103 STATUS_OTHER Command Data Byte Definition

Bit	Meaning	PE24103 Value	Description
7	Reserved		
6	Reserved		
5	Input A fuse of circuit breaker fault	Not supported by PE24103.	
4	Input B fuse of circuit breaker fault	Not supported by PE24103.	
3	Input A OR-ing Device Fault	Not supported by PE24103.	
2	Input B OR-ing Device Fault	Not supported by PE24103.	
1	Output OR-ing Device Fault	Not supported by PE24103.	
0	First to assert SMBALERT	Supported	

As with any of the other status related commands, individual bits of the STATUS_OTHER register can be cleared by writing to the STATUS_OTHER command. The data byte is a binary value. A 1 in any bit position indicates that bit is to be cleared, if set, and unchanged if not set. Examples of data bytes:

- 8'b0001_0000 indicates that bit [4] is to be cleared and all other bits are to be unchanged;
- 8'b0110_0010 indicates that bits [6], [5], and [1] are to be cleared and all other bits are to be unchanged;
- 8'b1111_1111, or 0xFF, indicates all bits are to be cleared.

The STATUS_OTHER data value will be returned to 0x00 (no faults) by the CLEAR_FAULTS command or by all 4 outputs of PE24103 being turned off and then on again using appropriate OPERATION commands. The STATUS_OTHER value will also be cleared by any reset of the digital block in PE24103.

The STATUS_OTHER value is not affected by the current PAGE. The value is also not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The STATUS_OTHER command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.41 STATUS_MFR_SPECIFIC (Command Code 80h)

The STATUS_MFR_SPECIFIC command is read/write and has a single data byte associated with the command. The bits of the data byte are defined in Table 36.

Table 36 PE24103 STATUS_MFR_SPECIFIC Command Data Byte Definition

Bit	Meaning	PE24103 Value	Description
7	Reserved	Not supported by PE24103.	
6	Reserved	Not supported by PE24103.	
5	Reserved	Not supported by PE24103.	
4	Reserved	Not supported by PE24103.	
3	Reserved	Not supported by PE24103.	
2	Reserved	Not supported by PE24103.	
1	Hardware Over-temperature Fault	Supported	In addition to the user set over-temperature threshold, there is a manufacturer set over-temperature threshold. Breaking the manufacturer threshold will result in an immediate shutdown of PE24103 and will be flagged via this status bit.
0	Charge Pump Not Ready	Supported	When set to 1'b1, this bit indicates the charge pump in PE24103 is not yet ready to support load from the switching outputs. Any TON_DELAY times will not start until the charge pump is ready.

As with any of the other status related commands, individual bits of the STATUS_MFR_SPECIFIC register can be cleared by writing to the STATUS_MFR_SPECIFIC command. The data byte is a binary value. A 1 in any bit position indicates that bit is to be cleared, if set, and unchanged if not set. Examples of data bytes:

- 8'b0001_0000 indicates that bit [4] is to be cleared and all other bits are to be unchanged;
- 8'b0110_0010 indicates that bits [6], [5], and [1] are to be cleared and all other bits are to be unchanged;
- 8'b1111_1111, or 0xFF, indicates all bits are to be cleared.

The STATUS_MFR_SPECIFIC data value will be returned to 0x00 (no faults) by the CLEAR_FAULTS command (although if the fault condition is still present, then the bit may be immediately set again). The STATUS_MFR_SPECIFIC value will also be cleared by any reset of the digital block in PE24103.

The STATUS_MFR_SPECIFIC value is not affected by the current PAGE. The value is also not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The STATUS_MFR_SPECIFIC command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.42 READ_VIN (Command Code 88h)

The READ_VIN command is essentially read only. It is possible (for debug purposes) to write a value using the READ_VIN command, but the value written could be updated at any time by the latest ADC result, so it is possible that the value readback will not match the debug value written.

Reading from PE24103 using the READ_VIN command will return a 2-byte value describing the most recent measured VIN voltage for PE24103. The data returned by a READ_VIN command will be returned using a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

The READ_VIN value is not affected by the current PAGE. The value is also not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The READ_VIN command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.43 READ_IIN (Command Code 89h)

The READ_IIN command is essentially read only. It is possible (for debug purposes) to write a value using the READ_IIN command, but the value written could be updated at any time by the latest ADC result, so it is possible that the value readback will not match the debug value written.

Reading from PE24103 using the READ_IIN command will return a 2-byte value describing the most recent measured load current for the input pin. The data returned by a READ_IIN command will be returned using a LINEAR11 data format as follows:

Exponent = -5 (5’b1_1011)

Mantissa which is expressed as sign_xxxxx_yyyyy where x are the integer bits and yyyyy are the fractional bits. This allows a range of values from -32.0A to +31.97A to be expressed with a resolution of approximately 31mA.

The input current can only be measured when a suitable in-line sense resistor is fitted at the board level.

The READ_IIN value is not affected by the current PAGE. The value is also not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The READ_IIN command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

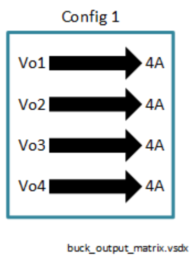
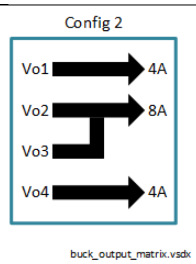
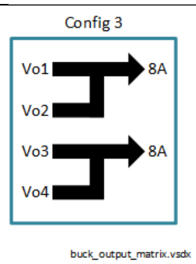
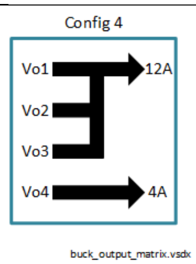
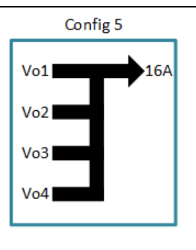
5.1.44 READ_VOUT (Command Code 8Bh)

The READ_VOUT command is essentially read only. It is possible (for debug purposes) to write a value using the READ_VOUT command, but the value written could be updated at any time by the latest ADC result, so it is possible that the value readback will not match the debug value written.

Reading from PE24103 using the READ_VOUT command will return a 2-byte value describing the most recent measured output voltage for the selected PAGE/switching output. The data returned by a READ_VOUT command will be returned using a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

The combined 2-byte value returned by the READ_VOUT command will depend both on the device configuration (whether outputs are connected together at the board level – as defined by the MFR_SPECIFIC_MATRIX value) and the current PAGE number as shown in Table 37.

Table 37 PE24103 READ_VOUT Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	READ_VOUT for PAGES 1 to 4 to get the VOUT values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	READ_VOUT for PAGES 1 & 4 to get the VOUT values for outputs 1 & 4 (respectively). READ_VOUT for PAGE 2 for the common VOUT for shared outputs 2 and 3.
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	READ_VOUT for PAGE 1 for the common VOUT for shared outputs 1 and 2. READ_VOUT for PAGE 3 for the common VOUT for shared outputs 3 and 4.
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	READ_VOUT for PAGE 1 for the common VOUT for shared outputs 1, 2 and 3. READ_VOUT for PAGE 4 for the VOUT for output 4.
5		All outputs are combined.	READ_VOUT for PAGE 1 for the common VOUT for shared outputs 1, 2, 3 and 4.

The READ_VOUT value is not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The READ_VOUT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.45 READ_IOUT (Command Code 8Ch)

The READ_IOUT command is essentially read only. It is possible (for debug purposes) to write a value using the READ_IOUT command, but the value written could be updated at any time by the latest ADC result, so it is possible that the value readback will not match the debug value written.

The IOUT measurements can be averaged on chip by PE24103. The description of the MFR_SPECIFIC_IOUTFILT command gives more details.

Reading from PE24103 using the READ_IOUT command will return a 2-byte value describing the most recent measured load current for the selected PAGE/switching output. The data returned by a READ_IOUT command will be returned using a LINEAR11 data format as follows:

Exponent = -5 (5'b1_1011)

Mantissa which is expressed as sign_xxxxx_yyyyy where x are the integer bits and yyyyy are the fractional bits. This allows a range of values from -32.0A to +31.97A to be expressed with a resolution of approximately 31mA.

The combined 2-byte value returned by the READ_IOUT will always reflect the output current value for the selected PAGE and selected switching output (with a 1 to 1 matching as shown in Table 38).

Table 38 PE24103 READ_IOUT Command vs Configuration and PAGE Number

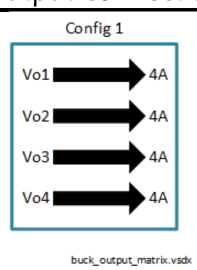
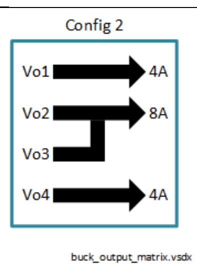
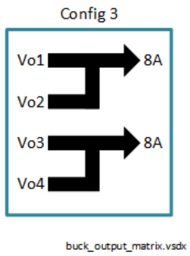
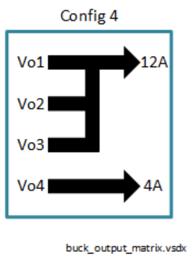
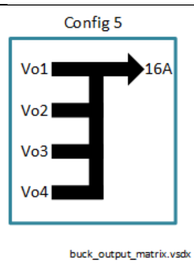
Config	Output Connection	Description	PAGE Numbers Used
1		Four independent outputs.	READ_IOUT for PAGES 1 to 4 to get the IOUT values for outputs 1 to 4 (respectively).
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	READ_IOUT for PAGES 1 to 4 to get the IOUT values for outputs 1 to 4 (respectively).

