1 Introduction

The UCD92xx family of digital power controller supports a wide range of commands that allow an external host to configure, control, and monitor the controller. Communication between an UCD92xx controller and the host is via an I2C electrical interface using the PMBus command protocol.

The PMBus specification describes the command protocol in general terms. This document describes implementation details that are specific to the UCD92xx Digital Power Controllers.

Table of Contents:

2 PMBus Specification

This document makes frequent mention of the PMBus specification. Specifically, this document is "PMBus Power System Management Protocol Specification Part II – Command Language", Revision 1.1, dated 5 February 2007. The specification is published by the Power Management Bus Implementers Forum and is available from http://pmbus.org.

3 Data Formats

Sections 7 and 8 of the PMBus standard provides for a number of different data formats: three for parameters related to output voltage and two for all other commands. Each PMBus device is expected to support only one of these formats.

3.1 Data Format for Output Voltage Parameters

For parameters related to output voltage, the UCD92xx supports the Linear format defined in Section 8.3.1 of the PMBus specification. The linear format uses a 16-bit unsigned mantissa for each parameter, along with an exponent that is shared by all the voltage-related parameters. The exponent is reported in the bottom 5 bits of the VOUT_MODE parameter. In the UCD92xx, this exponent is a read-only parameter whose value is fixed at -12. This allows setting voltage-related variables over a range from 0 to 15.9997V, with a resolution of 0.244mV.

The voltage value is calculated using the equation

Voltage = V * 2^X,

where

Voltage is the parameter of interest, in volts,
V is a 16-bit unsigned binary integer mantissa, and
X is the signed 5-bit twos-complement binary integer exponent from VOUT_MODE.

Exception: The PMBus standard assumes that all output voltages are expressed as positive numbers, so all parameters related to output voltage are unsigned integers, with a few notable exceptions. The VOUT_CAL_OFFSET and VOUT_CAL_MONITOR values are intended for making fine adjustments to the output voltage, and may take on small negative values. As such, these parameters are treated as signed twos-complement binary integers.

3.2 Data Format for Other Parameters

For parameters not directly related to output voltage, the UCD92xx supports the Linear Data Format described in section 7.1 of the PMBus specification. This linear format is a two-byte value that contains an 11-bit, twos-complement mantissa and a 5-bit, twos-complement exponent.

The relationship between the PMBus parameter and the "real world" value is given by the formula:

R = Y * 2^X,

Where

R is the "real world" value,
Y is an 11-bit, signed twos-complement binary integer mantissa, and
X is the signed 5-bit twos-complement binary integer exponent.

This pseudo floating point notation allows values as large as ~33E6 down to ~15E-6 to be sent over the PMBus. The internal variables used by the UCD92xx firmware are mostly 16 bits wide and do not support such a wide range of values. The resolution of a PMBus setting depends strongly on both the exponent of the PMBus value (larger values have coarser resolution) and the scaling of the internal variables.

3.3 Distinguishing Between Linear Data Formats

The PMBus specification uses the same term, “Linear”, to describe both the 16-bit+exponent format used for the voltage-related parameters as well as the 11-bit+exponent format used for other parameters. In cases where it is necessary to distinguish between these two data formats, this document will use the term LINEAR16 or LINEAR11.

3.4 Translation, Quantization, and Truncation

The internal variables used by the UCD92xx are often scaled to take optimal advantage of the hardware’s native units such as ADC or DAC counts rather than volts or amperes. As a result, values that are written and read via PMBus must undergo mathematical translations. These translations, with their inherent quantization, may result in very slight differences between the setting that was written to the UCD92xx and the value that was later read back from it. This is normal and compliant, described in section 7.4 of the PMBus specification.

In some cases, a value written to the device may cause it to exceed the range of its internal variable. In some cases the device will report this as an error; in other cases it will saturate the variable at a safe value. In all cases, the value read back via the PMBus will reflect as accurately as possible the internal variable actually being used by the UCD92xx.

4 Memory Model

Section 6 of the PMBus specification describes the memory model for PMBus devices. Values used by the PMBus device are loaded into volatile Operating Memory from one of more of the following places.

Values hard coded into an IC design (if any),
Values programmed from hardware pins (if any),
A non-volatile memory called the Default Store (if supported in the device),
A non-volatile memory called the User Store (if supported in the device), or
Communications from the PMBus.

The UCD92xx contains RAM that is used as Operating Memory. On-board Data Flash memory is used to implement the hard-coded values and the Default Store values. Hard-coded values require a new firmware revision. Values in the Default Store may be changed using the STORE_DEFAULT_ALL command described in section 10.4. The User Store is not supported.

Section 6.1 of the PMBus specification describes the ordering of memory loading and precedence. In general, the hard-coded parameters are loaded into Operating Memory first. Second, any pin-programmable settings take effect. Third, values from the Default Store are loaded. Later, commands issued from the PMBus take effect. In all cases, an operation on a parameter will overwrite any prior value that was already in the Operating Memory.

5 Clearing a Shutdown due to a Fault

Section 10.7 of the PMBus specification describes the conditions required to resume power conversion for a device or rail that has been shutdown due to a fault. One of these conditions is that a rail must be commanded to turn OFF and then ON before power conversion can resume. Rails that are configured to be “Always On” via the ON_OFF_CONFIG PMBus command will remain latched off. Toggling the CONTROL PIN or OPERATION command will not clear the latched state. These rails can only be enabled by device reset or by changing the ON_OFF_CONFIG command to allow the CONTROL PIN or OPERATION command to clear the latched state. The UCD9224, UCD9246 and UCD9248 devices also allow toggling a TURN_ON dependent input for the rail to clear the latched state. However, toggling a STAY_ON dependent input for the rail will not clear the latched state.

6 Temperature Warning and Fault Monitoring

The external temperature sensors are used for temperature warning and fault detection using the PMBus commands OT_WARN_LIMIT and OT_FAULT_LIMIT for all UCD92xx devices. The UCD9211 and UCD9212 devices will report an over-temperature warning if the internal temperature exceeds 85C.

7 Hardware Over-Current Fault Response

The UCD92xx device family has two forms of hardware assistance for over-current faults. The first is an external driver fault input pin and the second is a set of analog comparators that monitor the CS-1A, CS-2A, CS-3A and CS-4A input pins. The analog comparator threshold can be configured using the FAST_OC_FAULT_LIMIT command described in this document.

Both types of hardware over-current detection use the IOUT_OC_FAULT_RESPONSE_CODE to configure how the UCD92xx device reacts when a hardware over-current fault event occurs with the exception that only the RETRY_SETTING and DELAY_TIME bits are valid. The RESPONSE bits are ignored because the UCD92xx device will always shut down the output Rail in response to a hardware over-current event.

8 Alert Response Address Support

The UCD92xx supports using the PMBALERT# line to notify the host of warning or fault conditions and also supports the Alert Response Address protocol with the following exception. The UCD92xx will not respond to its address on the PMBus or another Alert Response Address event when it loses arbitration during an Alert Response Address event. Any PMBus command will clear this state, even commands addressed to other devices. Subsequent commands and Alert Response Address events will be responded to normally.

9 Supported PMBus Commands

Table 1 lists the PMBus commands. Commands 00h through CFh are defined in the PMBus specification and are considered to be “core” commands that are standardized for all manufacturers and products. Commands D0h through FEh are manufacturer-specific and may be unique for each manufacturer and product.

Commands that are not supported by the UCD92xx are indicated in the Comments column.

Most commands support writing and reading. Exceptions are indicated in the Comments column.

The Data Format column indicates the format of the data:

Byte	8-bit binary value. Refer to the PMBus specification for details for each command.
LINEAR16	16-bit Linear format used for output voltage parameters. Described in 3.1 above.
LINEAR11	11-bit Linear format used parameters other than output voltage. Described in section 3.2 above
n/a	Command does not have a data field.
String	ASCII string. Described in section 22.2 of the PMBus specification.
Byte Array	A block of data in binary format.

The Scope column indicates how each command is affected by the PAGE and PHASE settings.

Common	This command does not depend on the PAGE or PHASE setting. It is a common

	variable used by all pages and phases.
PAGE	This command applies to the page(s) set by the most recent PAGE command. See section 10.1 below for details.
PAGE & PHASE	This command applies to the phase(s) set by the most recent PHASE command. See section 10.3 below for details. All commands that depend on PHASE also depend on PAGE.

Table 1: PMBus Commands

Code	Command Name	Transaction Type	Data Format [Units]	Scope	Supported Models (Comments)
00h	PAGE	R/W Byte	Byte	Common	All
01h	OPERATION	R/W Byte	Byte	PAGE	All
02h	ON_OFF_CONFIG	R/W Byte	Byte	PAGE	All
03h	CLEAR_FAULTS	Send byte	n/a	Common	All (Write Only)
04h	PHASE	R/W Byte	Byte	PAGE	All
05h	PAGE_PLUS_WRITE	Write Block	n/a	Common	PMBus 1.2 (*1)
06h	PAGE_PLUS_READ	Block Write – Block Read Process Call	n/a	Common	‘PMBus 1.2 (*1)
07h-0Fh	Reserved				NOT SUPPORTED
10h	WRITE_PROTECT	R/W Byte	Byte		NOT SUPPORTED
11h	STORE_DEFAULT_ALL	Send byte	n/a	Common	All
12h	RESTORE_DEFAULT_ALL	Send byte	n/a	Common	All
13h	STORE_DEFAULT_CODE	Write Byte	Byte		NOT SUPPORTED
14h	RESTORE_DEFAULT_CODE	Write Byte	Byte		NOT SUPPORTED
15h	STORE_USER_ALL	Send byte	n/a	Common	NOT SUPPORTED
16h	RESTORE_USER_ALL	Send byte	n/a	Common	NOT SUPPORTED
17h	STORE_USER_CODE	Write Byte	Byte		NOT SUPPORTED
18h	RESTORE_USER_CODE	Write Byte	Byte		NOT SUPPORTED
19h	CAPABILITY	R/W Byte	Byte	Common	All (Read Only)
1Ah	QUERY	Block Write – Block Read Process Call	Byte	Common	NOT SUPPORTED
1Bh	SMBALERT_MASK	R/W Word	Word	PAGE	PMBus 1.2 (*1)
1Ch-1Fh	Reserved				NOT SUPPORTED
20h	VOUT_MODE	R/W Byte	LINEAR16 [V]	PAGE	All (Read Only)
21h	VOUT_COMMAND	R/W Word	LINEAR16 [V]	PAGE	All
22h	VOUT_TRIM	R/W Word	LINEAR16 [V]	PAGE	‘9211, ‘9212
23h	VOUT_CAL_OFFSET	R/W Word	LINEAR16 [V]	PAGE	All
24h	VOUT_MAX	R/W Word	LINEAR16 [V]	PAGE	All
25h	VOUT_MARGIN_HIGH	R/W Word	LINEAR16 [V]	PAGE	All
26h	VOUT_MARGIN_LOW	R/W Word	LINEAR16 [V]	PAGE	All
27h	VOUT_TRANSITION_RATE	R/W Word	LINEAR11 [V/ms]	PAGE	All
28h	VOUT_DROOP	R/W Word	LINEAR16 [V]	PAGE	NOT SUPPORTED
29h	VOUT_SCALE_LOOP	R/W Word	LINEAR11 [V/V]	PAGE	All
2Ah	VOUT_SCALE_MONITOR	R/W Word	LINEAR11 [V/V]	PAGE	All
2Bh-2Fh	Reserved				NOT SUPPORTED
30h	COEFFICIENTS	Block Write – Block Read Process Call	Byte Array	PAGE	NOT SUPPORTED
31h	POUT_MAX	R/W Word	LINEAR11 [W]	PAGE	NOT SUPPORTED