Table 38 PE24103 READ_IOUT Command vs Configuration and PAGE Number

Config	Output Connection	Description	PAGE Numbers Used
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	READ_IOUT for PAGES 1 to 4 to get the IOUT values for outputs 1 to 4 (respectively).
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	READ_IOUT for PAGES 1 to 4 to get the IOUT values for outputs 1 to 4 (respectively).
5		All outputs are combined.	READ_IOUT for PAGES 1 to 4 to get the IOUT values for outputs 1 to 4 (respectively).

The READ_IOUT value is not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The READ_IOUT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.46 READ_TEMPERATURE_1 (Command Code 8Dh)

The READ_TEMPERATURE_1 command is essentially read only. It is possible (for debug purposes) to write a value using the READ_TEMPERATURE_1 command, but the value written could be updated at any time by the latest ADC result, so it is possible that the value readback will not match the debug value written.

Reading from PE24103 using the READ_TEMPERATURE_1 command will return a 2-byte value describing the most recent measured die temperature for PE24103 in Celsius (as required by PMBus). The data returned by a READ_TEMPERATURE_1 command will be returned using a LINEAR11 data format as follows:

Exponent = -2 (5'b1_1110)

Mantissa which is expressed as sign_xxxx_xxxx.yy where x are the integer bits and y are the fractional bits. This allows a range of values from -256.0C to +255.75C to be expressed.

Note that although temperature is reported with a resolution of 0.25C, there is no implication that the underlying temperature sensor is calibrated to that level of accuracy.

The READ_TEMPERATURE_1 value is not affected by the current PAGE. The value is also not stored in NVM and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The READ_TEMPERATURE_1 command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.47 PMBUS_REVISION (Command Code 98h)

The PMBUS_REVISION command is read only and supplies a single byte of data. The 8-bits of data which are supplied in response to the PMBUS_REVISION command are defined for PE24103 in Table 39.

Table 39 PE24103 PMBUS_REVISION Command Response Byte

Bits	Description	PE24103 Value	Meaning
7:4	Part I Revision	4'b0011	PE24103 is compliant with version 1.3 of the PMBus specification
3:0	Part II Revision	4'b0011	PE24103 is compliant with version 1.3 of the PMBus specification

The value returned by the PMBUS_REVISION command is not affected by the PAGE number and cannot be changed by the user so is not affected by the STORE_USER_ALL or RESTORE_USER_ALL commands.

The PMBUS_REVISION command has a single byte as argument and uses the Read BYTE protocol as shown in Figure 3.

For silicon versions of PE24103 after ES2, attempting to write using the (read only) PMBUS_REVISION command will trigger a fault bit in the STATUS_CML register.

5.1.48 MFR_ID (Command Code 99h)

The MFR_ID command is read only and supplies a single byte of data. The 8-bits of data which are supplied in response to the MFR_ID command are defined for PE24103 in Table 40.

Table 40 PE24103 MFR_ID Command Response Byte

Bits	Description	PE24103 Value	Meaning
7:0	Manufacturer ID value	0xA5	This value comes from NVM so can be set to any 8-bit value we wish. The value A5 is a legacy value used on several previous generations of product.

The value returned by the MFR_ID command is not affected by the PAGE number and cannot be changed by the user so is not affected by the STORE_USER_ALL or RESTORE_USER_ALL commands.

The MFR_ID command uses a BLOCK Read protocol with a single data byte. A generic BLOCK Read packet is shown in Figure 13.

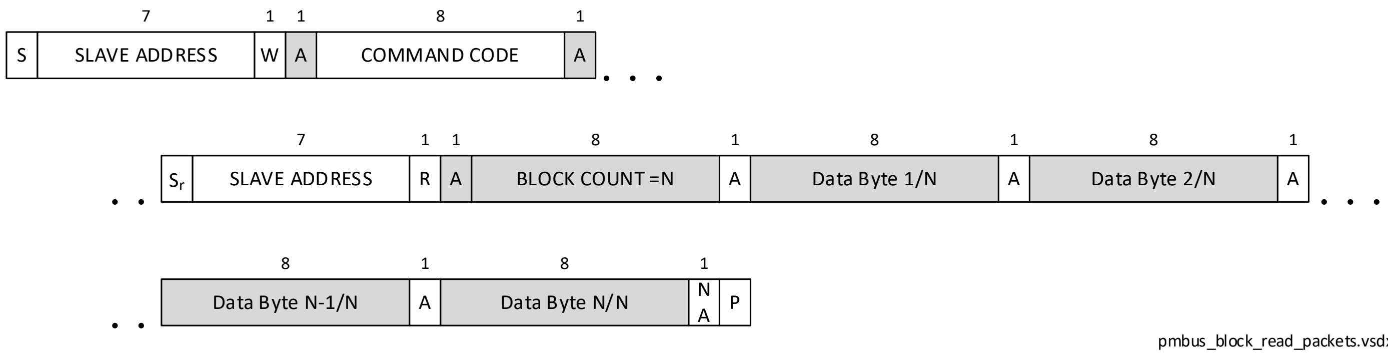


Figure 13 PE24103 Generic BLOCK Read Command Example

5.1.49 MFR_VIN_MIN (Command Code A0h)

The MFR_VIN_MIN command is read only. Attempting to write data using the MFR_VIN_MIN command will have no effect.

Reading from PE24103 using the MFR_VIN_MIN command will return a 2-byte value describing the minimum rated input voltage for PE24103. The expected value from the PRD is 9.6V. The data returned by a MFR_VIN_MIN command will be returned using a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

The expected PRD value of +9.6V would be supplied as the 16-bit value:

16'b01001.100_1100_1101 (0x4CCD)

The combined 2-byte value returned by the MFR_VIN_MIN command is expected to be 0x4CCD (with the least significant byte supplied first). This value comes from a (private) NVM location and will not be controllable by the user.

The MFR_VIN_MIN command is not PAGED and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The MFR_VIN_MIN command has two data bytes as argument and uses the Read WORD protocol as shown in Figure 10.

For silicon versions of PE24103 after Es2, attempting to write using the (read only) MFR_VIN_MIN command will trigger a fault bit in the STATUS_CML register.

5.1.50 MFR_VIN_MAX (Command Code A1h)

The MFR_VIN_MAX command is read only. Attempting to write data using the MFR_VIN_MAX command will have no effect.

Reading from PE24103 using the MFR_VIN_MAX command will return a 2-byte value describing the maximum rated input voltage for PE24103. The expected value from the PRD is 15.5V. The data returned by a MFR_VIN_MAX command will be returned using a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

The expected PRD value of +15.1V would be supplied as the 16-bit value:

16'b01111.100_0000_0000 (0x7C00)

The combined 2-byte value returned by the MFR_VIN_MAX command is expected to be 0x7C00 (with the least significant byte supplied first). This value comes from a (private) NVM location and will not be controllable by the user.

The MFR_VIN_MAX command is not PAGED and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The MFR_VIN_MAX command has two data bytes as argument and uses the Read WORD protocol as shown in Figure 10.

For silicon versions of PE24103 after ES2, attempting to write using the (read only) MFR_VIN_MAX command will trigger a fault bit in the STATUS_CML register.

5.1.51 MFR_IIN_MAX (Command Code A2h)

The MFR_IIN_MAX command is read only. Attempting to write data using the MFR_IIN_MAX command will have no effect.

Reading from PE24103 using the MFR_IIN_MAX command will return a 2-byte value describing the maximum rated output current for a single output from PE24103. The expected value from the PRD is (4 outputs at 4A and 2V = 32W divided by a minimum VIN voltage of 9.6V => 3.33A, so set the value to 3.5A). The data returned by a MFR_IIN_MAX command will be returned using a LINEAR11 data format as follows:

Exponent = -5 (5'b1_1011)

Mantissa = +3.5, which is expressed as sign_xxxxx_yyyyy where x are the integer bits and yyyyy are the fractional bits. For a value of +3.5 the value is

0_00011_10000

The combined 2-byte value returned by the MFR_IIN_MAX command is expected to be 0xD870 (with the least significant byte supplied first). This value comes from a (private) NVM location and will not be controllable by the user.

The MFR_IIN_MAX command is not PAGED and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The MFR_IIN_MAX command has two data bytes as argument and uses the Read WORD protocol as shown in Figure 10.

For silicon versions of PE24103 after Es2, attempting to write using the (read only) MFR_IIN_MAX command will trigger a fault bit in the STATUS_CML register.

5.1.52 MFR_VOUT_MIN (Command Code A4h)

The MFR_VOUT_MIN command is read only. Attempting to write data using the MFR_VOUT_MIN command will have no effect.

Reading from PE24103 using the MFR_VOUT_MIN command will return a 2-byte value describing the minimum rated output voltage for any of the power switching outputs from PE24103. The expected value from the PRD is 0.6V. The data returned by a MFR_VOUT_MIN command will be returned using a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

The expected PRD value of +0.6V would be supplied as the 16-bit value:

16'b00000.100_1100_1101 (0x04CD)

The combined 2-byte value returned by the MFR_VOUT_MIN command is expected to be 0x04CD (with the least significant byte supplied first). This value comes from a (private) NVM location and will not be controllable by the user.

The MFR_VOUT_MIN command is not PAGED and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The MFR_VOUT_MIN command has two data bytes as argument and uses the Read WORD protocol as shown in Figure 10.

For silicon versions of PE24103 after ES2, attempting to write using the (read only) MFR_VOUT_MIN command will trigger a fault bit in the STATUS_CML register.

5.1.53 MFR_VOUT_MAX (Command Code A5h)

The MFR_VOUT_MAX command is read only. Attempting to write data using the MFR_VOUT_MAX command will have no effect.

Reading from PE24103 using the MFR_VOUT_MAX command will return a 2-byte value describing the maximum rated output voltage for any of the power switching outputs from PE24103. The expected value from the PRD is 2.0V. The data returned by a MFR_VOUT_MAX command will be returned using a ULINEAR16 data format with an exponent (defined by the VOUT_MODE command) of -11. This allows the unsigned 16-bit value to represent values in the range of 0.0V to +31.99V with a resolution of approximately 500uV/bit.

The expected PRD value of +2.0V would be supplied as the 16-bit value:

16'b00010.000_0000_0000 (0x1000)

The combined 2-byte value returned by the MFR_VOUT_MAX command is expected to be 0x1000 (with the least significant byte supplied first). This value comes from a (private) NVM location and will not be controllable by the user.

The MFR_VOUT_MAX command is not PAGED and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The MFR_VOUT_MAX command has two data bytes as argument and uses the Read WORD protocol as shown in Figure 10.

For silicon versions of PE24103 after ES2, attempting to write using the (read only) MFR_VOUT_MAX command will trigger a fault bit in the STATUS_CML register.

5.1.54 MFR_IOUT_MAX (Command Code A6h)

The MFR_IOUT_MAX command is read only. Attempting to write data using the MFR_IOUT_MAX command will have no effect.

Reading from PE24103 using the MFR_IOUT_MAX command will return a 2-byte value describing the maximum rated output current for a single output from PE24103. The expected value from the PRD is 4.0A. The data from a MFR_IOUT_MAX command will be returned using a LINEAR11 data format as follows:

Exponent = -5 (5'b1_1011)

Mantissa = +4.0, which is expressed as sign_xxxxx_yyyyy where x are the integer bits and yyyyy are the fractional bits. For a value of +4.0 the value is

0_00100_00000

The combined 2-byte value returned by the MFR_IOUT_MAX command is expected to be 0xD880 (with the least significant byte supplied first). This value will come from a (private) NVM location and will not be controllable by the user.

The MFR_IOUT_MAX command is not PAGED and will not be affected by either the STORE_USER_ALL or RESTORE_USER_ALL commands.

The MFR_IOUT_MAX command has two data bytes as argument and uses the Read WORD protocol as shown in Figure 10.

For silicon versions of PE24103 after ES2, attempting to write using the (read only) MFR_IOUT_MAX command will trigger a fault bit in the STATUS_CML register.

5.1.55 IC_DEVICE_ID (Command Code ADh)

The IC_DEVICE_ID command is read only and supplies a single byte of data. The data value comes from NVM. The 8-bits of data which are supplied in response to the IC_DEVICE_ID command are defined for PE24103 in Table 41.

Table 41 PE24103 IC_DEVICE_ID Command Response Byte

Bits	Description	PE24103 Value	Meaning
7:0	Manufacturer chosen IC device ID value	0x0B	We maintain a rolling device number for each major die type in the power business. The value 0x0B has been sequentially assigned to PE24103.

The value returned by the IC_DEVICE_ID command is not affected by the PAGE number and cannot be changed by the user so is not affected by the STORE_USER_ALL or RESTORE_USER_ALL commands.

The IC_DEVICE_ID command uses a BLOCK Read protocol with a single data byte. A generic BLOCK Read packet is shown in Figure 13.

5.1.56 MFR_SPECIFIC_PGOOD (Command Code C4h)

The MFR_SPECIFIC_PGOOD command is unique to PE24103. PE24103 supports 4 switching, power outputs, but only a single "Power Good" status pin. The purpose of the MFR_SPECIFIC_PGOOD command is to allow a user to exercise some control over what "Power Good" should mean in relation to the output voltage at 1, 2, 3 or all 4 of the PE24103 power outputs.

The MFR_SPECIFIC_PGOOD command is read/write and accepts a 2-byte data value. The data bits for the 2-byte data word are defined in Table 42.

Table 42 PE24103 MFR_SPECIFIC_PGOOD Command Data Bytes

Bits	Description	PE24103 Value	Meaning
15:12	FALLING VOUT Logical OR enable for outputs 4 to 1		FALLING VOUT Logical OR enable for outputs 4 to 1
11:8	FALLING VOUT Logical AND enable for outputs 4 to 1		FALLING VOUT Logical AND enable for outputs 4 to 1
7:4	RISING VOUT Logical OR enable for outputs 4 to 1		RISING VOUT Logical OR enable for outputs 4 to 1
3:0	RISING VOUT Logical AND enable for outputs 4 to 1		RISING VOUT Logical AND enable for outputs 4 to 1

The MFR_SPECIFIC_PGOOD data value allows the (internal) power good status of each of the four individual outputs of PE24103 to be logically combined to create the final power good output pin voltage. The good (or not) state of each individual power output depends on the VOUT value compared to the POWER_GOOD_ON and POWER_GOOD_OFF values (which are PAGED).

Some example configurations are described below:

- All 4 outputs should contribute to power good state and PE24103 should only set the "good" pin state when all 4 output have exceeded their respective POWER_GOOD_ON and should flag a "bad" state as soon as any one of the outputs dips below its respective POWER_GOOD_OFF level. In this case, this is an AND function for both rising and falling and involves all 4 outputs. The MFR_SPECIFIC_PGOOD value should be set to 0x0F0F.
- If the same scenario is used again, but output #2 is not enabled for some reason, then we can make the "good" state ignore output #2 by not setting any of the associated bits of MFR_SPECIFIC_PGOOD for output #2: MFR_SPECIFIC_PGOOD = 16'b0000_1101_0000_1101.
- If two of the output rails are being combined together in some way at the board level, then it might make sense for the required "good" state to be an AND function for rising VOUT and an OR for falling VOUT. In that case (and assuming outputs 3 and 4 are used), an appropriate value for MFR_SPECIFIC_PGOOD might be 16'b1100_0000_0000_1100. Outputs 1 and 2 would not contribute to the "good" pin state using this example value.

Note that the effect of the MFR_SPECIFIC_PGOOD value is combinatorial within PE24103. One implication of this is that if some outputs are already enabled, changing the MFR_SPECIFIC_PGOOD value could cause the power good output pin state to switch depending on the power output states and the old, and new, values for MFR_SPECIFIC_PGOOD.

The value written using the MFR_SPECIFIC_PGOOD command is not PAGED. The value is stored in NVM using the STORE_USER_ALL command and can be restored from NVM using the RESTORE_USER_ALL command.

The MFR_SPECIFIC_PGOOD command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.57 MFR_SPECIFIC_MATRIX (Command Code C5h)

The MFR_SPECIFIC_MATRIX command provides one method for the user to configure the PE24103 for multi-channel operation. There are 5 pre-defined configurations which PE24103 can support to combine outputs for higher power output supplies. Figure 14 shows the pre-defined configurations that PE24103 can support.

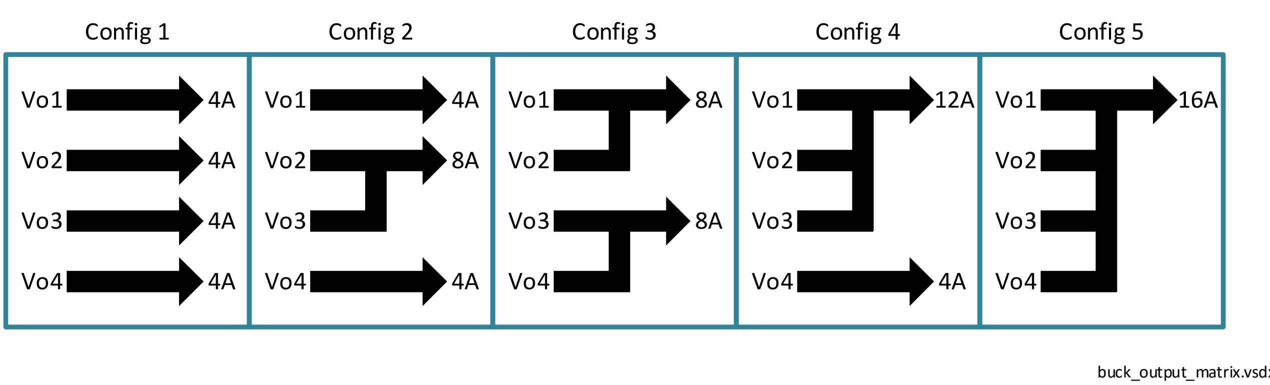


Figure 14 PE24103 Switching Output Configurations

It is also possible to use the Rtrim chip pins to define a matrix configuration. The details of this are not described here. What is important to understand is that if the Rtrim pins are being used to configure the matrix value, then the value used for the MFR_SPECIFIC_MATRIX command should be consistent with the matrix value defined by the Rtrim pins. A conflict between the two methods will result in PE24103 not enabling the power outputs until the conflict has been resolved.

The MFR_SPECIFIC_MATRIX command is a read/write command with a 2-byte data value which is described in Table 44. The bit values define how each of the switching outputs is connected.