Code	Command Name	Transaction Type	Data Format [Units]	Scope	Supported Models (Comments)
32h	MAX_DUTY	R/W Word	LINEAR11 [0-100%]	PAGE	All except '9224, '9246, '9248
33h	FREQUENCY_SWITCH	R/W Word	LINEAR11 [kHz]	PAGE	All
34h	Reserved				NOT SUPPORTED
35h	VIN_ON	R/W Word	LINEAR11 [V]	Common	All
36h	VIN_OFF	R/W Word	LINEAR11 [V]	Common	All
37h	INTERLEAVE	R/W Word	Word	Common	NOT SUPPORTED
38h	IOUT_CAL_GAIN	R/W Word	LINEAR11 [mV/A = mΩ]	PAGE & PHASE	All
39h	IOUT_CAL_OFFSET	R/W Word	LINEAR11 [A]	PAGE & PHASE	All
3Ah	FAN_CONFIG_1_2	R/W Byte	Byte	Common	'9220, 9240, '9246F
3Bh	FAN_COMMAND_1	R/W Word	LINEAR11 [%]	Common	'9220, 9240, '9246F
3Ch	FAN_COMMAND_2	R/W Word	LINEAR11 [%]	Common	NOT SUPPORTED
3Dh	FAN_CONFIG_3_4	R/W Byte	Byte	Common	NOT SUPPORTED
3Eh	FAN_COMMAND_3	R/W Word	LINEAR11 [%]	Common	NOT SUPPORTED
3Fh	FAN_COMMAND_4	R/W Word	LINEAR11 [%]	Common	NOT SUPPORTED
40h	VOUT_OV_FAULT_LIMIT	R/W Word	LINEAR16 [V]	PAGE	All
41h	VOUT_OV_FAULT_RESPONSE	R/W Byte	Byte	PAGE	All
42h	VOUT_OV_WARN_LIMIT	R/W Word	LINEAR16 [V]	PAGE	All
43h	VOUT_UV_WARN_LIMIT	R/W Word	LINEAR16 [V]	PAGE	All
44h	VOUT_UV_FAULT_LIMIT	R/W Word	LINEAR16 [V]	PAGE	All
45h	VOUT_UV_FAULT_RESPONSE	R/W Byte	Byte	PAGE	All
46h	IOUT_OC_FAULT_LIMIT	R/W Word	LINEAR11 [A]	PAGE	All
47h	IOUT_OC_FAULT_RESPONSE	R/W Byte	Byte	PAGE	All
48h	IOUT_OC_LV_FAULT_LIMIT	R/W Word	LINEAR16 [V]	PAGE	All
49h	IOUT_OC_LV_FAULT_RESPONSE	R/W Byte	Byte	PAGE	All
4Ah	IOUT_OC_WARN_LIMIT	R/W Word	LINEAR11 [A]	PAGE	All
4Bh	IOUT_UC_FAULT_LIMIT	R/W Word	LINEAR11 [A]	PAGE	All
4Ch	IOUT_UC_FAULT_RESPONSE	R/W Byte	Byte	PAGE	All
4Dh	Reserved				NOT SUPPORTED
4Eh	Reserved				NOT SUPPORTED
4Fh	OT_FAULT_LIMIT	R/W Word	LINEAR11 [°C]	PAGE	All
50h	OT_FAULT_RESPONSE	R/W Byte	Byte	PAGE	All
51h	OT_WARN_LIMIT	R/W Word	LINEAR11 [°C]	PAGE	All
52h	UT_WARN_LIMIT	R/W Word	LINEAR11 [°C]	PAGE	NOT SUPPORTED
53h	UT_FAULT_LIMIT	R/W Word	LINEAR11 [°C]	PAGE	NOT SUPPORTED
54h	UT_FAULT_RESPONSE	R/W Byte	Byte	PAGE	NOT SUPPORTED
55h	VIN_OV_FAULT_LIMIT	R/W Word	LINEAR11 [V]	Common	All
56h	VIN_OV_FAULT_RESPONSE	R/W Byte	Byte	Common	All
57h	VIN_OV_WARN_LIMIT	R/W Word	LINEAR11 [V]	Common	All
58h	VIN_UV_WARN_LIMIT	R/W Word	LINEAR11 [V]	Common	All
59h	VIN_UV_FAULT_LIMIT	R/W Word	LINEAR11 [V]	Common	All
5Ah	VIN_UV_FAULT_RESPONSE	R/W Byte	Byte	Common	All
5Bh	IIN_OC_FAULT_LIMIT	R/W Word	LINEAR11 [A]	PAGE	NOT SUPPORTED
5Ch	IIN_OC_FAULT_RESPONSE	R/W Byte	Byte	PAGE	NOT SUPPORTED
5Dh	IIN_OC_WARN_LIMIT	R/W Word	LINEAR11 [A]	PAGE	NOT SUPPORTED
5Eh	POWER_GOOD_ON	R/W Word	LINEAR16 [V]	PAGE	All
5Fh	POWER_GOOD_OFF	R/W Word	LINEAR16 [V]	PAGE	All
60h	TON_DELAY	R/W Word	LINEAR11 [ms]	PAGE	All
61h	TON_RISE	R/W Word	LINEAR11 [ms]	PAGE	All
62h	TON_MAX_FAULT_LIMIT	R/W Word	LINEAR11 [ms]	PAGE	All
63h	TON_MAX_FAULT_RESPONSE	R/W Byte	Byte	PAGE	All
64h	TOFF_DELAY	R/W Word	LINEAR11 [ms]	PAGE	All
65h	TOFF_FALL	R/W Word	LINEAR11 [ms]	PAGE	All
66h	TOFF_MAX_WARN_LIMIT	R/W Word	LINEAR11 [ms]	PAGE	All
67h	Reserved				NOT SUPPORTED
68h	POUT_OP_FAULT_LIMIT	R/W Word	LINEAR11 [W]	PAGE	NOT SUPPORTED
69h	POUT_OP_FAULT_RESPONSE	R/W Byte	Byte	PAGE	NOT SUPPORTED
6Ah	POUT_OP_WARN_LIMIT	R/W Word	LINEAR11 [W]	PAGE	NOT SUPPORTED

Code	Command Name	Transaction Type	Data Format [Units]	Scope	Supported Models (Comments)
6Bh	PIN_OP_WARN_LIMIT	R/W Word	LINEAR11 [W]	Common	NOT SUPPORTED
6Ch-77h	Reserved				NOT SUPPORTED
78h	STATUS_BYTE	R/W Byte	Byte	Common	All (Read Only, *2)
79h	STATUS_WORD	R/W Word	Word	Common	All (Read Only, *2)
7Ah	STATUS_VOUT	R/W Byte	Bye	PAGE	All (Read Only, *2)
7Bh	STATUS_IOUT	R/W Byte	Byte	PAGE	All (Read Only, *2)
7Ch	STATUS_INPUT	R/W Byte	Byte	PAGE	All (Read Only, *2)
7Dh	STATUS_TEMPERATURE	R/W Byte	Byte	PAGE	All (Read Only, *2)
7Eh	STATUS_CML	R/W Byte	Byte	Common	All (Read Only, *2)
7Fh	STATUS_OTHER	R/W Byte	Byte	Common	All (Read Only, *2)
80h	STATUS_MFR_SPECIFIC	R/W Byte	Byte	PAGE	All (Read Only, *2)
81h	STATUS_FANS_1_2	R/W Byte	Byte	Common	‘9220, ‘9240, ‘9246F (Read Only, *2)
82h	STATUS_FANS_3_4	R/W Byte	Byte	Common	‘9220, ‘9240, ‘9246F (Read Only, *2) Always returns 0x00
83h-87h	Reserved				NOT SUPPORTED
88h	READ_VIN	R/W Word	LINEAR11 [V]	Common	All (Read Only)
89h	READ_IIN	R/W Word	LINEAR11 [A]	Common	All except ‘9244 variants (Read Only)
8Ah	READ_VCAP	R/W Word	LINEAR16 [V]	PAGE	NOT SUPPORTED
8Bh	READ_VOUT	R/W Word	LINEAR16 [V]	PAGE	All (Read Only)
8Ch	READ_IOUT	R/W Word	LINEAR11 [A]	PAGE & PHASE	All (Read Only)
8Dh	READ_TEMPERATURE_1	R/W Word	LINEAR11 [°C]	Common	All (Read Only)
8Eh	READ_TEMPERATURE_2	R/W Word	LINEAR11 [°C]	PAGE & PHASE	All (Read Only)
8Fh	READ_TEMPERATURE_3	R/W Word	LINEAR11 [°C]	Common	NOT SUPPORTED
90h	READ_FAN_SPEED_1	R/W Word	LINEAR11 [RPM]	Common	‘9220, ‘9240, ‘9246F (Read Only)
91h	READ_FAN_SPEED_2	R/W Word	LINEAR11 [RPM]	Common	NOT SUPPORTED
92h	READ_FAN_SPEED_3	R/W Word	LINEAR11 [RPM]	Common	NOT SUPPORTED
93h	READ_FAN_SPEED_4	R/W Word	LINEAR11 [RPM]	Common	NOT SUPPORTED
94h	READ_DUTY_CYCLE	R/W Word	LINEAR11 [0-100%]	PAGE	All (Read Only)
95h	READ_FREQUENCY	R/W Word	LINEAR11 [kHz]	PAGE	NOT SUPPORTED
96h	READ_POUT	R/W Word	LINEAR11 [W]	PAGE	All (Read Only)
97h	READ_PIN	R/W Word	LINEAR11 [W]	Common	All except ‘9244 variants (Read Only)
98h	PMBUS_REVISION	R/W Byte	Byte	Common	All (Read Only)
99h	MFR_ID	R/W block 18 bytes	String	Common	All
9Ah	MFR_MODEL	R/W block 12 bytes	String	Common	All
9Bh	MFR_REVISION	R/W block 12 bytes	String	Common	All
9Ch	MFR_LOCATION	R/W block 12 bytes	String	Common	All
9Dh	MFR_DATE	R/W block 6 bytes	String	Common	All
9Eh	MFR_SERIAL	R/W block 12 bytes	String	Common	All
9Fh	Reserved				
A0h	MFR_VIN_MIN	R/W Word	LINEAR11 [V]	Common	NOT SUPPORTED
A1h	MFR_VIN_MAX	R/W Word	LINEAR11 [V]	Common	NOT SUPPORTED
A2h	MFR_IIN_MAX	R/W Word	LINEAR11 [A]	Common	NOT SUPPORTED

Code	Command Name	Transaction Type	Data Format [Units]	Scope	Supported Models (Comments)
A3h	MFR_PIN_MAX	R/W Word	LINEAR11 [W]	Common	NOT SUPPORTED
A4h	MFR_VOUT_MIN	R/W Word	LINEAR11 [V]	Common	NOT SUPPORTED
A5h	MFR_VOUT_MAX	R/W Word	LINEAR11 [V]	Common	NOT SUPPORTED
A6h	MFR_IOUT_MAX	R/W Word	LINEAR11 [A]	PAGE	NOT SUPPORTED
A7h	MFR_POUT_MAX	R/W Word	LINEAR11 [W]	Common	NOT SUPPORTED
A8h	MFR_TAMBIENT_MAX	R/W Word	LINEAR11 [℃]	Common	NOT SUPPORTED
A9h	MFR_TAMBIENT_MIN	R/W Word	LINEAR11 [℃]	Common	NOT SUPPORTED
AAh	MFR_EFFICIENCY_LL	R/W Word	LINEAR11 [%]	Common	NOT SUPPORTED
ABh	MFR_EFFICIENCY_LL	R/W Word	LINEAR11 [%]	Common	NOT SUPPORTED
ACh	MFR_EFFICIENCY_PIN	R/W Word	LINEAR11 [%]	Common	NOT SUPPORTED
ADh	IC_DEVICE_ID	Read Block 10-11 bytes	String	Common	PMBus 1.2 (*1), Read Only
AEh	IC_DEVICE_REV	Read Block 16-20 bytes	String	Common	PMBus 1.2 (*1), Read Only
AFh	Reserved				NOT SUPPORTED
B0h	USER_DATA_00	R/W Block 32 bytes	String	Common	All except ‘9220, ‘9240,
B1h	STATISTICS_VALUES (USER_DATA_01)	R/W Block 16 bytes	Byte Array	Common	‘9211, ‘9212
B2h	STATISTICS_MAXMIN (USER_DATA_02)	R/W Block 32 bytes	Byte Array	Common	‘9211, ‘9212
B3h	BLACKBOX_VALUES (USER_DATA_03)	R/W Block 20 bytes	Byte Array	Common	‘9211, ‘9212
B4h	BLACKBOX_MAXMIN (USER_DATA_04)	R/W Block 32 bytes	Byte Array	Common	‘9211, ‘9212
B5h	FAULT_EVENT_CONTROL (USER_DATA_05)	R/W Word	Word	Common	‘9211, ‘9212
B6h-B8h					
B9h	SYNC_CONFIG (USER_DATA_09)	R/W Block 9 bytes	Byte Array	Common	‘9222, ‘9244
BAh	READ_AUX_ADCS (USER_DATA_10)	Read Block 9 bytes?	Byte Array	Common	‘9222, ‘9244
BBh	VID_CONFIG (USER_DATA_11)	R/W Block 8 bytes?	Byte Array	PAGE	‘9222, ‘9244
BCh	VID_CODE_RAIL1 (USER_DATA_12)	R/W Byte	Byte	Common	‘9222, ‘9244
BDh	VID_CODE_RAIL2 (USER_DATA_13)	R/W Byte	Byte	Common	‘9222, ‘9244
BEh	VID_CODE_RAIL3 (USER_DATA_14)	R/W Byte	Byte	Common	‘9244
BFh	VID_CODE_RAIL4 (USER_DATA_15)	R/W Byte	Byte	Common	‘9244
C0h-CFh	Reserved				NOT SUPPORTED
D0h	SEQ_TIMEOUT (MFR_SPECIFIC_00)	R/W Word	LINEAR11 [ms]	PAGE	‘9220, ‘9240, ‘9224, ‘9246, ‘9248 ‘9246F, ‘9224E, ‘9246E, ‘9248E
D1h	VOUT_CAL_MONITOR (MFR_SPECIFIC_01)	R/W Word	LINEAR16 [V]	PAGE	All
D2h	PHASE_INFO (MFR_SPECIFIC_02)	R/W block 4 bytes	Byte Array	Common	All
D3h	VIN_SCALE_MONITOR (MFR_SPECIFIC_03)	R/W Word	LINEAR11 [V/V]	Common	All
D4h	CLA_BANK (MFR_SPECIFIC_04)	R/W Byte	Byte		All
D5h	CLA_GAINS (MFR_ SPECIFIC_05)	R/W Block 16 bytes	Byte Array	PAGE & BANK	All
D6h	PAGE_ISOLATED (MFR_SPECIFIC_06)	R/W Byte	Byte	PAGE	‘9220, ‘9240 ‘9211, ‘9212
	DRIVER_CONFIG (MFR_SPECIFIC_06)	R/W Byte	Byte	PAGE	All others
D7h	EADC_SAMPLE_TRIGGER (MFR_SPECIFIC_07)	R/W Word	LINEAR11 [ns]	PAGE	All
D8h	ACTIVATE_CLA_BANK (MFR_SPECIFIC_08)	R/W Byte	Byte	PAGE	All
D9h	ROM_MODE (MFR_SPECIFIC_09)	Send Byte	n/a	Common	All
DAh	USER_RAM_00 (MFR_SPECIFIC_10)	R/W Byte	Byte	Common	All