Table 43 PE24103 MFR_SPECIFIC_MATRIX Command Data Values

Bits	Description	PE24103 Value	Meaning
15:12	Not used		
11	SW4 Valid		When set to '1', the SW4 bits are considered as valid and will affect chip operation. When set to '0', the SW4 bits are not used and the value is not important.
10	SW3 Valid		When set to '1', the SW3 bits are considered as valid and will affect chip operation. When set to '0', the SW3 bits are not used and the value is not important.
9	SW2 Valid		When set to '1', the SW2 bits are considered as valid and will affect chip operation. When set to '0', the SW2 bits are not used and the value is not important.
8	SW1 Valid		When set to '1', the SW1 bits are considered as valid and will affect chip operation. When set to '0', the SW1 bits are not used and the value is not important.
7:6	SW4		When flagged as valid, these bits indicate whether switching output 4 is operating independently or is shared at the board level and is controlled by another channel/PAGE.
5:4	SW3		When flagged as valid, these bits indicate whether switching output 3 is operating independently or is shared at the board level and is controlled by another channel/PAGE.
3:2	SW2		When flagged as valid, these bits indicate whether switching output 2 is operating independently or is shared at the board level and is controlled by another channel/PAGE.
1:0	SW1		When flagged as valid, these bits indicate whether switching output 1 is operating independently or is shared at the board level and is controlled by another channel/PAGE.

Table 44 expands on the values used for the various configurations that PE24103 is expecting to support when 1, or more, of the outputs is combined (at the PCB level) to create a single, higher capacity output.

Table 44 PE24103 Configurations & the MFR_SPECIFIC_MATRIX Data Byte

Register Bits	[7:6] "SW4"	[5:4] "SW3"	[3:2] "SW2"	[1:0] "SW1"
Description	Defines the output that VOUT4 is connected to (externally). 3'b11 – VOUT4 is connected to VOUT4 2'b10 – VOUT4 is connected to VOUT3 2'b01 – VOUT4 is connected to VOUT2 2'b00 – VOUT4 is connected to VOUT1	Defines the output that VOUT3 is connected to (externally). 3'b11 – VOUT3 is connected to VOUT4 2'b10 – VOUT3 is connected to VOUT3 2'b01 – VOUT3 is connected to VOUT2 2'b00 – VOUT3 is connected to VOUT1	Defines the output that VOUT2 is connected to (externally). 3'b11 – VOUT2 is connected to VOUT4 2'b10 – VOUT2 is connected to VOUT3 2'b01 – VOUT2 is connected to VOUT2 2'b00 – VOUT2 is connected to VOUT1	Defines the output that VOUT1 is connected to (externally). 3'b11 – VOUT1 is connected to VOUT4 2'b10 – VOUT1 is connected to VOUT3 2'b01 – VOUT1 is connected to VOUT2 2'b00 – VOUT1 is connected to VOUT1

Note that although Table 44 could imply that up to 256 possible combinations were allowed, in fact there are 5 pre-defined configurations and the valid values of the data byte for the MFR_SPECIFIC_MATRIX command for the configurations in Figure 14 are shown in Table 45. Values for the MFR_SPECIFIC_MATRIX command other than those shown in Table 45 could lead to inaccurate or unpredictable device operation.

Table 45 PE24103 Supported Configurations & Legal MFR_SPECIFIC_MATRIX Value

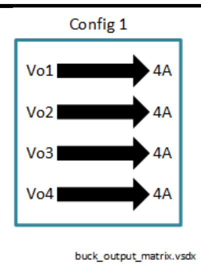
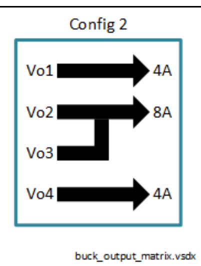
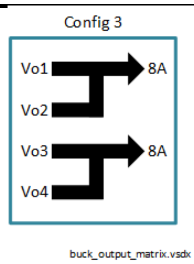
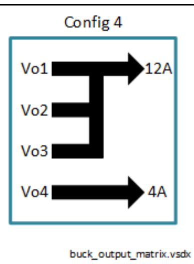
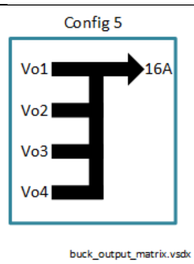
Config	Output Connection	Description	Legal MFR_SPECIFIC_MATRIX data byte value
1		Four independent outputs.	12'b1111_11_10_01_00 (12'hFE4)
2		Outputs 2 and 3 are combined, outputs 1 and 4 remain independent.	12'b1111_11_01_01_00 (12'hFD4)

Table 45 PE24103 Supported Configurations & Legal MFR_SPECIFIC_MATRIX Value

Config	Output Connection	Description	Legal MFR_SPECIFIC_MATRIX data byte value
3		Outputs 1 and 2 are combined; outputs 3 and 4 are also combined but remain independent of outputs 1 & 2.	12'b1111_10_10_00_00 (12'hFA0)
4		Outputs 1, 2 and 3 are combined, output 4 remains independent.	12'b1111_11_00_00_00 (12'hFC0)
5		All outputs are combined.	12'b1111_00_00_00_00 (12'hF00)

The MFR_SPECIFIC_MATRIX command is not PAGED. The value for the MFR_SPECIFIC_MATRIX is stored to on-chip non-volatile memory using the STORE_USER_ALL command. The data value is restored from NVM using the RESTORE_USER_ALL command.

The MFR_SPECIFIC_MATRIX command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.58 MFR_SPECIFIC_IOUTFILT (Command Code C6h)

The MFR_SPECIFIC_IOUTFILT command allows the user to configure low pass filtering which can be applied to measurements of the IOUT current level.

The MFR_SPECIFIC_IOUTFILT command is not PAGED. A single value controls whether averaging is used, or not, for all four of the IOUT values.

The value for IOUT which is read using the READ_IOUT command is the filtered version of the IOUT data.

The filtered IOUT data value is also used for comparisons with the IOUT_OC_FAULT_LIMIT & IOUT_UC_FAULT_LIMIT values.

The MFR_SPECIFIC_IOUTFILT command is read/write and accepts a 2-byte data value. The data bits for the 2-byte data word are defined in Table 46.

Table 46 PE24103 MFR_SPECIFIC_IOUTFILT Command Data Byte

Bits	Description	PE24103 Value	Meaning
15:11	Not used		
10:0	Unsigned numerical value for the inverse of the filter time constant		The numerical value represents a fractional value (n/2048) for the scaling factor applied to the delta from one IOUT measurement to the next

The IOUT digital low pass filtering uses a simple approach commonly known as exponential smoothing:

https://en.wikipedia.org/wiki/Exponential_smoothing

In brief, the delta of the latest IOUT measurement from the smoothed value is calculated. The delta value is then scaled (by the MFR_SPECIFIC_IOUTFILT value divided by 2048) and then added to the smoothed value.

There is a special value for MFR_SPECIFIC_IOUTFILT = 0 which effectively disables the IOUT filter and, in this case, the READ_IOUT value always reflects the most recent IOUT measurement. Similarly the IOUT_OC/UC_FAULT_LIMIT value is compared with the latest IOUT measurement without any filtering applied when MFR_SPECIFIC_IOUTFILT=0.

The effect of the MFR_SPECIFIC_IOUTFILT value on the IOUT results for a nominal step function are shown in Figure 15. The figure is based on one IOUT result being taken every 4ms (FIXME, this should be a datasheet parameter). The figure also shows MFR_SPECIFIC_IOUTFILT set to powers of 2 values (512, 256 etc.), but this is not required. The only constraint value for MFR_SPECIFIC_IOUTFILT is that it should be an unsigned binary value between 0 and 2047.

It is important to understand that the MFR_SPECIFIC_IOUTFILT value is not directly a time constant. The time between consecutive IOUT ADC results may not be constant. The MFR_SPECIFIC_IOUTFILT can only be truly considered as a time constant if the time from one conversion result to the next is known and fixed. Nevertheless the approximation to low pass filtering is clear in Figure 15. The figure also shows one constant value which allows the time constant to be assessed assuming one conversion every 4ms.

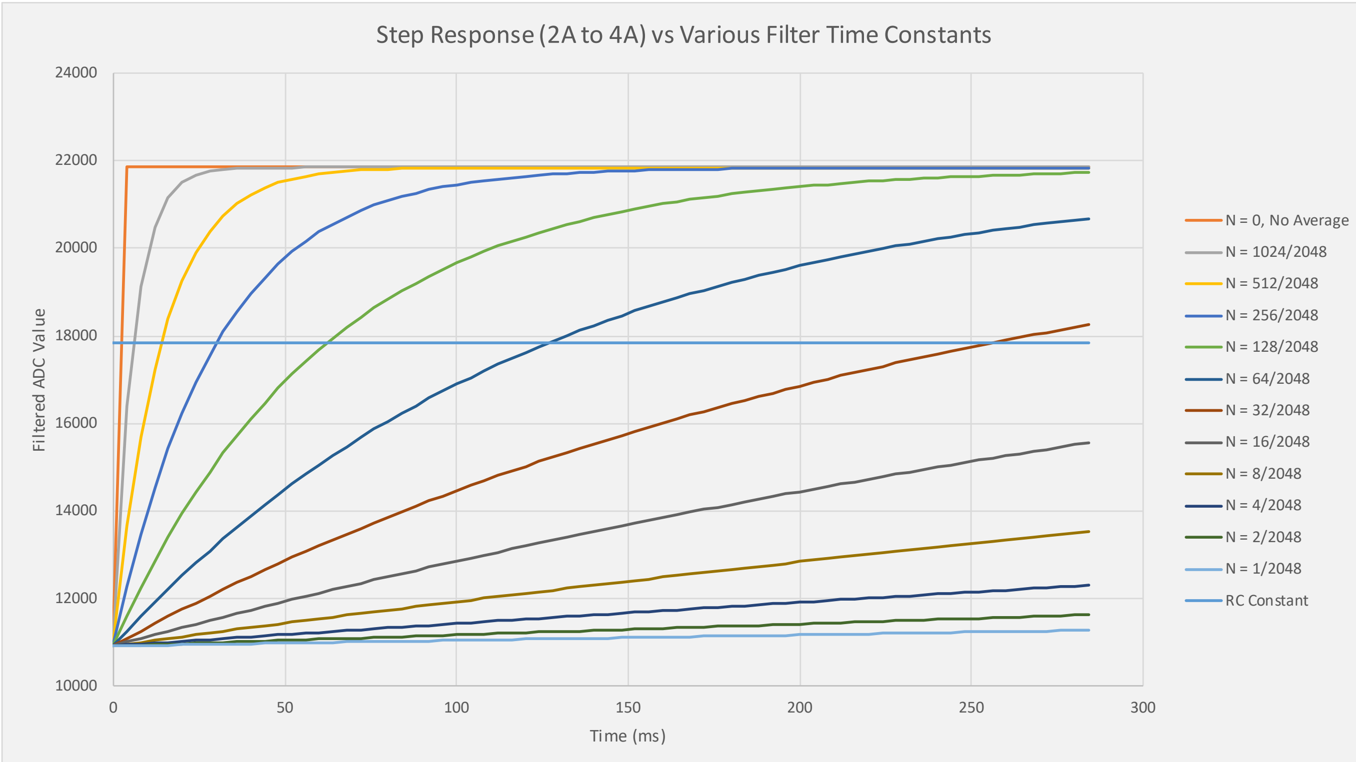


Figure 15 PE24103 Sketch of the Effect of the MFR_SPECIFIC_IOUTFILT Value

The MFR_SPECIFIC_IOUTFILT command is not PAGED. The value for MFR_SPECIFIC_IOUTFILT is stored to on-chip non-volatile memory using the STORE_USER_ALL command. The data value is restored from NVM using the RESTORE_USER_ALL command.

The MFR_SPECIFIC_IOUTFILT command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.59 MFR_SPECIFIC_OP_BEH (Command Code C7h)

The MFR_SPECIFIC_OP_BEH command allows the user to configure various aspects of the VOUT output behaviour which are not already covered by existing PMBus commands.

The read/write MFR_SPECIFIC_OP_BEH command is PAGED and accepts a single byte data value.

Table 47 PE24103 MFR_SPECIFIC_OP_BEH Command Data Byte

Bits	Bit Name	Meaning

Table 47 PE24103 MFR_SPECIFIC_OP_BEH Command Data Byte

Bits	Bit Name	Meaning
4	PRBS	When the PRBS bit is set to '1', the clock for the power switches driving VOUTx (depending on the PAGE value) will have some dither added using a Pseudo-Random Binary Sequence. The degree of dither depends on the clock frequency. The dither on the clocks for each of the four chip outputs operates independently. The goal of the dither is to implement limited spectrum spreading around the core switching rate.
3	SOFT	When the GLOBAL bit is set to '1' and a fault condition causes this BUCK to be turned off, the other three BUCK outputs will also be turned off at the same time. If the SOFT bit is set to '1', the other (unfaulted) channels will go through a controlled (PMBus configured) shutdown. When SOFT = '0', and GLOBAL = '1', then a hard fault on this BUCK will also cause immediate shutdown of the other BUCKs. The SOFT bit has no function if the GLOBAL bit is set to '0'.
2	GLOBAL	When the GLOBAL bit is set to '1' and a fault condition causes this BUCK to be turned off, the other three BUCK outputs will also be turned off at the same time. This is to allow for the possibility that the BUCK outputs are linked in some way, to create multiple supplies to a device for example. When GLOBAL='0', a fault on this BUCK will not affect the other BUCK outputs.
1	LATCH	When the LATCH bit is set to '1' and a fault condition causes the BUCK to be turned off, the output will remain OFF (even when the fault condition clears) until the OPERATION command has been used to turn the BUCK off and then ON again. Setting the EN pin low and then high again will also re-enable the BUCK output. When LATCH = '0', PE24103 will allow this BUCK to go into a hiccup mode where the output will attempt to re-enable periodically to see if the fault condition is still present or not.
0	FLOAT	When the FLOAT bit is set to '1' and a fault condition causes the BUCK to be turned off, the output will not be driven to 0V, instead the BUCK will simply stop supplying power to the load. When FLOAT = '0', the output will be ramped to 0V using the times in the TOFF_DLY and TOFF_FALL PMBus settings.

The MFR_SPECIFIC_OP_BEH command is PAGED. The value for MFR_SPECIFIC_OP_BEH is stored to on-chip non-volatile memory using the STORE_USER_ALL command. The data value is restored from NVM using the RESTORE_USER_ALL command.

The MFR_SPECIFIC_OP_BEH command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.1.60 MFR_SPECIFIC_MASK (Command Code C8h)

The MFR_SPECIFIC_MASK value operates much like the standard status register mask registers and allows the user to decide which status bits are allowed to trigger the SMBALERT pin. The reason that this data value is manufacturer specific is that it allows the bits in the STATUS_WORD to be globally masked or not. For example, if the VOUT bit of MFR_SPECIFIC_MASK is set to ‘1’, no VOUT related faults will cause assertion of the SMBALERT signal regardless of the mask bits for the individual (PAGED) VOUT Status Mask register.

The MFR_SPECIFIC_MASK command bits also allow masking of bits in the STATUS_WORD value which are not otherwise maskable (for example the Power Good status bit).

Table 48 shows the data values which can be set using the MFR_SPECIFIC_MASK command. The bits have a one-to-one mapping with the bits defined in the standard PMBus STATUS_WORD (and STATUS_BYTE) commands.

Table 48 PE24103 MFR_SPECIFIC_MASK Command Data Byte

Bits	Bit Name	Meaning
15	VOUT	Setting this bit to '1' will prevent any VOUT faults in the PAGED STATUS_VOUT registers from causing the SMBALERT pin to be asserted.
14	IOUT	Setting this bit to '1' will prevent any IOUT faults in the PAGED STATUS_IOUT registers from causing the SMBALERT pin to be asserted.
13	INPUT	Setting this bit to '1' will prevent any VIN/IIN faults in the STATUS_INPUT register from causing the SMBALERT pin to be asserted.
12	MFRSP	Setting this bit to '1' will prevent any faults in the STATUS_MFR_SPECIFIC register from causing the SMBALERT pin to be asserted.
11	PG	Setting this bit to '1' will prevent the PE24103 Power Good status signal from causing the SMBALERT pin to be asserted.
10	FANS	Not used in PE24103.
9	OTHER	Setting this bit to '1' will prevent any fault flags in the STATUS_OTHER register from causing the SMBALERT pin to be asserted.
8	UNKNOWN	Not used in PE24103.
7	BUSY	Setting this bit to '1' will prevent the BUSY fault flag from causing the SMBALERT pin to be asserted.
6	OFF	Setting this bit to '1' will prevent the fact that any VOUT output is not providing power to the load (including simply not being enabled by the user) from causing the SMBALERT pin to be asserted.
5	VOUTOV	Setting this bit to '1' will prevent any VOUT overvoltage faults in the PAGED STATUS_VOUT registers from causing the SMBALERT pin to be asserted.
4	IOUTOC	Setting this bit to '1' will prevent any IOUT overcurrent faults in the PAGED STATUS_IOUT registers from causing the SMBALERT pin to be asserted.
3	VINUV	Setting this bit to '1' will prevent the VIN undervoltage fault flag in the STATUS_INPUT register from causing the SMBALERT pin to be asserted.
2	TEMP	Setting this bit to '1' will prevent any temperature faults in the STATUS_TEMPERATURE register from causing the SMBALERT pin to be asserted.
1	CML	Setting this bit to '1' will prevent any fault flags in the STATUS_CML register from causing the SMBALERT pin to be asserted.
0	NOA	Setting this bit to '1' will prevent a "none of the above" fault flag from causing the SMBALERT pin to be asserted.

The MFR_SPECIFIC_MASK command is not PAGED. The value for the MFR_SPECIFIC_MASK is stored to on-chip non-volatile memory using the STORE_USER_ALL command. The data value is restored from NVM using the RESTORE_USER_ALL command.

The MFR_SPECIFIC_MASK command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.61 MFR_SPECIFIC_DCR_TC (Command Code C9h)

The MFR_SPECIFIC_DCR_TC command allows the user to add an estimated temperature coefficient correction for the DC resistance value of the external inductor which is used at each VOUTx pin of PE24103. There is one temperature coefficient for all 4 VOUT pins.

The value used for the MFR_SPECIFIC_DCR_TC command is a 16-bit, 2s complement value which provides the tempco value in parts per 2^20 (approximately 1 million). Typically the tempco would be quoted in parts per million relative to 25C. Using the quoted ppm value directly in this register will incur a small (5%) error. Alternatively the datasheet ppm value can be scaled by 2^20/10^6 to remove this error.

This command is optional and setting a tempco value of 0 will disable the function. When a non-zero value is set for the MFR_SPECIFIC_DCR_TC value then the PE24103 **on-chip DIE** temperature will be assumed to be the same as the temperature of the external inductor component and used to automatically adjust the measured IOUT value to compensate for the effect of temperature on the effective DC resistance of the inductor. The correlation between die temperature and the actual temperature of the inductor at the time of the IOUT measurement will be system, and board layout, dependent.