Code	Command Name	Transaction Type	Data Format [Units]	Scope	Supported Models (Comments)
DBh	SOFT_RESET (MFR_SPECIFIC_11)	Send Byte	n/a	Common	All
DCh	IIN_SCALE_MONITOR (MFR_SPECIFIC_12)	R/W Word	LINEAR11 [V/A]	Common	All except ‘9244 variants
DDh	THERMAL_COEF (MFR_SPECIFIC_13)	R/W Word	LINEAR11 [%/°C]	PAGE	All
DEh	PHASE_ENABLE (MFR_SPECIFIC_14)	R/W Byte	Byte	PAGE	All
DFh	DRIVER_MIN_PULSE (MFR_SPECIFIC_15)	R/W Word	LINEAR11 [ns]	PAGE	All
E0h	MIN_DUTY (MFR_SPECIFIC_16)	R/W Word	LINEAR11 [0-100%]	PAGE	‘9220, ‘9240 ‘9211, ‘9212
E1h	SYNC_IN_OUT (MFR_SPECIFIC_17)	R/W Word	2 Bytes	Common	All except ‘9222, ‘9244
E2h	PARM_INFO (MFR_SPECIFIC_18)	R/W block 5 bytes	Byte Array	Common	All
E3h	PARM_VALUE (MFR_SPECIFIC_19)	R/W block	Byte Array	Common	All
E4h	TEMPERATURE_CAL_GAIN (MFR_SPECIFIC_20)	R/W Word	LINEAR11 [°C/V]	PAGE & PHASE	All
E5h	TEMPERATURE_CAL_OFFSET (MFR_SPECIFIC_21)	R/W Word	LINEAR11 [°C]	PAGE & PHASE	All
E6h	TRACKING_SOURCE (MFR_SPECIFIC_22)	R/W Byte	Byte	PAGE	All (‘9244 variants support only internal tracking)
E7h	TRACKING_SCALE_MONITOR (MFR_SPECIFIC_23)	R/W Word	LINEAR11 [V/V]	PAGE	All
E8h	FAN_SPEED_FAULT_LIMIT (MFR_SPECIFIC_24)	R/W Word	LINEAR11 [RPM]	Common	‘9220, ‘9240, ‘9246F
E9h	LOGGED_PEAKS (MFR_SPECIFIC_25)	R/W Block 17 bytes	Byte Array Byte [°C] and LINEAR11 [A]	Common	All
EAh	LOGGED_FAULTS (MFR_SPECIFIC_2	R/W Block 9 bytes	Byte Array	Common	All
EBh	LIGHT_LOAD_LIMIT_HIGH (MFR_SPECIFIC_27)	R/W Word	LINEAR11 [A]	PAGE	All
ECh	DEVICE_ID / ROM_VER (MFR_SPECIFIC_28)	Read block 4 bytes or up to 32 bytes	Byte Array or String	Common	All (Read Only)
EDh	LIGHT_LOAD_CONFIG (MFR_SPECIFIC_29)	R/W Byte	Byte	PAGE	All
EEh	PREBIAS_OFFSET	R/W Word	LINEAR11 [%]	PAGE	‘9220, ‘9240
EFh	PREBIAS_GAIN	R/W Word	LINEAR11 [%/%]	PAGE	‘9220, ‘9240
F0h	EXECUTE_FLASH (MFR_SPECIFIC_32)	Send Byte	n/a	Common	All (Write Only)
F1h	MFR_SETUP_PASSWORD (MFR_SPECIFIC_33)	R/W Block 6 bytes	Binary Array	Common	All
F2h	DISABLE_SECURITY (MFR_SPECIFIC_34)	R/W Block 6 bytes	Binary Array	Common	All
F3h	GPIO_SEQ_CONFIG (MFR_SPECIFIC_35)	R/W Block 29 bytes	Binary Array	Common	‘9220, ‘9240
	GPIO_SEQ_CONFIG (MFR_SPECIFIC_35)	R/W Block 32 bytes	Binary Array	Common	‘9224, ‘9246, ‘9248 ‘9246F, ‘9224E, ‘9246E, ‘9248E
F4h	MFR_SECURITY_BIT_MASK (MFR_SPECIFIC_36)	R/W Block 32 bytes	Binary Array	Common	All
F5h	TEMP_BALANCE_IMIN (MFR_SPECIFIC_37)	R/W Word	LINEAR11 [A]	PAGE	All except ‘9211, ‘9222, ‘9244
F6h	LIGHT_LOAD_LIMIT_LOW (MFR_SPECIFIC_38)	R/W Word	LINEAR11 [A]	PAGE	All
F7h	FAST_OC_FAULT_LIMIT (MFR_SPECIFIC_39)	R/W Word	LINEAR11 [A]	PAGE	All
F8h	POWER_GOOD_CONFIG (MFR_SPECIFIC_40)	R/W Byte	Byte	Common	‘9222A, ‘9244A, ‘9244N
F9h	(MFR_SPECIFIC_41)				NOT SUPPORTED

Code	Command Name	Transaction Type	Data Format [Units]	Scope	Supported Models (Comments)
FAh	PHASE_DROP_CAL (MFR_SPECIFIC_42)	R/W Word	LINEAR11 [Switching cycles / Amp]	PAGE	'All except '9220, '9240
FBh	SYNC_OFFSET (MFR_SPECIFIC_43)	R/W Word	LINEAR11 [ns]	PAGE	'9224, '9246, '9248 '9246F, '9224E, '9246E, '9248E
FCh	EADC_TRIM (MFR_SPECIFIC_44)	R/W Byte	Byte	PAGE	'9220, '9240,
	VID_RESTART (MFR_SPECIFIC_44)	Send Byte	n/a	PAGE	'9222A, '9244A, '9244N
FDh	DEVICE_ID (MFR_SPECIFIC_45)	Read Block up to 32 bytes	String	Common	All (Read Only)
FEh	Mfr_Specific_Extended_Command				NOT SUPPORTED
FFh	PMBUS_Extended_Command				NOT SUPPORTED

[12]

[1] PMBus Revision 1.1 is supported by UD9220, '9240, '9211, '9212, '9224, '9246, '9248, and '9246F. PMBus Revision 1.2 is supported by 9224E, '9246E, '9248E, and all variants of '9222 and '9244.

[2] STATUS_xxx commands are read-only in PMBus revision 1.1 and are R/W in PMBus revision 1.2.

10 Implementation Details for PMBus Core Commands

These PMBus core commands are defined in the PMBus specification. This section describes details that are unique to the UCD92xx implementation.

10.1 (00h) PAGE

The PAGE command provides the ability to configure, control, and monitor multiple outputs on one unit using a single PMBus physical address. All subsequent commands that depend on page will be applied to the rail selected by the PAGE command.

The Fusion Digital Power Designer software uses the term “Rail” to refer to a voltage output. Rails are numbered starting with one, while pages are numbered starting at zero. The relationship between the PMBus PAGE value and the Rail number is shown in Table 2.

Setting PAGE = 0xFF means that the following write command are to be applied to all outputs. A page setting of 0xFF is invalid for all read commands with scope of PAGE or PAGE & PHASE.

Table 2: Relationship between PAGE and Rail

Page	Output Rail
0	1
1*	2
2*	3
3*	4
4 – 0xFE	Invalid
0xFF	All

(*) The maximum number of rails on the UCD9240 is four. In some power supply configurations, multiple phases may be bridged together to increase the current capability. This may result in fewer than four independent rails. (Note that the UCD9220 only supports two rails and the UCD921x only supports one). The number of rails is configured using the PHASE_INFO command, described in section 13 below.

Section 11.10 of the PMBus specification describes the PAGE command in more detail.

10.2 (01h) OPERATION

This command is used to turn outputs on and off in conjunction with input from the CONTROL pin. Section 12.1 of the PMBus specification describes this command in more detail.

The UCD92xx supports the following modes for the Operation command:

Immediate Off (No Sequencing)
Soft Off (With Sequencing)
On Nominal (No Margining)
On Margin High (Ignore Faults)
On Margin High (Act on Fault)
On Margin Low (Ignore Faults)
On Margin Low (Act on Fault)

The parameter is not stored in the Default Store in Data Flash.

10.3 (04h) PHASE

This command selects a specific phase on a multi-phase output rail. All subsequent commands that depend on phase will be applied to the phase(s) selected by the PHASE command.

The number of phases for each page may be determined from the PHASE_INFO command (D2h) (see section 13 below). The phase numbering for each page starts at zero and goes up to one less than the number of phases on the selected page.

Setting PHASE = FF will cause subsequent write commands to be applied to all the phases on the selected page. Issuing a read command while PHASE = 0xFF has results that depend on the command.

Command	Response when PHASE = 0xFF for Read Commands
READ_IOUT	Returns total current for all phases on selected page.
IOUT_CAL_GAIN	Invalid Data
IOUT_CAL_OFFSET	Invalid Data
READ_TEMPERATURE_2	Returns highest temperature for all phases on selected page.
TEMPERATURE_CAL_GAIN (MFR_SPECIFIC_20)	Invalid Data
TEMPERATURE_CAL_OFFSET (MFR_SPECIFIC_21)	Invalid Data

10.4 (11h) STORE_DEFAULT_ALL

The STORE_DEFAULT_ALL command will save the PMBus parameters from Operating Memory into the Default Store in Data Flash. The UCD92xx will use the most recently written set of Default Store values at startup or after a RESTORE_DEFAULT_ALL command. If the Default Store has never been written, values from the hard-coded memory will be used. NOTE: The OPERATION and PHASE_ENABLE parameters are not stored in the Default Store in Data Flash.

10.5 (12h) RESTORE_DEFAULT_ALL

The RESTORE_DEFAULT_ALL command will restore the PMBus parameters from the Default Store into Operating Memory. If the Default Store has never been written, values from the hard-coded memory will be used.

CAUTION: If the RESTORE_DEFAULT_ALL command is issued while one or more voltage outputs are enabled, the output could behave erratically, with possibly catastrophic consequences.

10.6 (20h) VOUT_MODE

This command, described in sections 8.1 and 8.2 of the PMBus specification, indicates the data format used for all commands related to output voltage. In the UCD92xx, VOUT_MODE is a read-only parameter that is fixed to use Linear data format (described in section 3.1 above), with a fixed scaling exponent of -12.

10.7 (33h) FREQUENCY_SWITCH

This command, described in sections 14.4 of the PMBus specification. All outputs must be OFF before this parameter can be written to the device. If any outputs are enabled the device will NACK writes to the FREQUENCY_SWITCH command.

10.8 (38h) IOUT_CAL_GAIN

This command, described in section 14.8 of the PMBus specification, is used to configure the gain of the current sense circuit. The units for this command are milliohms (mV/A).

Note that there is some ambiguity in Rev 1.1 of the PMBus specification. One sentence says that the command uses the *conductance* of the sense resistor, but a later sentence says that the resistance should be used. In one place the units are listed as ohms, but the default value is declared as 0 milliohms.

Rev 1.0 of the PMBus specification used resistance (not conductance) and milliohms. It is expected that Rev 1.2 will clear up this ambiguity and revert to the Rev 1.0 wording. In the UCD92xx firmware, the ambiguous language in Rev 1.1 is interpreted to have the same meaning as Rev 1.0 (i.e. milliohms).

10.9 (3Ah) FAN_CONFIG_1_2

This command, described in section 14.10 of the PMBus specification, is used to configure the fan controlled by the UCD9240. Note that UCD9220 and UCD921x do not have I/O pins to support a fan so this command is meaningless for those models.

The UCD9240 does not support all possible settings of FAN_CONFIG_1_2. Notably, only Fan 1 is supported, not Fan 2. Writing any non-zero value to the Fan 2 controls (bits 3:0), will return a NACK due to Invalid Data.
In addition, Fan1 may only be commanded in Duty Cycle mode, not in RPM. Attempting to write a non-zero value to bit 6 will return a NACK due to Invalid Data.

Bit(s)	Value	Meaning
7	1	A fan is installed in Position 1
	0	No fan is installed in Position 1
6	1	Fan 1 is commanded in RPM. – NOT SUPPORTED
	0	Fan 1 is commanded in Duty Cycle
5:4	0:3	Fan 1 Tachometer Pulses Per Revolution 0= 1 pulse per revolution 1= 2 pulses per revolution 2= 3 pulses per revolution 3= 4 pulses per revolution
3:0		Fan 2 commands – NOT SUPPORTED

10.10 (3Bh) FAN_COMMAND_1

This command, described in section 14.12 of the PMBus specification, is used to adjust the operation of Fan 1 controlled by the UCD9240. . Note that UCD9220 and UCD921x do not have I/O pins to support a fan so this command is meaningless for those models. The PMBus specification describes two ways of setting the fan command (RPM or Duty Cycle), but the UCD9240 only supports the Duty Cycle mode.

The command has two data bytes formatted in the Linear11 data format. The units are percent duty cycle, from 0 to 100%.

10.11 (41h – 69h) xxx_FAULT_RESPONSE

The data bytes of the _FAULT_RESPONSE commands are described in sections 10.5.1 and 10.5.2 of the PMBus specification. In each case, bits 2:0 identify the number of delay time. The delay time is used for either the amount of time a unit is to continue operating after a fault is detected, or for the amount of time between attempts to restart. The units of the time delay, which vary depending on the type of fault, are shown in the table below.