PE24103 will use the most recent die temperature to estimate the effect of the tempco on the external inductor DCR value and then adjust the reported IOUTx result to take account of the tempco as shown in the following equation.

$$IOUT_{reported} = \frac{IOUT_{measured}}{1 + \frac{TC * (Die\ Temp - 25C)}{2^{20}}}$$

Where:

TC = the value entered using the MFR_SPECIFIC_DCR_TC command

Die Temp = the most recent die temperature measurement (in degrees Celsius)

To give an example: assuming the quoted tempco was +3000ppm, then the value used for the MFR_SPECIFIC_DCR_TC would be +3000 (0x0BB8). For better accuracy, the tempco could be scaled by 2^20/10^6 (giving 3,146 or 0x0C4A). The tempco scheme assumes 25C as the reference temperature for the tempco. The on-chip tempco calculation will limit the maximum effect of the tempco to +/-50%.

With a MFR_SPECIFIC_DCR_TC value of +3,146, a die temperature of 50C and a measured IOUT value of 3.5A, the reported current value would be:

$$IOUT_{reported} = \frac{3.5}{1 + \frac{3146 * (50C - 25C)}{2^{20}}} = \frac{3.5}{1.075} = 3.26A$$

Table 49 PE24103 MFR_SPECIFIC_DCR_TC Command Data Byte

Bits	Bit Name	Meaning

Table 49 PE24103 MFR_SPECIFIC_DCR_TC Command Data Byte

Bits	Bit Name	Meaning
15:0	IOUT_DCR_TC	The value is entered as a 16-bit, 2s complement value. So a positive tempco of +3,900 would be entered as 16'h0F3C. A negative tempco can also be entered (although not expected), so a negative tempco of -4,500ppm could be entered as approximately 16'hEE6C (or more exactly in parts per 2^20 as 16'hED91). Setting a value of 16'h0000 will disable the on-chip temperature compensation calculation.

The MFR_SPECIFIC_DCR_TC command is not PAGED, the same value is used for all 4 inductors. The value for MFR_SPECIFIC_DCR_TC is stored to on-chip non-volatile memory using the STORE_USER_ALL command. The data value is restored from NVM using the RESTORE_USER_ALL command.

The MFR_SPECIFIC_DCR_TC command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.62 MFR_SPECIFIC_RSENSE_TC (Command Code CAh)

The MFR_SPECIFIC_RSENSE_TC command allows the user to add an estimated temperature coefficient correction for the DC resistance value of the external sense resistor which can (optionally) be fitted to allow measurement of the input current to PE24103.

The value used for the MFR_SPECIFIC_RSENSE_TC command is a 16-bit, 2s complement value which provides the tempco value in parts per 2^20 (approximately 1 million). Typically the tempco would be quoted in parts per million relative to 25C. Using the quoted ppm value directly in this register will incur a small (5%) error. Alternatively the datasheet ppm value can be scaled by 2^20/10^6 to remove this error.

This command is optional and setting a tempco value of 0 will disable the function. When a non-zero value is set for the MFR_SPECIFIC_RSENSE_TC value then the PE24103 **on-chip DIE** temperature will be assumed to be the same as the temperature of the external sense resistor component and used to automatically adjust the measured IIN value to compensate for the effect of temperature on the effective resistance of the sense resistor. The correlation between die temperature and the actual temperature of the sense resistor at the time of the IIN measurement will be system, and board layout, dependent.

PE24103 will use the most recent die temperature to estimate the effect of the tempco on the external sense resistor value and then adjust the reported IIN result to take account of the tempco as shown in the following equation.

$$IIN_{REPORTED} = \frac{IIN_{MEASURED}}{1 + \frac{TC * (Die\ Temp - 25C)}{2^{20}}}$$

Where:

TC = the value entered using the MFR_SPECIFIC_RSENSE_TC command

Die Temp = the most recent die temperature measurement (in degrees Celsius)

Table 50 PE24103 MFR_SPECIFIC_RSENSE_TC Command Data Byte

Bits	Bit Name	Meaning
15:0	IOUT_RSENSE_TC	The value is entered as a 16-bit, 2s complement value. So a positive tempco of +3,900 would be entered as 16'h0F3C. A negative tempco can also be entered (although not expected), so a negative tempco of -4,500ppm could be entered as approximately 16'hEE6C (or more exactly in parts per 2^20 as 16'hED91). Setting a value of 16'h0000 will disable the on-chip temperature compensation calculation.

The MFR_SPECIFIC_RSENSE_TC command is not PAGED. The value for MFR_SPECIFIC_RSENSE_TC is stored to on-chip non-volatile memory using the STORE_USER_ALL command. The data value is restored from NVM using the RESTORE_USER_ALL command.

The MFR_SPECIFIC_RSENSE_TC command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.63 MFR_IOUT_CAL_GAIN (Command Code CBh)

The MFR_IOUT_CAL_GAIN command serves the same purpose as the standard IOUT_CAL_GAIN command. It allows the user to specify the value of the external sense "resistor" (or equivalent series resistance of an inductor) which is being used for measuring current. The on-chip ADC measures a voltage and to convert the measured voltage back to a current, knowledge of the sense resistor value is required.

The standard PMBus IOUT_CAL_GAIN command allows definition of a sense resistance value in milliohms. The Manufacturer specific IOUT_CAL_GAIN command allows exactly the same value to be specified, but for convenience in the internal implementation, the units for the MFR_IOUT_CAL_GAIN are a ratio relative to 10 milli-ohms.

This register value provides the value of the DCR for the inductor used to sense IOUT current as a RATIO relative to a nominal value of 10mOhms. The expected range of DCR values is in the range of 3mOhms to 30mOhms. The ratio of the actual DCR value is expressed as an unsigned binary value with 3 integer bits and 13 fraction bits.

For example, for a 3mOhm DCR value, the ratio to be stored would be 10/3 = 3.333 (0x6AAB), for a 30mOhm DCR value, the ratio would be 10/30 = 0.33 (0x0AAB) and if the actual DCR value were 10.8mOhms, the ratio required in this register would be 10/10.8 = 0.926. The value of 0.926 would be stored as 0.926*2^13 or 0x1DA1. The 3 integer bits (15:13) would be 3'b000 (0) and the fraction bits would be 13'b1_1101_1010_0001 (0.926). If the IOUT_CAL_GAIN command is unused a default value of 1.0 can be stored as 0x2000.

It is important to note that the value set using the MFR_IOUT_CAL_GAIN command will have a direct impact on the accuracy of any measured current values returned by the READ_IOUT command.

The MFR_IOUT_CAL_GAIN command is PAGED.

The MFR_IOUT_CAL_GAIN command is read/write and accepts a 2-byte data value. The data bits for the 2-byte data word are defined in Table 51.

Table 51 PE24103 MFR_IOUT_CAL_GAIN Command Data Byte

Bits	Description	PE24103 Value	Meaning
15:0	Ratio relative to 10mOhm		Number is stored as a 3.13 unsigned value allowing values from 0 to 7.999 to be stored corresponding to inductor equivalent DC resistance values of 1.25mOhm and above to be defined. Note that the expected range of values for the DCR value is 3 to 30mOhms.

The MFR_IOUT_CAL_GAIN command is PAGED. The value for MFR_IOUT_CAL_GAIN is stored to on-chip non-volatile memory using the STORE_USER_ALL command. The data value is restored from NVM using the RESTORE_USER_ALL command.

- **This command is not supported on PE24103 ES1/ES2 silicon.**

The MFR_IOUT_CAL_GAIN command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.64 MFR_IIN_CAL_GAIN (Command Code CCh)

The MFR_IIN_CAL_GAIN command serves the same purpose as the standard IOUT_CAL_GAIN command except that it is applied to sensing the current drawn by PE24103 from the VIN supply. It allows the user to specify the value of the external sense resistor which is being used for measuring current. The on-chip ADC measures a voltage and to convert the measured voltage back to a current, knowledge of the sense resistor value is required.

The standard PMBus IOUT_CAL_GAIN command allows definition of a sense resistance value in milliohms. The Manufacturer specific IIN_CAL_GAIN command allows exactly the same value to be specified, but for convenience in the internal implementation, the units for the MFR_IIN_CAL_GAIN are a ratio relative to 10 milli-ohms.

This register value provides the value of the external sense resistor used to sense IIN current as a RATIO relative to a nominal value of 10mOhms. The expected range of sense resistor values is in the range of 3mOhms to 30mOhms. The ratio of the actual sense resistor value is expressed as an unsigned binary value with 3 integer bits and 13 fraction bits.

For example, for a 3mOhm sense resistor value, the ratio to be stored would be 10/3 = 3.333 (0x6AAB), for a 30mOhm sense resistor value, the ratio would be 10/30 = 0.33 (0x0AAB) and if the actual sense resistor value were 10.8mOhms, the ratio required in this register would be 10/10.8 = 0.926. The value of 0.926 would be stored as 0.926*2^13 or 0x1DA1. The 3 integer bits (15:13) would be 3'b000 (0) and the fraction bits would be

13'b1_1101_1010_0001 (0.926). If the IIN_CAL_GAIN command is unused a default value of 1.0 can be stored as 0x2000.

It is important to note that the value set using the MFR_IIN_CAL_GAIN command will have a direct impact on the accuracy of any measured current values returned by the READ_IIN command.

The MFR_IIN_CAL_GAIN command is read/write and accepts a 2-byte data value. The data bits for the 2-byte data word are defined in Table 52.

Table 52 PE24103 MFR_IIN_CAL_GAIN Command Data Byte

Bits	Description	PE24103 Value	Meaning
15:0	Ratio relative to 10mOhm		Number is stored as a 3.13 unsigned value allowing values from 0 to 7.999 to be stored corresponding to sense resistor values of 1.25mOhm and above to be defined. Note that the expected range of values for the sense resistor is 3 to 30mOhms.