Code	Command	Shutdown Delay Units	Retry Delay Units	Comment
41h	VOUT_OV_FAULT_RESPONSE	5 ms	5 ms	Note 1
45h	VOUT_UV_FAULT_RESPONSE	5 ms	5 ms	Note 1
47h	IOUT_OC_FAULT_RESPONSE	5 ms	5 ms	

49h	IOUT_OC_LV_FAULT_RESPONSE	5 ms	5 ms	Note 1
4Ch	IOUT_UC_FAULT_RESPONSE	5 ms	5 ms	
50h	OT_FAULT_RESPONSE	1.0 sec	1.0 sec	Note 1
54h	UT_FAULT_RESPONSE	1.0 sec	1.0 sec	NOT SUPPORTED
56h	VIN_OV_FAULT_RESPONSE	10 ms	5 ms	Note 1
5Ah	VIN_UV_FAULT_RESPONSE	10 ms	5 ms	Note 1
5Ch	IIN_OC_FAULT_RESPONSE	10 ms	5 ms	NOT SUPPORTED
63h	TON_MAX_FAULT_RESPONSE	5 ms	5 ms	Note 1
69h	POUT_OP_FAULT_RESPONSE	5 ms	5 ms	NOT SUPPORTED

NOTE 1: Section 10.5.1 of the PMBus specification describes the response to voltage, temperature and TON_MAX faults. For the UCD92xx, the Response setting (bits [7:6]) of 11b “The device’s output is disabled while the fault is present. Operation resumes and the output is enabled when the fault condition no longer exists” is not supported.

10.12 (61h) TON_RISE

This command, described in section 16.2 of the PMBus specification, sets the time, in ms, from when the output starts to rise until the voltage has entered the regulation band. The UCD92xx implementation deviates slightly from the PMBus standard. According to the PMBus standard, setting this parameter to a value of 0 will instruct the unit to bring its output voltage to the programmed regulation value as quickly as possible. The UCD92xx will bring its output voltage to the programmed regulation value as quickly as possible any time that TON_RISE is set to a value faster than it can achieve.

10.13 (62h) TON_MAX FAULT LIMIT

This command, described in section 16.3 of the PMBus specification, sets an upper limit, in ms, on how long the unit can attempt to power up the output without reaching the under-voltage fault limit. The UCD92xx implementation of TON_MAX_FAULT_LIMIT does not include the TON_DELAY time as part of this upper limit. The UCD92xx measured the time from the end of TON_DELAY until the output voltage reaches the commanded output voltage.

10.14 (65h) TOFF_FALL

This command, described in section 16.6 of the PMBus specification, sets the time, in ms, from the end of the turn-off delay time until the voltage is commanded to zero. The UCD92xx implementation deviates slightly from the PMBus standard. According to the PMBus standard, setting this parameter to a value of 0 will instruct the unit to bring its output voltage down as quickly as possible without exceeding the IOUT_UC_FAULT_LIMIT current. The UCD92xx will bring its output voltage down as quickly as possible any time that TOFF_FALL is set to a value faster than it can achieve.

10.15 (66h) TOFF_MAX_WARN_LIMIT

This command, described in section 16.7 of the PMBus specification, sets an upper limit on how long the unit can attempt to power down the output without reaching a lower voltage threshold. The UCD92xx implementation deviates slightly from the PMBus standard. According to the PMBus standard, the lower voltage threshold is 12.5% of the output voltage programmed at the time the unit is turned off. In the UCD92xx, the lower voltage threshold, VREF_MIN, is a variable that depends on the measured input voltage, the minimum pulse-width of the power driver, and the switching frequency.

VREF_MIN [V] = VIN [V] * DRIVER_MIN_PULSE [ns] * FREQUENCY_SWITCH [kHz} / 1E6

10.16 (80h) STATUS_MFR_SPECIFIC

The UCD9240 defines the STATUS_MFR_SPECIFIC bits as shown in the following table.

Bit	Name	Description
7	CLF	Analog current monitor fault (paged)
6	FLT	External current fault (paged)
5	CONFIG_INVALID	Invalid PHASE_INFO or GPIO_SEQ_CONFIG
4	CONFIG_CHANGED	PHASE_INFO or GPIO_SEQ_CONFIG changed and a SOFT_RESET or hard reset is required
3	PKGID_MISMATCH	Hardware Package ID does not match firmware
2	HARDCODED_PARMS	PMBus hard-coded defaults were loaded into operating memory due to invalid or empty data flash image
1	SEQ_TIMEOUT	Sequencing timeout waiting for external event (paged)
0	OVF	UCD921x ONLY: Over-voltage Monitor Fault

10.17 (81h) STATUS_FAN_1_2

The UCD9240 only supports one fan, so this read-only command reports the status of fan 1. All Fan_2 status bits will report zero. Note that UCD9220 and UCD921x do not have I/O pins to support a fan so this command is meaningless for those models.

The FAN_1_FAULT bit will be set to 1 when the measured fan speed is less than the value set by the FAN_SPEED_FAULT_LIMIT (0xE8) command for 5 consecutive seconds.

The FAN_1_WARN and SPD_1 bits are not supported.

Bit	Name	Description	Supported
7	FAN_1_FAULT	Fan 1 Fault	Yes
6	FAN_2_FAULT	Fan 2 Fault	No
5	FAN_1_WARN	Fan 1 Warning	No
4	FAN_2_WARN	Fan 2 Warning	No
3	SPD_1	Fan 1 Speed Overridden	No
2	SPD_2	Fan 2 Speed Overridden	No
1	AIRFLOW_FAULT	Airflow Fault	No
0	AIRFLOW_WARN	Airflow Warning	No

10.18 (82h) STATUS_FAN_3_4

The UCD92xx does not support fans 3 or 4. This read-only command always returns 0x00.

10.19 (8Dh) READ_TEMPERATURE_1

This read-only command returns the temperature from a sensor embedded inside the UCD92xx controller.

10.20 (8Eh) READ_TEMPERATURE_2

This read-only command returns the temperature from an external temperature sensor located in or near an output power module. The UCD921x supports a single external temperature sensor, while the UCD9240 and UCD9220 use an external analog multiplexer to cycle through all the external temperature sensors. The PHASE command is used to select which temperature sensor is reported by the READ_TEMPERATURE_2 command. If PHASE = 0xFF, the highest temperature on the selected page is reported.

11 Implementation Details for User Data Commands

11.1 (B0h) USER_DATA_00

This Read/Write Block command provides 32 bytes of space for the user to store information about their particular system.

11.2 (B9h) SYNC_CONFIG (USER_DATA_09)

This Read/Write Block command configures the relative timing between DPWM units and between devices.

Need long description of DPWMs, trigger sources, and Sync_In and Sync_Out pins.

Issuing this command while any DPWM outputs are enabled could cause voltage glitches on the output. It is recommended that all rails be turned off when issuing this command.

This command replaces the SYNC_IN_OUT (E1h) and the SYNC_OFFSET (FBh) commands.

The data for this command is shown in this table.

Byte Number (Write)	Byte Number (Read)	Description
0		CMD = 0xB9
1	0	Length = 9
2	1	Sync Output Pin Source
3	2	Rail 1 Sync Trigger Source
4	3	Rail 1 Sync Delay
5	4	Rail 2 Sync Trigger Source
6	5	Rail 2 Sync Delay
7	6	Rail 3 Sync Trigger Source
8	7	Rail 3 Sync Delay
9	8	Rail 4 Sync Trigger Source
10	9	Rail 4 Sync Delay

Sync Output Pin Source Options

Value	Sync Output Pin Source
0	None. Sync_Out pin is not driven. (*)
1	Sync_Out pin driven by DPWM1
2	Sync_Out pin driven by DPWM2
3	Sync_Out pin driven by DPWM3
4	Sync_Out pin driven by DPWM4
5-255	Invalid

(*) The UCD9244 does not have a Sync_Out pin, so 0 is the only allowed value on a UCD9244.

Note that this numbering is different from the SYNC_OUT setting used in the SYNC_IN_OUT command.

Sync Trigger Source Options

Value	Sync Trigger Source
0	No trigger - not synchronized
1	Triggered by DPWM1
2	Triggered by DPWM2
3	Triggered by DPWM3
4	Triggered by DPWM4
5-254	Invalid
255	Triggered by Sync_In pin

Sync Delay: This is the delay from the trigger source to the start of the DPWM period. It is expressed as a fraction of the switching period on the target rail (not the period of the source DPWM or the Sync_In pin). The unsigned 8-bit value from 0 to 255 corresponds to a fraction from 0.0 to 99.6% of the switching period using the following formula:

Delay%(n)= Sync_Delay_n / 256 * 100%

Validity Checking: These error conditions are tested:

1) Message length other than 9.
2) Sync Output Pin source larger than 4.
3) Sync Trigger Source from 5-254.

Any of these conditions will report a NACK due to INVALID_DATA and will be ignored.

Default Values:

Description	Default Value	Explanation
Sync Output Pin Source	0	Sync_Out pin not driven
Rail 1 Sync Trigger Source	0	Rail 1= Free running with no delay
Rail 1 Sync Delay	0	
Rail 2 Sync Trigger Source	1	Rail 2 = Quarter-period delay after Rail 1
Rail 2 Sync Delay	64	
Rail 3 Sync Trigger Source	2	Rail 3 = Quarter period delay after Rail 2
Rail 3 Sync Delay	64	
Rail 4 Sync Trigger Source	3	Rail 4 = Quarter period delay after Rail 3
Rail 4 Sync Delay	64	

The default values are chosen to spread the DPWM outputs evenly, with quarter-period spacing between rails. Rail 1 is free-running, and acts as a trigger source for Rail 2 after a quarter-period delay. Rail 2 triggers Rail 3 after a quarter-period delay, and Rail 3 triggers Rail 4 after a quarter-period delay.

Add examples. Include section about UCD7242 and interleave.

11.3 (BAh) READ_AUX_ADCS (USER_DATA_10)

This Read Block command reads the four Auxiliary ADC inputs. The UCD92xx combines multiple raw ADC readings together to average out the effect of high-frequency noise. No attempt is made to compensate for gain or offset errors, however.

Each AUX ADC returns a 16-bit unsigned value from 0 to 65535, corresponding to a voltage from 0 to 2.5V from the formula

V(n) = AUX_ADC_n / 65536 * 2.500V

The data returned by this command is shown in this table.

Byte Number (Read)	Description
0	Length = 8
1	AUX_ADC_1 (LSB)
2	AUX_ADC_1 (MSB)
3	AUX_ADC_2 (LSB)
4	AUX_ADC_2 (MSB)
5	AUX_ADC_3 (LSB)
6	AUX_ADC_3 (MSB)
7	AUX_ADC_4 (LSB)
8	AUX_ADC_4 (MSB)

11.4 (BBh) VID_CONFIG (USER_DATA_11)

This Read/Write Block command configures the VID interface and the VID code table. It is only supported on the UCD9222 and UCD9244 variants. Each page (rail) may have a unique configuration setting.

The data for this command is shown in this table.

Byte Number (Write)	Byte Number (Read)	Description
0		CMD = 0xBB
1	0	Length = 5, 6, 7, or 9
2	1	VID Format (Number of bits in VID Code)
3	2	VID Vout Low (LSB)
4	3	VID Vout Low (MSB)
5	4	VID Vout High (LSB)
6	5	VID Vout High (MSB)
7	6	VID Code Init (LSB)
8	7	Reserved (VID Code Init MSB)
9	8	VID Lockout Interval (LSB)
10	9	VID Lockout Interval (MSB)

VID Format: Supported values

Format	Table Size	Description
0x00	n/a	Ignore VID input pins and commands. Use the PMBus OPERATION command to select between the VOUT_COMMAND, VOUT_MARGIN_HIGH, or VOUT_MARGIN_LOW setting as the target voltage.
0x04	16	4-bit Parallel VID Mode Use the 4 VID input pins for each rail to create a 4-bit VID code. The VID pins are polled periodically. The VIDxS inputs act as the most-significant data bit.

0x06	64	6-bit Serial VID Mode Use the 4 VID input pins for each rail to create a 6-bit VID code. The VIDxS inputs trigger the reading of the other VID data pins. The level of the VIDxS signal selects which half-word to store the three data bits.
0x07	128	7-bit Parallel Mode Combines the 4 VID input pins for the designated rail plus 3 VID input pins from the next rail to create a 7-bit VID code. The VID pins are polled periodically. This option is only allowed on Rails 1 and 3. When Rail 1 (or 3) is set for 7-bit mode, then Rail 2 (or 4) may not use the VID pins.
0x08 0x18	256 256	8-bit I2C/PMBus Mode Use the VID_CODE_RAILx commands to issue an 8-bit VID code via PMBus. The VID input pins are ignored. 8-bit Parallel Mode Combines the 4 VID input pins for the designated rail plus 4 VID input pins from the next rail to create an 8-bit VID code. The VID pins are polled periodically. This option is only allowed on Rails 1 and 3. When Rail 1 (or 3) is set for 8-bit parallel mode, then Rail 2 (or 4) may not use the VID pins.

The default value of VID Format is 0.

VID Vout Low: This is the output voltage that will be commanded when VID code 0 is sent to the device. It is a 2-byte value that is stored in LINEAR16 format with an exponent of -12, like other output voltage commands. The default value is 0.0V.

VID Vout High: This is the output voltage that will be commanded when the highest VID code (15, 63, or 255) is sent to the device. It is a 2-byte value that is stored in LINEAR16 format with an exponent of -12, like other output voltage commands. The default value is 0.0V.

VID Code Init: A one-byte value that sets the initial VID Code value that is used at power-up, after a soft reset, or after a RESTORE_DEFAULTS_ALL command. The allowable values range from 0 to ((2^VID_Format)-1). The default value is 0.

VID Lockout Interval: A two-byte signed value that sets the duration of the interval during which the VID inputs are to be ignored. This can be used to ignore stray signals on the VID inputs while the output voltage is turning on, to prevent them from corrupting the VID voltage setting. The lockout interval timer starts when the output voltage reaches the top of the soft-start ramp. Positive values range from 1 to 32767 ms, with 1 ms resolution. A value of 0 will ignore the VID inputs during the start ramp and re-enable them immediately at the top of the ramp. Negative values bypass the lockout mechanism, allowing the VID inputs to remain active all the time regardless of the output voltage state. The default value is 0.

Validity Checking: Only the VID Format and VID Code Init fields undergo range checking. VID Format values other than the ones shown in the table above will return a NACK due to Invalid Data and will be rejected. The same response will occur for VID Code Init values that exceed the range allowed by the VID Format setting, except when the VID Format is set to 0.

11.5 (BCh) VID_CODE_RAIL1 (USER_DATA_12)

This Read/Write Byte command is used to set the output voltage for output Rail 1 (PAGE 0) via PMBus. This command uses a VID format that is an alternative to the standard PMBus VOUT_COMMAND command. This command is only supported on the UCD9222 and UCD9244.

The commanded output voltage reference is set according to this formula:

*Vref_cmd = (VID_CODE *ID_Slope) + VID_Offset,*

where

VID_Slope = (VID_Vout_High – VID_Vout_Low) / ((2^VID_Format) -1),

and

VID_Offset = VID_Vout_Low.

The VID_Vout_High, VID_Vout_Low, and VID_Format values are set by the VID_CONFIG command (0xBB).

The data for this command is a single byte, with values from 0 to 255.

The parameter is not stored in the Default Store in Data Flash. Its default value at power-on is 255.

This VID code only applies when operating in 8-bit VID mode. If this command is issued when VID_Format is not in 8-bit mode, the value will be stored but will not affect the output voltage until the VID_Format is set to 8-bit mode.

This command is unique in that it does not use the PMBus PAGE command to select its output rail. For most PMBus commands, the host first sends a PAGE command to select an output rail, followed by one or more commands to set or read parameters for the selected rail. In a system that has multiple masters, it would be possible for one master to insert a command between the commands from a different master, thus corrupting the settings on both master's rails. To avoid these conflicts, each output rail has a unique VID_CODE_RAILx command that both selects the rail and sets its VID voltage code.

Note: this command uses the R/W Byte data format rather than the R/W Block format specified in the PMBus standard for the USER_DATA_12 command 0xBC. This violation of the PMBus standard was necessary to meet the needs of a key customer whose host could only issue commands in R/W Byte format.

11.6 (BDh) VID_CODE_RAIL2 (USER_DATA_13)

This Read/Write Byte command is used to set the VID code for output Rail 2 (PAGE 1) via PMBus. See the VID_CODE_RAIL1 command (0xBC) for details.

11.7 (BEh) VID_CODE_RAIL3 (USER_DATA_14)

This Read/Write Byte command is used to set the VID code for output Rail 3 (PAGE 2) via PMBus. See the VID_CODE_RAIL1 command (0xBC) for details.

11.8 (BFh) VID_CODE_RAIL4 (USER_DATA_15)

This Read/Write Byte command is used to set the VID code for output Rail 4 (PAGE 3) via PMBus. See the VID_CODE_RAIL1 command (0xBC) for details.

12 Implementation Details for Manufacturer-Specific Commands

12.1 (D0h) SEQ_TIMEOUT (MFR_SPECIFIC_00)

This Read/Write Word command defines a window after a page has reached its POWER_GOOD_ON threshold during which an external event must occur. If the event does not occur in the time specified by this command, the rail is shut down and an error is posted. The event is configured using the GPIO_SEQ_CONFIG command to select and configure a pin as the input source.

A timeout value of 0 will disable the timeout monitoring function.

The units are msec.

12.2 (D1h) VOUT_CAL_MONITOR (MFR_SPECIFIC_01)

This Read/Write Word command is used to apply a fixed offset voltage to the output voltage measured by the device and reported by the READ_VOUT command. It is typically used by the PMBus device manufacturer to calibrate a device in the factory.

The VOUT_CAL_MONITOR has two data bytes formatted as a twos-complement binary integer. The effect on this command depends on the settings of the VOUT_MODE command.

12.3 (D2h) PHASE_INFO (MFR_SPECIFIC_02)

This Read/Write Block command is used to define the relationship between the DPWM units (the phases) and the output rails (the pages). It is described in detail in section 13 below.

12.4 (D3h) VIN_SCALE_MONITOR (MFR_SPECIFIC_03)

This Read/Write Word command represents the gain of the external sensor measuring VIN. Typically this will be a passive resistor divider network.

This command has two data bytes formatted in the Linear Data format.

The units are (volts at the VIN ADC input pin) per (volt at the power stage input). With a passive resistor divider network, the value will be between 0.0 and +1.0.

12.5 (D4h) CLA_BANK (MFR_SPECIFIC_04)

See section 15.1 below.

12.6 (D5h) CLA_GAINS (MFR_SPECIFIC_05)

See section 15.2 below.

12.7 (D6h) PAGE_ISOLATED (MFR_SPECIFIC_06) (UCD9220, '9240, '9211, '9212)

This Read/Write Byte command configures the way that the page controller behaves during soft-stop.

When PAGE_ISOLATED = 1, the output rail is assumed to be isolated (i.e. not bridged to any other voltage source). In this case the controller will keep synchronous rectification active during the soft-stop sequence so that the output voltage ramps down smoothly.

When PAGE_ISOLATED = 0, the output rail is assumed to be connected to another voltage source. To avoid sinking excess current when the output voltage ramps below that of the other source, synchronous rectification will be disabled during soft-stop.

Note: This same command code was re-used for a different command on the UCD9224, '9246, '9248, and '9244 products.

12.8 (D6h) DRIVER_CONFIG (MFR_SPECIFIC_06) (UCD9224, '9246, '9248, '9244)

This Read/Write Byte command determines how the controller interacts with the driver for the selected page. Three characteristics of the interactions are defined:

- The polarity of the driver fault feedback signal
- How an existing fault should be handled at startup.
- How the DPWM and SRE outputs should be handled at shutdown.

This command has one data byte whose contents are described in the table below. The default setting is 0x00.

Bit(s)	Name	Description
7:3	*unused*	*Reserved for future use.*
2	Polarity	FLT interrupt polarity. 0=active high, 1=active low
1	FLT Abort Start	FLT prevents start attempt. 0=Assume DPWM will clear FLT, 1=Abort
0	Tri-state	DPWM action on shutdown. 0=Drive outputs low, 1=Tri-state outputs

Note: This same command code was previously used for a different command on the UCD9220, '9240, '9211, and '9212 products. Those products behave as if DRIVER_CONFIG was set to 0x00.

12.9 (D7h) EADC_SAMPLE_TRIGGER (MFR_SPECIFIC_07)

This Read/Write Word command configures the time point where the EADC samples the output feedback voltage used by Control Law Acceleration hardware. The pre-trigger time is measured backward from the rising edge of the DPWM output signal to the end of the EADC sampling window. The EADC sampling window opens ~32 ns before it closes.

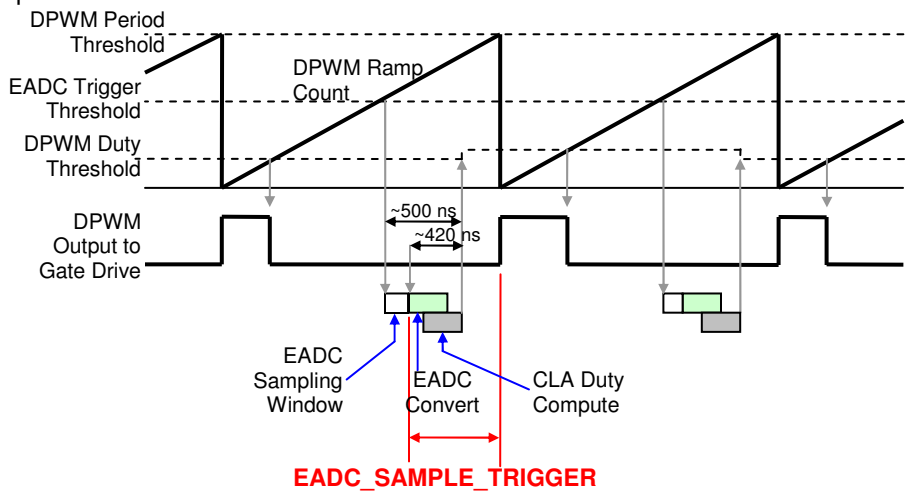

Figure 1: EADC Sample Trigger Timing

When configuring the EADC triggering time, there are three main considerations:

1) The EADC sampling window should avoid PWM switching edges, as they may introduce noise into the feedback signal. Switching edges from the selected page or from other pages may have an effect and should be avoided.

2) The EADC sampling time must allow sufficient time (~224 ns) for the EADC to make its measurement and the CLA to complete its control calculations for the next period. If the EADC sample time is configured to be too close to the start of the next switching period, the control calculations will not be complete in time to be used for the next period. This will result in an additional full period of delay in the control loop.
3) The EADC sampling should be as late as possible to minimize the computational delay and optimize the phase margin of the control loop, subject to the other two constraints.

This command has two data bytes formatted in the Linear11 Data format. The units are nanoseconds. The sampling point is adjustable in multiples of 16 ns. The default value is 224 ns.

Validity Checking: Valid values must be positive numbers in Linear11 format and must be less 32767 ns.

If the value is less than the minimum CLA computation time (~224 ns), then one full switching period will be added to the requested value. This reflects the additional delay that the control loop will see when it tries sampling too late in the switching period for the CLA to finish its calculations. If the requested setting is more than one switching period plus the minimum computation time, the effective value will be the modulo value of one switching period. These limits are dependent upon the switching period, and will be re-calculated if the switching period is changed using the FREQUENCY_SWITCH command or by a RESTORE_DEFAULT_ALL command. The order in which the EADC_SAMPLE_TRIGGER and FREQUENCY_SWITCH commands are issued is not important.

Timing: Some of the timings shown in this section are described as approximate. This does not mean that the timing of the hardware is variable, only that it has not been finalized at the time this document was written. The timing will not vary for a particular unit or for units with the same revision of silicon. The values shown here are exact values when the UCD92xx's internal master oscillator is operating at 250 MHz. Later revisions of the UCD92xx may use a different clock speed for better performance. While a faster clock may allow the user to set a smaller minimum value for EADC_SAMPLE_TRIGGER, in most cases settings chosen with a slower clock will run exactly the same with the faster clock. If so, the lower limit in the EADC_SAMPLE_TRIGGER command will automatically reflect the faster internal clock and will require no change from the user.

12.10 (D8h) ACTIVATE_CLA_BANK (MFR_SPECIFIC_08)

See section 15.3 below.

12.11 (D9h) ROM_MODE (MFR_SPECIFIC_09)

This Send Byte command sends the system into ROM mode. Issue this command before attempting to download new firmware to the device.

ROM, PFlash, and Integrity

After a reset or power-up, the UCD92xx device starts executing a boot loader algorithm stored in ROM. This boot loader supports a small set of specialty commands (not listed in this document) to allow device testing and updating of the firmware in Program Flash. After performing a few simple wakeup diagnostics, the boot ROM scans the Program Flash to validate its integrity. If the integrity is good, the processor reconfigures itself to disable the ROM and execute from Program Flash.

When a ROM_MODE or SOFT_RESET(see section 12.13 below) command is issued, the ROM is re-enabled and processor is reset. For a SOFT_RESET command, the Program Flash Integrity test will be successful and the Program Flash firmware will start executing automatically.

To prevent this automatic execution, the ROM_MODE command erases the Program Flash Integrity Word before resetting the processor. Once the Integrity Word has been erased, the Flash firmware must

be reloaded using the Fusion Digital Power tools. Until then, the existing Flash firmware may only be executed by issuing the EXECUTE_FLASH command (see section 12.34 below).

12.12 (DAh) USER_RAM_00 (MFR_SPECIFIC_10)

This Read/Write Byte command allows the user to read/write a byte value to RAM in the device. This RAM value will be reset to a known value (0) when the controller is reset. By monitoring this value, the user will be able to tell that the controller has been reset during operation.

Note that this parameter is not stored to non-volatile Default Store memory when the STORE_DEFAULT_ALL command is issued.

12.13 (DBh) SOFT_RESET (MFR_SPECIFIC_11)

This Write-only Send Byte command restarts the controller firmware. Any active voltage outputs are turned off before the firmware restarts.

This is the same as the ROM_MODE command except that it does not modify the Program Flash Integrity Word. See section 12.11 above for details about ROM and the Program Flash Integrity Word.

12.14 (DCh) IIN_SCALE_MONITOR (MFR_SPECIFIC_12)

This Read/Write Word command is used to set the ratio of the input current to the voltage at the input current sense pin to the input current. For devices that use a fixed current sense resistor and a sense amplifier, the value of IIN_SCALE_MONITOR will be equal to the sense resistance multiplied by the amplifier gain.
This command has two data bytes formatted in the Linear11 format. The units of this command are ohms (V/A). Valid values are in the range from -2.0 to + 1.99994 V/A. The default value is 0 ohms.