The MFR_IIN_CAL_GAIN command is not PAGED. The value for MFR_IIN_CAL_GAIN is stored to on-chip non-volatile memory using the STORE_USER_ALL command. The data value is restored from NVM at power-up and by using the RESTORE_USER_ALL command.

- **This command is not supported on PE24103 ES1/ES2 silicon.**

The MFR_IIN_CAL_GAIN command has two data bytes as argument and uses the Write WORD and Read WORD protocols as shown in Figure 9 and Figure 10.

5.1.65 MFR_CFG_PMBUS (Command Code CDh)

The MFR_CFG_PMBUS command allows the user to configure various aspects of the PMBus serial interface behaviour in a single command. The MFR_CFG_PMBUS command is read/write for PE24103 and requires a single byte of write data. Table 53 shows how the data bit is defined.

Table 53 PE24103 MFR_CFG_PMBUS Command Data Byte

Bits	Description	PE24103 Value	Meaning
7	NO_TO (“No Timeout”)		This bit can be used to DISABLE the SMBus tTIMEOUT function. The SMBus standard describes this function as recommended, but optional, for bus slaves which cannot hold SCL low (like PE24103). Setting the NO_TO bit to '1' will disable the internal timeout circuit which will reset the PMBus hardware which interfaces with the external bus after a timeout period of 25-35ms with the SCL pin is held low.
6:0	Device Address		These register bits will set the PMBus device address value which PE214103 recognizes and responds to when the STOP condition for the MFR_CFG_PMBUS command occurs. The next command will only respond to the new address value.

Note that the MFR_CFG_PMBUS command allows all 7-bits of the PE24103 PMBus Device Address value to be set to a user selected value. However, it should be noted, that both PMBus and SMBus reserve several device address values for special purposes, so not all 128 possible device address values may be legal. In addition to those device address values which are reserved by 3rd party standards, the device address value 7’b0111_000 is reserved by the PE24103 device manufacturer and should not be used.

The MFR_CFG_PMBUS command is not PAGED. The value for MFR_CFG_PMBUS is stored to on-chip non-volatile memory using the STORE_USER_ALL command. The data value is restored from NVM at power-up and by using the RESTORE_USER_ALL command.

- **This command is not supported on PE24103 ES1/ES2 silicon.**

The MFR_CFG_PMBUS command has a single byte as argument and uses the Write BYTE and Read BYTE protocols as shown in Figure 2 and Figure 3.

5.2 Unsupported Commands

Table 54 is a summary table for the PMBus commands which PE24103 does not support. The main exclusions are:

- The FAULT Response registers. ELSA supported hiccup mode (which is likely the hardest option), so we could support the FAULT response option registers (even if only as read-only locations to confirm how PE24103 will respond?).
- RAMP Rate controls. ELSA supported a limited range of soft-start and soft-shut ramp rates under user control. Some of this is included in the TON_RISE and TOFF_FALL commands although existing hardware had only fixed values, not a widely variable range.

Table 54 PMBus Commands Not Supported by PE24103

Command Code	Command Name
04h	PHASE
07h	ZONE_CONFIG
08h	ZONE_ACTIVE
10h	WRITE_PROTECT
11h	STORE_DEFAULT_ALL
12h	RESTORE_DEFAULT_ALL
13h	STORE_DEFAULT_CODE
14h	RESTORE_DEFAULT_CODE
17h	STORE_USER_CODE
18h	RESTORE_USER_CODE
1Ah	QUERY
22h	VOUT_TRIM
23h	VOUT_CAL_OFFSET
24h	VOUT_MAX
28h	VOUT_DROOP
29h	VOUT_SCALE_LOOP
2Ah	VOUT_SCALE_MONITOR
2Bh	VOUT_MIN
30h	COEFFICIENTS
31h	POUT_MAX
32h	MAX_DUTY
34h	POWER_MODE
37h	INTERLEAVE
38h	IOUT_CAL_GAIN
39h	IOUT_CAL_OFFSET
3Ah	FAN_CONFIG_1_2
3Bh	FAN_COMMAND_1
3Ch	FAN_COMMAND_2
3Dh	FAN_CONFIG_3_4
3Eh	FAN_COMMAND_3
3Fh	FAN_COMMAND_4
41h	VOUT_OV_FAULT_RESPONSE
42h	VOUT_OV_WARN_LIMIT
43h	VOUT_UV_WARN_LIMIT
45h	VOUT_UV_FAULT_RESPONSE
47h	IOUT_OC_FAULT_RESPONSE
48h	IOUT_OC_LV_FAULT_LIMIT
49h	IOUT_OC_LV_FAULT_RESPONSE
4Ah	IOUT_OC_WARN_LIMIT
4Ch	IOUT_UC_FAULT_RESPONSE
50h	OT_FAULT_RESPONSE
52h	UT_WARN_LIMIT
53h	UT_FAULT_LIMIT
54h	UT_FAULT_RESPONSE
56h	VIN_OV_FAULT_RESPONSE
57h	VIN_OV_WARN_LIMIT

Table 54 PMBus Commands Not Supported by PE24103

Command Code	Command Name
58h	VIN_UV_WARN_LIMIT
5Ah	VIN_UV_FAULT_RESPONSE
5Ch	IIN_OC_FAULT_RESPONSE
5Dh	IIN_OC_WARN_LIMIT
62h	TON_MAX_FAULT_LIMIT
63h	TON_MAX_FAULT_RESPONSE
66h	TOFF_MAX_WARN_LIMIT
68h	POUT_OP_FAULT_LIMIT
69h	POUT_OP_FAULT_RESPONSE
6Ah	POUT_OP_WARN_LIMIT
6Bh	PIN_OP_WARN_LIMIT
81h	STATUS_FANS_1_2
82h	STATUS_FANS_3_4
83h	READ_KWH_IN
84h	READ_KWH_OUT
85h	READ_KWH_CONFIG
86h	READ_EIN
87h	READ_EOUT
8Ah	READ_VCAP
8Eh	READ_TEMPERATURE_2
8Fh	READ_TEMPERATURE_3
90h	READ_FAN_SPEED_1
91h	READ_FAN_SPEED_2
92h	READ_FAN_SPEED_3
93h	READ_FAN_SPEED_4
94h	READ_DUTY_CYCLE
95h	READ_FREQUENCY
96h	READ_POUT
97h	READ_PIN
9Ah	MFR_MODEL
9Bh	MFR_REVISION
9Ch	MFR_LOCATION
9Dh	MFR_DATE
9Eh	MFR_SERIAL
9Fh	APP_PROFILE_SUPPORT
A3h	MFR_PIN_MAX
A7h	MFR_POUT_MAX
A8h	MFR_TAMBIENT_MAX
A9h	MFR_TAMBIENT_MIN
AAh	MFR_EFFICIENCY_LL
ABh	MFR_EFFICIENCY_HL
ACh	MFR_PIN_ACCURACY
AEh	MFR_IC_DEVICE_REV
B0h	USER_DATA_00
B1h	USER_DATA_01
B2h	USER_DATA_02

Table 54 PMBus Commands Not Supported by PE24103

Command Code	Command Name
B3h	USER_DATA_03
B4h	USER_DATA_04
B5h	USER_DATA_05
B6h	USER_DATA_06
B7h	USER_DATA_07
B8h	USER_DATA_08
B9h	USER_DATA_09
BAh	USER_DATA_10
BBh	USER_DATA_11
BCh	USER_DATA_12
BDh	USER_DATA_13
BEh	USER_DATA_14
BFh	USER_DATA_15
C0h	MFR_MAX_TEMP_1
C1h	MFR_MAX_TEMP_2
C2h	MFR_MAX_TEMP_3
CEh to FDh	MFR_SPECIFIC_xx
FEh	MFR_SPECIFIC_COMMAND EXT
FFh	PMBUS_COMMAND_EXT