Note that because the internal calculation for the READ_IIN and READ_PIN commands use division, very small values of IIN_SCALE_MONITOR may result in numeric overflow. A value of zero ohms is given special handling and will return zero amperes, not infinity.

12.15 (DDh) THERMAL_COEFF (MFR_SPECIFIC_13)

This command allows for temperature compensation of the current sense element. The temperature sensed by the external temperature sensor on each phase is used to compensate for temperature-dependent changes in the current sense voltage. For example, when current is sensed by the voltage drop across the DC resistance of a copper inductor, the resistance R is

R(Temp) = R(RefTemp) * (1 + alpha * (Temp - TempRef)),
where alpha = 0.393%/°C for copper and TempRef = 20 degrees C.

Examples:

Current Sense Circuit	**Recommended Value**
DC resistance of a copper inductor.	0.393 %/°C (thermal coefficient of resistance for copper)
Inductor DCR with temperature-compensated sense circuit.	0.0
Series Sense Resistor	Thermal coefficient for sense resistor, typically very close to zero.
Series FET resistance	Thermal coefficient for FET, typically negative.

This command has two data bytes formatted in the LINEAR11 Linear Data format. The units are percent per Celsius degree. Allowable values range from -50 to +50 %/°C. The default value is 0.393 %/°C, which is the thermal coefficient of copper.

12.16 (DEh) PHASE_ENABLE (MFR_SPECIFIC_14)

This Read/Write Byte command allows specific phases controlled by a page to be enabled or disabled. When a phase is disabled, its DPWM output is put into high-impedance mode and synchronous rectification is disabled.

The command has one data byte. Each bit corresponds to one phase controlled by the page most recently chosen by the PMBus PAGE command. Phases start numbering from zero and number up to one less than the number of phases configured using the PHASE_INFO command (see section 13 below).

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Phase	7	6	5	4	3	2	1	0

Examples: For a page that has four phases, valid phases are numbered 0 through 3.
Setting PHASE_ENABLE = [0000 1011] would disable phase 2 while leaving phases 0, 1, and 3 enabled.
Setting PHASE_ENABLE = [0000 0010] would enable phase 1 and disable all other phases.

Enabling a single phase at a time may be useful when calibrating the IOUT_CAL_OFFSET and IOUT_CAL_GAIN values.

PHASE_ENABLE may be issued with the outputs turned on or off. If the PHASE_ENABLE command turns off all the phases on a page that is turned on, the output voltage will drop to zero and under-voltage faults will be reported. If the PHASE_ENABLE command is issued to a page that is turned off, it will have no effect until that output is turned on using the OPERATION command or the CONTROL pin.

The parameter is not stored in the Default Store in Data Flash.

12.17 (DFh) DRIVER_MIN_PULSE (MFR_SPECIFIC_15)

When very short PWM pulses are fed into some drivers, they misbehave by dropping output pulses or generating shoot-through conditions. This Read/Write Byte command defines the minimum pulse width (in nanoseconds) that the driver is capable of supporting reliably.

This command has two data bytes formatted in the Linear11 Data format. The units are nanoseconds.

The lowest output voltage that can be supported, VREF_MIN, is a variable that depends on the measured input voltage, the minimum pulse-width of the power driver, and the switching frequency.

VREF_MIN [V] = VIN [V] * DRIVER_MIN_PULSE [ns] * FREQUENCY_SWITCH [kHz] / 1E6

During soft-start and soft-stop operation and when operating as a slave in tracking mode, the outputs will be disabled while the dynamic set-point voltage is less than VREF_MIN.

12.18 (E0h) MIN_DUTY (MFR_SPECIFIC_16)

This Read/Write Word command set the minimum duty cycle, from -99 to 99 percent, of the unit's power conversion stage. A large negative value is highly recommended for this value to prevent truncation of internal calculations which will result in large output voltage excursions. Use DRIVER_MIN_PULSE to limit the minimum driver pulse width instead of this command. The default value is -50.

This command has two data bytes formatted in the LINEAR11 Linear Data format.

12.19 (E1h) SYNC_IN_OUT (MFR_SPECIFIC_17)

This Read/Write Word command configures the DPWM hardware to allow the PWM clocks for multiple UCD92xx devices to be synchronized together.

The data for this command is two bytes, as shown in this table.

Byte Number (Write)	Byte Number (Read)	Description
0		CMD = E1
1	0	SYNC_IN: Bit Mask for Slave Sync Inputs
2	1	SYNC_OUT: Page Number for Master Sync Output

SYNC_IN:
The SYNC_IN byte contains a bit mask, where each bit corresponds to one page.

Bit	7	6	5	4	3	2	1	0
Page					3	2	1	0

Setting a bit to 1 makes the corresponding page become a slave to the input sync signal. Any number of pages may be configured as slaves, from zero pages up to the maximum number of pages configured by the PHASE_INFO command. When writing to SYNC_IN_OUT, if SYNC_IN bits are set for page numbers higher than the maximum page configured by PHASE_INFO, the SYNC_IN_OUT command will report a NACK due to Invalid Data.

The default value of SYNC_IN is 00, indicating that no slaves are active.

SYNC_OUT:
The SYNC_OUT byte selects one of the pages to be used to drive the Master Sync Out signal. A value of 0xFF will select no master and will disable the master sync output. When writing to SYNC_IN_OUT, if SYNC_OUT is set for a page number higher than the maximum page configured by PHASE_INFO, the SYNC_IN_OUT command will report a NACK due to Invalid Data.

The default value of SYNC_OUT is 0xFF, indicating that the master sync output is disabled.

It is acceptable to configure a page to be both a master and a slave at the same time. The Sync In signals used to synchronize all the slaves on a device is distinct from the Master Sync Out signal, which is used to synchronize slave on an external device.

When an external sync is configured for a rail, the frame timer resets at the sync signal. (If the period expires before the sync arrives, the timer will reset at that point and the sync will then cause a second reset.) There are some implications of using an external sync which should be considered.

- The sync input will adjust the default rail-to-rail timing (described in section 1.1) initially set by the device at power-up; this can not be restored without a soft-reset or a cycling of the power.
- The time from the EADC Sample Trigger point to the sync input must be sufficient for the A/D conversion and the control calculation to be made. If the calculation does not complete in time, an entire switching cycle delay time will be added to the control loop. Without sync, EADC_SAMPLE_TRIGGER has been optimized to allow 224 nsec for the ADC conversion and the control calculation. This number must be increased by the maximum amount that the sync input may shorten the period. For example, if a rail is nominally configured to run at 500 kHz and

the master sync input may be as fast as 510 kHz, the difference in Ts between the two frequencies is 39.2 nsec. EADC_SAMPLE_TRIGGER should be increased to 264 nsec.

- With a single phase rail, the device has the ability to sync to an external input that is either slightly faster or slower than the defined FREQUENCY_SWITCH. Faster frequencies will leave the on time unaffected while shortening the period; slower frequencies stretch the period by increasing the on time. Either adjustment will cause a small voltage excursion which the integrator in the controller will quickly absorb. In a multi-phase rail however, the slower frequency will increase only the A-phase output of the DPWM. This will result in a current imbalance. The current balance controller runs at a much slower bandwidth and the mismatch in phase currents has a high probability of causing IOUT_OC_FAULT events. To prevent this, the FREQUENCY_SWITCH value should always be set slower than sync master.

12.20 (E2h) PARM_INFO (MFR_SPECIFIC_18)

This Read/Write Block command is used to configure the parameters used by the PARM_VALUE command.

The PARM_INFO command updates four variables that are needed to issue a generic read/write of RAM or hardware registers. The four variables are parm_index, parm_offset, parm_count and parm_size and are described below.

Parm_index – Index for base address
0 = RAM
1 = Hardware Peripherals
2 = Constants in Data Flash (Read Only)
3 = Constants in Program Flash (Read Only)
4 = Program in Program Flash (Read Only)
Option 4 may be disabled in firmware for security reasons.
Parm_offset – offset from the base address selected by parm_base.
Parm_count – number of elements to read or write
Parm_size – the size of each element in bytes. (Valid values are 1, 2 or 4).

PARM_INFO and PARM_VALUE are combined to provide a method for reading or writing to any RAM address or hardware register. A map file specific to the firmware release may be required to determine the offset for a particular RAM variable, since variables may be in different locations for each release.

Byte Number (Write)	Byte Number (Read)	Description
0		CMD = E2
1	0	BYTE_COUNT=5
2	1	Index
3	2	Offset low byte
4	3	Offset high byte
5	4	Count
6	5	Size

12.21 (E3h) PARM_VALUE (MFR_SPECIFIC_19)

This Read/Write Block command is used to read and write to RAM addresses or hardware peripheral registers. This command assumes that the PARM_INFO command has be previously run to set up the parm_base, parm_offset, parm_count and parm_size variables as needed.

12.22 (E4h) TEMPERATURE_CAL_GAIN (MFR_SPECIFIC_20)

This Read/Write Word command sets the gain calibration for the external sensors used by the READ_TEMPERATURE_2 command. Each external temperature sensor (typically one per power output phase) has its own calibration setting.

This command has two data bytes formatted in the Linear11 Data format. The units are Celsius degrees per volt.

12.23 (E5h) TEMPERATURE_CAL_OFFSET (MFR_SPECIFIC_21)

This Read/Write Word command sets the offset calibration for the external sensors used by the READ_TEMPERATURE_2 command. Each external temperature sensor (typically one per power output phase) has its own calibration setting.

This command has two data bytes formatted in the Linear11 Data format. The units are degrees Celsius.

12.24 (E6h) TRACKING_SOURCE (MFR_SPECIFIC_22)

This Read/Write Byte command configures the output to track a reference source. The tracking source may be an external voltage connected to the Vtrack input pin (External Tracking), or it may be from another rail controlled by the same device (Internal Tracking). Internal tracking uses the value from the master rail's setpoint DAC to drive both the master and the slave. This generally keeps both rails tightly matched to each other and makes the slave more immune to noise on the master rail.

Setting	Tracking Mode	Tracking Reference Source
-128 to -1	Disabled	--
0	Internal	Page 0 setpoint DAC
1	Internal	Page 1 setpoint DAC
2	Internal	Page 2 setpoint DAC
3	Internal	Page 3 setpoint DAC
4 to 127	External	Vtrack Input Pin
128 to 255	Disabled	--

The slave rail output voltage will be proportional to the selected source voltage. The TRACKING_SCALE_MONITOR command is used to set the ratio between the output voltage and the tracking source.

This command has one data byte. The default value is 255 (Tracking Disabled).

12.25 (E7h) TRACKING_SCALE_MONITOR (MFR_SPECIFIC_23)

This Read/Write Word command specifies the ratio between the output voltage on the selected page and the voltage tracking input signal.

Volts_Out = Volts_Track_In * TRACKING_SCALE_MONITOR

This command has two data bytes formatted in the Linear11 Data format. The units are volts out per volt in.

12.26 (E8h) FAN_SPEED_FAULT_LIMIT (MFR_SPECIFIC_24)

This Read/Write Word command specifies the speed at which the fan speed control will report a fault. If the fan speed is below this limit for 5 consecutive seconds, the controller will assert the PMBus ALERT signal and set the fault bits in the STATUS_FANS_1_2 and STATUS_WORD registers.

This command has two data bytes formatted in the Linear11 Data format. The units are RPM.

12.27 (E9h) LOGGED_PEAKS (MFR_SPECIFIC_25)

This Read/Write Block command returns the maximum temperatures and maximum currents seen during operation and logged into non-volatile memory. Provisions exist to reset this non-volatile logged information.

This command returns a binary array in the order shown in Table 3. Each temperature is one unsigned byte that contains the temperature in degrees C. Each current is two bytes in Linear11 format.

The internal temperature sensor reading is the same one reported by the READ_TEMPERATURE_1 command. The external temperature sensors are the ones reported by the READ_TEMPERATURE_2 command. The current measurements are the total of all phase currents for each page.

Table 3: LOGGED_PEAKS Data Format

Byte Number (Write)	Byte Number (Read)	Description
0		CMD = E9
1	0	BYTE_COUNT = 17
2	1	Internal Temperature Sensor
3	2	External Temperature Sensor for DPWM Output 1A
4	3	External Temperature Sensor for DPWM Output 1B
5	4	External Temperature Sensor for DPWM Output 2A
6	5	External Temperature Sensor for DPWM Output 2B
7	6	External Temperature Sensor for DPWM Output 3A
8	7	External Temperature Sensor for DPWM Output 3B
9	8	External Temperature Sensor for DPWM Output 4A
10	9	External Temperature Sensor for DPWM Output 4B
11	10	Page 0 Current (low byte)
12	11	Page 0 Current (high byte)
13	12	Page 1 Current (low byte)
14	13	Page 1 Current (high byte)
15	14	Page 2 Current (low byte)
16	15	Page 2 Current (high byte)
17	16	Page 3 Current (low byte)
18	17	Page 3 Current (high byte)

Unused values will be reported as 0℃ or 0A.

Clearing the Log: Writing a block whose data bytes are all 0x00 will reset all logged entries to 0. Non-zero values in any data byte will NACK due to Invalid Data.

FLASH Memory Management: To reduce unnecessary stress on the FLASH memory, the peak values are store in volatile RAM memory and only written to FLASH memory under certain conditions:

a) If at least one temperature or current value exceeds both its previously logged maximum value and either its warning or fault limit, a 30-second timer is started. At the end of this timer interval, the values are copied from RAM to FLASH. During a transient event the peak value may reach several new maximums in rapid succession; the 30-second timer combines them together for a single write operation.
b) If a new fault is recorded in the fault log (see section 12.28), both the peak log and the fault log are written to FLASH.

At power-up, the peak values in RAM are initialized from the peaks previously stored in FLASH. The current peaks in RAM are only updated when the output current exceeds the IOUT_OC fault or warning limits.

Peak temperatures are handled somewhat differently. The peaks stored in RAM are the highest values seen since the RAM was initialized at power-up. If no faults or warnings occur to cause a write to FLASH, the peak temperature values will not be stored and will be reset at the next power-up.

12.28 (EAh) LOGGED_FAULTS (MFR_SPECIFIC_26)

This Read/Write Block command reports a history of all faults that have ever been reported and logged into non-volatile memory.

This command returns a binary array in the order shown in Table 4. The bit definitions for common faults are shown in Table 5. The bit definitions for page-dependent faults are shown in Table 6. Log entries for unused pages will be reported as 0x0000.

Table 4: LOGGED_FAULTS Data Format

Byte Number (Write)	Byte Number (Read)	Description
0		CMD = EA
1	0	BYTE_COUNT = 9
2	1	Non-Paged Faults
3	4	Page 0 Variable Faults - Low Byte
4	3	Page 0 Variable Faults - High Byte
5	4	Page 1 Variable Faults - Low Byte
6	5	Page 1 Variable Faults - High Byte
7	6	Page 2 Variable Faults - Low Byte
8	7	Page 2 Variable Faults - High Byte
9	8	Page 3 Variable Faults - Low Byte
10	9	Page 3 Variable Faults - High Byte

Note that the log include pages 0 through 3, even when fewer than four pages have been selected by the PHASE_INFO command.

Table 5: Non-Paged Fault Log Bit Definitions

Bit	Description
0	LOG_NOT_EMPTY
1	Reserved
2	Reserved
3	Reserved for IIN_OC Fault
4	Fan 1 Fault
5	
6	
7	

Bit 0: By examining this single bit, a host can determine whether any page-dependent variable faults have occurred. A value of 0 in this bit indicates that all of the page-dependent fault log entries are zero and need not be read. A value of 1 in this bit indicates that one or more of the Page-Dependent Variable Faults has occurred. In that case, the host must examine all the page-dependent log entries to determine which ones are pending.

Table 6: Page-Dependent Variable Fault Log Bit Definitions

Bit	Description
0	VOUT_OV Fault
1	VOUT_UV Fault
2	IOUT_OC Fault
3	IOUT_UC Fault
4	Reserved for Current Share Fault
5	Reserved for POUT_OP Fault
6	TEMPERATURE_OT Fault
7	Reserved for TEMPERATURE_UT Fault
8	FLT Faults
9	CLF Faults
10	TON_MAX Fault
11	SEQ_TIMEOUT Fault
12	VIN_OV Fault
13	VIN_UV Fault
14	
15	

Clearing the Log: Writing a block whose data bytes are all 0x00 will reset all logged entries to 0. Non-zero values in any data byte will NACK due to Invalid Data.

12.29 (EBh) LIGHT_LOAD_LIMIT_HIGH (MFR_SPECIFIC_27)

The LIGHT_LOAD_LIMIT_HIGH (EBh) and LIGHT_LOAD_LIMIT_LOW (F6h) commands set the output current levels at which the controller will switch between Normal and Light Load modes of operation. When the output current is below the LOW limit, the controller may disable one or more phase outputs, or may change the control law gains. When the output current is above the HIGH limit, the controller may re-enable any phase outputs that were disabled in Light Load mode, or may change the control law gains. The behavior of the controller is determined by the setting of the LIGHT_LOAD_CONFIG command (EDh).

It is recommended that LIGHT_LOAD_LIMIT_HIGH be set higher than LIGHT_LOAD_LIMIT_LOW by an amount large enough to prevent ripple and noise in the current sense circuit from causing frequent toggling between Light Load and Normal modes. This will minimize transients that may occur during the mode changes.

This command has two data bytes formatted in the Linear11 Data format. The units are amperes. The default value is 0 Amps.

12.30 (ECh) DEVICE_ID / ROM_VER (MFR_SPECIFIC_28)

When the device is executing in FLASH mode, this command returns an ASCII string identical to the DEVICE_ID command (see section 12.35 below for details). When the device is executing in ROM mode, this command returns a four-byte binary string with the ROM version number.

Most PMBus commands are not supported by the ROM and may cause the PMBus peripheral in the device to hang if issued while in ROM mode. This command can be safely issued to determine whether the device is executing in ROM mode or in FLASH mode.

12.31 (EDh) LIGHT_LOAD_CONFIG (MFR_SPECIFIC_29)

This Read/Write Byte command configures the way that the controller for the selected page behaves under light current loads. Whenever the output current is less than the value specified by the LIGHT_LOAD_LIMIT command (EBh), the controller will enter Light Load mode. While in Light Load mode, the controller may disable one or more phase outputs, and/or may use a different set of control law gains.

This command has one data byte whose contents are described in Table 7. The default setting is 0x00.

Table 7: LIGHT_LOAD_CONFIG Bit Definitions

Bit(s)	Description
0	**Manual/Automatic CLA Gain Control** 0 = Unit does not automatically switch between normal load and light load settings. Bit 1 selects between Normal and Light Load CLA Gain setting. 1 = Unit automatically switches between normal load and light load CLA Gain settings.
1	**CLA Gain Control** 0 = Use Normal Load settings, regulating using control gains from CLA bank 1. 1 = Use Light Load settings, regulating using CLA bank 2 gains
2	**Manual/Automatic Phase Control** 0 = Unit does not automatically switch between normal load and light load settings. Bit 3 selects between Normal and Light Load setting of Phase Control. 1 = Unit automatically switches between normal load and light load Phase Control settings.
3	**Phase Control** 0 = Use Normal Load settings, with all phases enabled. 1 = Use Light Load settings, with one or more phases disabled.
4:6	**Number of Light Load Phases** Selects the number of phases that will be enabled while the controller is running in Light Load mode. 000 = 1 Phase Enabled 001 = 2 Phases Enabled 010 = 3 Phases Enabled 011 = 4 Phases Enabled 100 = 5 Phases Enabled 101 = 6 Phases Enabled 110 = 7 Phases Enabled 111 = 8 Phases Enabled
7	Reserved

Validity Checking: If Number of Light Load Phases is set to be larger than the number of phases available for the selected page, an Invalid Data NACK will be reported and the value will be rejected.

12.32 (EEh) PREBIAS_OFFSET (MFR_SPECIFIC_30)

This Read/Write Word command, along with the PREBIAS_GAIN command, allows the controller to be fine-tuned to compensate for non-ideal power stages during soft start with a non-zero starting voltage on the selected page.

Background: When the output voltage is turned on during soft start, the starting voltage may not always be zero volts. This prebias voltage may be non-zero when multiple power supplies are connected in parallel, or when the output capacitor still holds some residual charge.

In an ideal buck regulator, the nominal duty cycle is given by the formula

Duty_nominal = Vout / Vin.

Real power stages often differ from the ideal case. For example, slow gate drivers will yield PWM output pulses with sloped edges. The effective width of these sloped pulses may be wider than the pulses sent from the controller output, resulting in an output voltage higher than ideal.

In other cases, there may be losses in the power stage output path. These losses will cause the output voltage to be less than the ideal voltage. These losses may be dependent on the load current.

Normally the controller will automatically compensate for any inaccuracies or losses by adjusting the duty cycle to make the output voltage match the commanded voltage. In these cases, the duty cycle will not match the ideal value, but that will not be a problem.

However, when the controller is first turned on at a non-zero voltage, it has not yet had time to compensate for any inaccuracies in the power stage. An incorrect starting duty cycle may cause the output voltage to glitch up or down until the controller has had time to adjust the duty cycle for the correct voltage. In such cases, it is useful to have a reasonably accurate estimate of the duty cycle required when first turning on. The PREBIAS_OFFSET and PREBIAS_GAIN settings are available for fine-tuning the starting duty cycle estimate.

At the beginning of the soft start ramp up, the controller measures the output voltage and the input supply voltage. It then calculates an initial duty cycle to preload the controller using this formula:

Duty_preload = ((Vout_prebias / Vin) + PREBIAS_OFFSET) * PREBIAS_GAIN

Note that in the default case where PREBIAS_OFFSET = 0 and PREBIAS_GAIN = 1.0 that this formula simplifies to the ideal case where duty = Vout/Vin.

This preload duty cycle is only used for initialization during soft start. It has no effect during normal operation. It also has no effect when the prebias voltage is so small that PWM pulse width is less than the minimum pulse width set by the DRIVER_MIN_PULSE (DFh) command.

Format: This command has two data bytes formatted in the Linear11 Data format. The units are percent duty cycle. The default value is 0.0%. Typical values are likely to be in the range from -2% to +2%. The command allows values from -100% to +99.9%.

12.33 (EFh) PREBIAS_GAIN (MFR_SPECIFIC_31)

This Read/Write Word command, along with the PREBIAS_OFFSET command, allows the controller to be fine-tuned to compensate for non-ideal power stages during soft start with a non-zero starting voltage.

This command has two data bytes formatted in the Linear11 Data format. The units are dimensionless [V/V]. The default value is 1.0. Typical values are likely to be in the range from 0.8 to 1.2. The command accepts values from 0.0 to +3.996.

12.34 (F0h) EXECUTE_FLASH (MFR_SPECIFIC_32)

If the device is in ROM mode, this command starts the device executing in FLASH mode. If the device is already in FLASH mode, the command has no effect.

12.35 (F1h) MFR_SETUP_PASSWORD (MFR_SPECIFIC_33)

This Read/Write Block command configures the password used to secure the unit against unauthorized modification of its settings.

This command may only be issued after security has been temporarily disabled by sending the proper old password using the DISABLE_SECURITY (F2) command.

The data is a 6-byte binary string containing the new password. The new password can be stored in nonvolatile memory by issuing a STORE_DEFAULT_ALL command.

After the new password has been set to any value other than [0xFF FF FF FF FF FF], security will be enabled.

For security reasons, reading this value will not return the actual password. The response code will depend on the present security setting.

Read Response Code	Meaning
0x00 00 00 00 00 00	Security is turned off.
0x00 00 00 00 00 01	Security is turned on.
0x00 00 00 00 00 02	This command is locked due to incorrect password entry.

A PMBus host does not need to validate the entire 6-byte response code. Only the last byte is significant to determining the security status: 0x00 (security off), 0x01 (security on), 0x02 (Invalid password).

12.36 (F2h) DISABLE_SECURITY (MFR_SPECIFIC_34)

This Read/Write Block commands allows temporarily disabling password security. While security is turned on, certain commands (defined by the MFR_SECURITY_BIT_MASK (F4) command) are write-protected and may not be modified. Attempts to write to a protected command while security is turned on will result in NACK due to invalid data.

The data for this command is a 6-byte binary string which contains a password.

If this sent password matches the password previously stored in the device's nonvolatile memory using the MFR_SETUP_PASSWORD (F1) command, a write will be ACKed and security will be turned off until the next time the PMBus device is power cycled.

If the sent password does not match the stored password, the write command will be NACKed. Additional attempts to issue this command will be NACKed until the next power cycle. This prevents an attacker from merely sending the command repeatedly with all possible passwords.

For security reasons, reading this value will not return the actual password. The response code will depend on the present security setting.

Read Response Code	Meaning
0x00 00 00 00 00 00	Security is turned off.
0x00 00 00 00 00 01	Security is turned on.
0x00 00 00 00 00 02	This command is locked due to incorrect password entry.

A PMBus host does not need to validate the entire 6-byte response code. Only the last byte is significant to determining the security status: 0x00 (security off), 0x01 (security on), 0x02 (Invalid password).

12.37 (F4h) MFR_SECURITY_BIT_MASK (MFR_SPECIFIC_36)

This Read/Write Block command controls which PMBus commands are password-protected to prevent unauthorized modification.

The data for this command is a 32-byte binary string. Each bit in the string corresponds to one of the 256 possible PMBus command codes.

When a bit is set to 1, the corresponding PMBus command is write-protected if security is on.
When a bit is set to 0, the corresponding PMBus command is not write-protected even if security is on.
For PMBus commands that do not support any write, the corresponding mask bit is ignored by firmware.

To password-protect a command, the byte and bit to be set in the security bit mask is determined from these formulas:
Byte = floor (Command_Code / 8)
Bit = 7 – (Command_Code – Byte * 8)

Byte	0								1								2		
Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	…
Command Code	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	…

Byte		29		30								31							
Bit	…	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Command Code	…	EE	EF	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	FA	FB	FC	FD	FE	FF

If security is turned on, the DISABLE_SECURITY command (F2h) must be issued to allow modifying the security bit mask. If security is turned off, the security bit mask can be modified anytime.