6 CHANGE HISTORY

Table 55 Document Change History

Date	Doc Version	Author	Description of Changes
29/Nov/19	0.1	DW	Original draft for discussion
13/Jan/20	0.1	DW	In the face-to-face meeting in Burlington 3rd-5th Dec 2019, we agreed to change support for a handful of extra commands: PAGE_PLUS_WRITE (0x05) is removed PAGE_PLUS_READ (0x06) is removed STORE_DEFAULT_ALL (0x11) is removed RESTORE_DEFAULT_ALL (0x12) is removed VOUT_TRANSITION_RATE (0x27) is added OT_WARN_LIMIT (0x51) is removed Update the various tables and section headings to match.
13/Jan/20	0.2	DW	Update headers/footers and file version.
13/Jan/20	0.2	DW	Start to fill in the sections which had only been place holders until now. This is a major update of the content.
7/Feb/20	0.3	DW	Update headers/footers and file version.
7/Feb/20	0.3	DW	Fix PE24013 used in many figure titles and replace with the correct PE24103.
7/Feb/20	0.3	DW	Move the PAGE_PLUS_READ and PAGE_PLUS_WRITE from the unsupported commands table to the supported commands. Add section headings for the description.
7/Feb/20	0.3	DW	Add a new Manufacturer Specific Command for IOUT Filtering. Add a suitable section header.
10/Feb/20	0.3	DW	Add a new table to the STORE_USER_ALL command summarizing the data/commands which are affected by the STORE_USER_ALL and RESTORE_USER_ALL commands.
11/Feb/20	0.3	DW	Add the STATUS_OTHER command (although we will only support one bit). Amend the figure for the nested hierarchy of status registers to match.
11/Feb/20	0.3	DW	Add MFR_ID and IC_DEVICE_ID to the list of supported commands. These are in Olimpiu's list and can be directly NVM values with small gate count impact.
11/Feb/20	0.4	DW	Update headers/footers and file version.
26/Feb/20	0.4	DW	Correct a typo: command 0x5B (IIN_OC_FAULT_LIMIT) is supported, so remove it from the table of Unsupported commands. Add 0xAE (IC_DEVICE_REV) to the table of Unsupported commands.
26/Feb/20	0.4	DW	Update all the figures for Configuration 2. It is now clear that when we combine 2 outputs (and have 2 independent outputs) we will combine outputs 2 and 3 (when numbered 1 to 4) rather than outputs 1 and 2.
26/Feb/20	0.4	DW	Update the description of the PAGE numbers. To support outputs 2/3 together as well as the 1/2 and 3/4 combinations, we need one extra virtual PAGE number. This affects text and tables in quite a few places.
26/Feb/20	0.4	DW	Replace all the thumbnail pictures in the tables with embedded .png pictures instead of linked visio files. This seems to make the file much quicker to open and save which makes it feel more stable too….
26/Feb/20	0.5	DW	Update headers/footers and file version.
3/Mar/20	0.5	DW	Remove the FILT_RST bit from the MFR_SPECIFIC_IOUTFILT register definition. We can make the READ_IOUT locations writable, so the user can reset values that way if they wish when changing the filter characteristics.
3/Mar/20	0.5	DW	Big update of the VOUT_MODE command. In a meeting 2/Mar/20 it was decided that we wanted to support the RELATIVE mode for VOUT related commands as well as the absolute mode. This is also wanted to be manageable on a per-output basis. To achieve this, VOUT_MODE[7] is now a writable bit (previously read only). In addition the VOUT_MODE command is PAGED (4 copies) although it is only bit[7] which can change. Finally we need to update the description of the VOUT_MARGIN_HIGH, VOUT_MARGIN_LOW, VOUT_OV_FAULT_LIMIT, VOUT_UV_FAULT_LIMIT, POWER_GOOD_ON and POWER_GOOD_OFF commands to indicate that their data values may all be relative.
3/Mar/20	0.5	DW	We will have at least one bit of Manufacturer specific STATUS info, so add in the STATUS_MFR_SPECIFIC command (and remove it from the list of unsupported commands).
16/Apr/20	0.5	DW	Edit the text for OPERATION[3:2] to remove some text imported from the PMBus standard which does not apply to PE24103.

Table 55 Document Change History

Date	Doc Version	Author	Description of Changes
01/Jun/20	0.5	DW	We need to add manufacturer specific values which serve the same function as the IOUT_CAL_GAIN command. Instead of storing a value in Ohms (as IOUT_CAL_GAIN does), we want to store a value in 1/Ohms (purely to help with on-chip computing, to avoid a divide operation). We also need an MFR_IIN_CAL_GAIN value for 1/Rsense at the Vin pin. For some reason there does not appear to be an IIN_CAL_GAIN command in the documentation.
11/Jun/20	0.5	DW	Amend the description of the CAPABILITY command. We are planning to support 1MHz PMBus operation although the PRD only requires 400kHz (in case we hit some issues at 1MHz).
11/Jun/20	0.5	DW	We have decided to support IOUT undercurrent faults. We need the command to set the fault limit plus associated fault flags. The new text is very similar to the IOUT overcurrent text.
12/Jun/20	0.6	DW	Update headers/footers and file version.
18/Nov/20	0.7	DW	Update headers/footers and file version.
	0.7	DW	Update the text for the CLEAR_FAULTS command throughout the document. This command had previously been described as not being PAGED, but in section 10.3 of Part II of the spec (revision 1.3.1), the CLEAR_FAULTS command is clearly described as being PAGED.
	0.7	DW	Update the text for the VOUT_MODE command throughout the document. This command had previously been described as not being PAGED, but we have decided during the course of the design work that it is PAGED.
	0.7	DW	Update the descriptions of the VOUT_MARGIN_HIGH, VOUT_MARGIN_LOW commands. In the final, gate level, implementation we decided to only store 10-bits of data for these values which limits both the max value and the resolution. Section 5.4.2 of Part II of the spec (revision 1.3.1) specifically notes that the received value may be stored with fewer bits than the data format allows, and that reading a value which does not completely match the input value due to bits which are not stored is not considered an error.
	0.7	DW	The STATUS_BYTE and STATUS_WORD commands are PAGED (although some bits are common). Update the text to match.
	0.7	DW	Add descriptions for the MANUFACTURER SPECIFIC OP_BEH, DCR_TC & RSENSE_TC PMBus commands.
	0.7	DW	Update the command codes for MFR_IOUT_CAL_GAIN and MFR_IIN_CAL_GAIN. These got missed from RTL 1.0 http://jira.psemi.com/browse/PMI-68 and so the old command codes in the doc were reused.
	0.7	DW	The MFR_SPECIFIC_MATRIX command was updated from a single, to a 2-byte command during the course of the design. Update the doc where needed.
	0.7	DW	Big update to the documentation for PAGEs and PAGING throughout the document. Previously we supported specific pages for specific combinations of outputs, but in the final implementation this seemed very complex for the user and so we retreated to a simple 4 PAGE structure with each PAGE directly tied to a specific VOUTx. This means we do NOT sum IOUT values, or expect increased values for IOUT_OC when 2, or more, outputs are tied together. All IOUT values are expected to be 0-4A, will be read on that scale and should have over/under current levels set relative to a nominal MAX of 4A per (single) output.
	0.7	DW	Update the description of the FREQUENCY_SWITCH command. We only expect certain values to be written.
	0.7	DW	Update the description of the MFR_SPECIFIC_IOUTFILT command. It is no longer PAGED, there is just a single value for all IOUT values.
	0.7	DW	Update the description of the CAPABILITY command. PE24103 does NOT implement support for PEC, but can support bus speeds of 1MHz.
	0.7	DW	Update description of the VOUT_TRANSITION_RATE command concerning a value of 0. A value of 0 means that the output voltage will not change at all, rather than change at the MAX possible rate. Also note that writing a –ve value will be treated as a +ve value.
	0.7	DW	Adjust description of TON_RISE and TOFF_FALL for negative values. If entered, the negative values will result in unexpected delay values.
	0.7	DW	Update the description of the STATUS_MFR_SPECIFIC register to reflect the bits in the finished 1.0 version of the design.
	0.7	DW	Throughout the document, tidy up various "FIXME" comments if the issue/decision is now resolved.
	0.7	DW	Add new Manufacturer Specific Command to allow the user to set the PMBus Device Address value. Note this command is not supported on ES1 silicon for PE24103.
	0.7	DW	Add a note to indicate which PMBus commands are not supported on R2D2 ES1 silicon.
25/Jan/21	0.7	DW	Add a note to the MFR_CFG_PMBUS command to note that there are reserved device address values. In particular they should not use the value we reserve for our private I²C address.
27/Jan/21	0.7	DW	For the following commands which accept Linear11 data formats: VOUT_TRANSITION_RATE, FREQUENCY_SWITCH, TON_DELAY, TOFF_DELAY, TON_RISE and TOFF_FALL negative values make no sense. Update the description to note what happens if negative values should be written to PE24103 silicon versions **AFTER** ES1.

Table 55 Document Change History

Date	Doc Version	Author	Description of Changes
27/Jan/21	0.7	DW	Fix the table for those value sstored by STORE_USER_ALL to add the manufacturer specific command which can set the PMBus device address. It got missed from the table.
27/Jan/21	0.8	DW	Update headers/footers and file version.
26/Mar/21	0.8	DW	Add new figures to illustrate the simple send byte, write byte, read byte, write word and read word packets which are used for almost all the commands. Insert the new drawings at the first instance of an appropriate command in the document.
26/Mar/21	0.8	DW	For each command include a new bit of text to point to the figure which shows the read/write packet type used for the command.
26/Mar/21	0.8	DW	Add new figure showing a generic BLOCK read command and refer to it for the MFR_ID command which PE24103 supports.
26/Mar/21	0.8	DW	Add new text for the PAGE_PLUS_WRITE and PAGE_PLUS_WRITE commands to detail that the commands will change the PAGE value on-chip (rather than just temporarily for the duration of the current command). This is apparently not universally true for PMBus devices.
26/Mar/21	0.8	DW	Add new text to the PAGE command to indicate that writing an illegal PAGE value will result in the PAGE value being set to 0.
29/Mar/21	0.8	DW	Add new text for the read-only commands to indicate that (for ES2 silicon) attempting to write to these commands will trigger a fault in the STATUS_CML register.
29/Mar/21	0.8	DW	Minor text clean-up. A few typos and a couple of Figure or Table cross-references which had gone wrong for some reason.
29/Mar/21	0.8	DW	Add new text for the STATUS_IOUT register decription around the differences between ES1 and ES2 silicon which were initially captured in this JIRA ticket: http://jira/browse/PMI-70 .
30/Mar/21	0.8	DW	Add new text to show that the IC_DEVICE_ID command uses a BLOCK read format (same as the MFR_ID).
31/Mar/21	0.9	DW	Update headers/footers and file version.
9/Jun/21	0.9	DW	Update all references to ES1/ES2 silicon. We now know that ES1 and ES2 versions of R2D2 silicon have the same digital block. Some of the features described in this document will only apply to future versions of R2D2 (later than ES2).

Table 55 Document Change History

Date	Doc Version	Author	Description of Changes