12.38 (F3h) GPIO_SEQ_CONFIG (MFR_SPECIFIC_35)

This Read/Write Block command configures the functionality of several input/output pins. Available options include:

- Internal Sequencing – Configures interdependency between voltage rails controlled by the same controller.
- External Sequencing – Configures interdependency between voltage rails controlled by separate controllers
- Fault Dependencies – Configure dependant pages which will also shut down when a fault occurs.
- Fault and Warning Status – Output to pins when certain faults or warnings occur.
- Power Good Indication – Configure a pin to reflect the Power Good status of the device (voltage for all configured pages has reached their POWER_GOOD_ON threshold). The polarity of the output pin can be selected.
- Define Input Source to be used with SEQ_TIMEOUT.
- Primary Function – Pins may be used for their primary function, as labeled in the Data Sheet.
- Sequencing examples can be found in the application note: “UCD92xx Sequencing Configuration” (SLUA481).

The UCD92xx controllers allow great flexibility in assigning output pins to a variety of signals. Not all configurations will need to use every possible pin for its primary purpose. For example, a 64-pin UCD9240 configured to have five output phases would have one free DPWM output and a corresponding SRE output free for use in dependency communication. Configurations that do not use fan control could use the FAN_PWM and FAN_TACH pins. Power drivers that do not report faults to the controller would free up the FLT pins.

The first generation of UCD92xx models (UCD9220 and UCD9240) supported GPIO_SEQ_CONFIG with a command format that used a 29-byte data field. Later models changed to a second-generation format that uses a 32-byte data field. The data fields between the two generations have significant differences between them.

12.38.1 First-Generation GPIO_SEQ_CONFIG (UCD9220 and '9240)

A total of 12 pin assignments may be configured as control inputs or status outputs. Of these, up to 8 pins (A through H) may be used as status outputs, and up to 4 pins (W through Z) may be configured as control inputs. This is a limitation of the firmware. In most configurations far fewer than 12 pins will be unassigned from their primary purpose leaving them available for GPIO or sequencing, so the firmware limits will not be a factor.

The PowerGood pin is configured as an output active high by default. It uses up one of the 12 GPIO pin assignments.

NOTE: Only one PowerGood event pin can be configured. This is a limitation of the firmware. If more than one PowerGood event pin is configured only the last one in the command will be configured properly.

NOTE: Only one Over-current Warning event pin can be configured. This is a limitation of the firmware. If more than one Over-current Warning event pin is configured only the last one in the command will be configured properly.

Table 8: GPIO_SEQ_CONFIG Command (First-Generation Data Format)

Byte Number (Write)	Byte Number (Read)	Description
0		CMD = F3
1	0	BYTE_COUNT = 24
		Control Input Dependency Masks
2	1	PAGE0 Input Dependency Mask
3	2	PAGE1 Input Dependency Mask
4	3	PAGE2 Input Dependency Mask
5	4	PAGE3 Input Dependency Mask
		Status Output Dependency Masks
6	5	Output A Dependency Configuration
7	6	Output B Dependency Configuration
8	7	Output C Dependency Configuration
9	8	Output D Dependency Configuration
10	9	Output E Dependency Configuration
11	10	Output F Dependency Configuration
12	11	Output G Dependency Configuration
13	12	Output H Dependency Configuration
		Control Input Pin Configurations
14	13	Input Pin W Configuration
15	14	Input Pin X Configuration
16	15	Input Pin Y Configuration
17	16	Input Pin Z Configuration
		Status Output Pin Configurations
18	17	Output A Pin Configuration
19	18	Output B Pin Configuration
20	19	Output C Pin Configuration
21	20	Output D Pin Configuration
22	21	Output E Pin Configuration
23	22	Output F Pin Configuration
24	23	Output G Pin Configuration
25	24	Output H Pin Configuration
		Fault Dependants Configurations
26	25	PAGE0 Fault Dependant Mask
27	26	PAGE1 Fault Dependant Mask
28	27	PAGE2 Fault Dependant Mask
29	28	PAGE3 Fault Dependant Mask
		Seq Timeout Configuration
30	29	Input source for SEQ_TIMEOUT

GPIO_SEQ_CONFIG Control Input Dependency Masks (First-Generation Data Format)
Each of the four pages has its own Input Dependency Mask, whose bits are defined as follows:

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Purpose	PAGE3 POWER GOOD	PAGE2 POWER GOOD	PAGE1 POWER GOOD	PAGE0 POWER GOOD	SEQ_IN PIN 'Z'	SEQ_IN PIN 'Y'	SEQ_IN PIN 'X'	SEQ_IN PIN 'W'

Each page can be dependent on the state of several other pages and/or input pins. The same pages and pins may also be used to control multiple pages.

The POWER GOOD bits will be active when the selected page has reached the threshold defined by POWER_GOOD_ON. When the selected controlling page drops below the POWER_GOOD_OFF threshold (either due to a fault that triggers shutdown or due to a host commands), then this page will turn off.

The SEQ_IN bits will be active when the selected sequencing input pin is active. When the input pin goes inactive, this page will turn off.

GPIO_SEQ_CONFIG Status Output Dependency Configuration (First-Generation Data Format)
Each status output signal (A through H) has one configuration byte whose bits are defined as follows:

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Purpose	Event Type from Table 9				PAGE3	PAGE2	PAGE1	PAGE0

Bits 7:4 contain the Event Type, which selects the type of controller event that will change the output status signal. The possible events are shown in Table 9 below.
Bits 3:0 are mask bits that select which page(s) are monitored to create the output status signal. When multiple pages are selected, they may be combined with a logical AND (turning on only when ALL selected pages are active) or a logical OR (turning on when ANY selected page is active).

Table 9: GPIO_SEQ_CONFIG Output Dependency Event Type (First-Generation Data Format)

Event Number	Abbreviation	Description	Multi-Page Logical Combination
1	SEQ_REG	Sequencing. Selected rail(s) have reached their POWER_GOOD_ON threshold.	AND
2	PWR_GOOD	Power Good. Note that this event is now redundant to configuring SEQ_REG for all pages. It is retained for backward compatibility.	AND
3	OD_VALID	Open-Drain Outputs Valid	None
4	OCW	Over-Current Warning	OR

Input and Output Pin Assignment (First-Generation Data Format)
Each of the input pins (W through Z) and output pins (A through H) are configured using one byte whose bits are defined as follows:

Bit	7	6	5	4	3	2	1	0
Purpose	**Pin ID** from Table 10					**Polarity** 0= Active Low 1= Active High	**Mode** 0= Unused 1= Input 2= Actively Driven Output 3= Open-Drain Output	

Bits 1:0 set the mode for the pin.
Bit 2 sets the output polarity.
Bits 7:3 select the Pin ID of the desired I/O pin. The pin IDs are numbers from 0 through 27. Note that the Pin ID numbers are the same for all models of the UCD92xx. This is different from the hardware pin numbers which differ for each package type. Table 10 shows the relationship between Pin ID and hardware Pin Number for each package type.

Table 10: GPIO_SEQ_CONFIG Pin ID Definitions (First-Generation Data Format)

Pin ID	Pin Name	Direction	UCD9240 80 Pin Number	UCD9240 64 Pin Number	UCD9220 40 Pin Number
0	FLT-1A	I/O	15	11	6
1	FLT-1B	I/O	16	12	--
2	FLT-2A	I/O	17	13	7
3	FLT-2B	I/O	18	14	--
4	FLT-3A	I/O	29	25	--
5	FLT-3B	I/O	41	--	--
6	FLT-4A	I/O	42	34	--
7	FLT-4B	I/O	43	--	--
8	DPWM-1A	I/O	21	17	10
9	DPWM-1B	I/O	22	18	--
10	DPWM-2A	I/O	23	19	12
11	DPWM-2B	I/O	24	20	--
12	DPWM-3A	I/O	25	21	14
13	DPWM-3B	I/O	26	--	--
14	DPWM-4A	I/O	27	23	16
15	DPWM-4B	I/O	28	--	--
16	SRE-1A	I/O	12	22 (1)	11
17	SRE-1B	I/O	11	24 (2)	--
18	SRE-2A	I/O	51	33	13 (3)
19	SRE-2B	I/O	24	35	--
20	SRE-3A	I/O	25	29	15
21	SRE-3B	I/O	26	--	--
22	SRE-4A	I/O	27	30	17 (4)
23	SRE-4B	I/O	28	--	--
24	POWER_GOOD	I/O	49	39	22
25	FAN_TACH	I/O	32	36	--
26	FAN_PWM	In Only	53	41	--
27	DIAG_LED	In Only	10	--	--

(1) UCD9240-46 Errata – use Pin ID 13 for SRE-1A
(2) UCD9240-46 Errata – use Pin ID 15 for SRE-1B
(3) UCD9220-48 Errata – use Pin ID 11 for SRE-2A

(4) UCD9220-48 Errata – use Pin ID 15 for SRE-4A

The FAN_PWM and Diag_LED pins are outputs when configured for their primary purpose. When configured for sequencing, they may be used only as inputs.

GPIO_SEQ_CONFIG Fault Dependant Masks (First-Generation Data Format)

Each of the four pages has its own Fault Dependant Mask, whose bits are defined as follows:

Bit	**3**	**2**	**1**	**0**
Purpose	Page3 dependant	Page2 dependant	Page1 dependant	Page0 dependant

Each page can have multiple dependant pages. When a fault occurs on any page, if its response is to shut down, all dependant pages will also be shut down.

Sequencing Timeout Configuration

The SEQ_TIMEOUT command (D0h) defines a window during which an external event is expected to occur. That event is defined by the pin selected by this byte. The pin's polarity and ID are configured using the Input Pin Assignment described above. 0 selects pin W, 1 selects pin X, 2 selects pin Y and 3 selects pin Z. Other values are invalid. The pin is ignored if the SEQ_TIMEOUT window is set to 0.

Validity Checking:

Pin Configuration Rules:

- Zero to four input pins may be assigned.
- The input pin configurations, if used, must start in the Input Pin W Configuration byte and continue in consecutive order with no gaps.
- The mode bits for the four Input Pin W-Z Configuration bytes must contain either 0 (unused) or 1 (input).

- Zero to eight output pins may be assigned.
- The output pin configurations, if used, must start in the Output Pin A Configuration byte and continue in consecutive order with no gaps.
- The mode bits for the eight Output Pin A-H Configuration bytes must contain either 2 (actively driven output) or 3 (open-drain output).
- All outputs used for sequencing (event type 1 from Table 9) must be assigned together before assigning any other output event type from Table 9).

Pin Usage Conflicts:

It is possible to issue this command with selections that may conflict with other settings. In most cases, this command will take priority over the other setting. The notable exceptions are the DPWM and SRE pins.

DPWM and SRE (8 pins each): The PHASE_INFO command is used to control the assignment of DPWM output to voltage rails. DPWM and SRE outputs that are assigned for normal use by the PHASE_INFO command will not be allowed to be used for GPIO or sequencing. Attempts to do so will generate a MFR_CONFIG_INVALID error (bit 5 of STATUS_MFR).

FLT (8 pins): Some power drivers provide fault reporting back to the UCD92xx controller; others do not. For this reason, the PHASE_INFO command cannot be relied on to determine whether the FLT input pins are to be used for fault detection. If an FLT pin is selected for use by this command, it will no longer be used for fault protection.

If this command is used to configure a pin for a specific GPIO or sequencing purpose and then issued again with the same pin unassigned, the pin may not revert back to its default usage until after the controller has been reset or power cycled.

Because the interactions between settings are so complex and depend on the order in which the PMBus commands are issued, the UCD92xx firmware will not attempt to detect and prevent all possible invalid setting combinations. The Fusion Digital Power Designer GUI provides some additional validity checking, but it is ultimately up to the user to ensure that conflicting GPIO configurations are not selected.

12.38.2 Second-Generation GPIO_SEQ_CONFIG (UCD9224, '9246, and '9248)

This Read/Write Block command configures the functionality of several input/output pins and the interaction (sequencing) among those pins and the internal rails. Features include:

- Internal Sequencing – Configures interdependency between voltage rails controlled by the same controller.
- External Sequencing – Configures interdependency between voltage rails controlled by separate controllers.
- Fault Slaves – Configure slave pages which will also shut down when a fault occurs (immediately or after retries are exhausted).
- OC Warning Status – Output to a pin when an over current warning occurs.
- Power Good Indication – Configure a pin to reflect the Power Good status of a combination of internal rails. The polarity of the output pin can be selected.
- Define an Input Source to be used with the SEQ_TIMEOUT command which is used to test that an external event occurs within a specified time
- Sequencing examples can be found in the application note: "Configuring the Sequencing Capabilities of the UCD92xx" (SLUA481).

This command description applies to the UCD9224, UCD9246 and UCD9248 only. The UCD9220 and UCD9240 (both 64 and 80 pin packages) had errata which limited their sequencing capabilities. If sequencing is required for a system, the design should use the updated devices. The UCD921x and UCD9244 do not support The GPIO_SEQ_CONFIG command.

The UCD92xx controllers allow great flexibility in assigning pins to a variety of input and output signals. Not all configurations will need to use every possible pin for its primary purpose. For example, a UCD9246 configured to have five output phases would leave one DPWM pin, one SRE pin and one FLT pin available for alternate uses. Power drivers that do not report faults to the controller free up the FLT pins.

A total of 10 pins may be configured as control inputs or status outputs. Of these, up to 6 pins (A through F) may be used as status outputs, up to 4 pins (W through Z) can be configured as control inputs. These are limitations of the firmware, however in most configurations far fewer than 10 pins will be unassigned from their primary purpose (and thus available for GPIO or sequencing) so the firmware limits will not typically be a factor.

The GPIO_SEQ_CONFIG command comprises 31 bytes as defined in the following table. Each group of pins will be described further in the proceeding sections.

Byte Number (Write)	Byte Number (Read)	Description
0		CMD = F3
1	0	BYTE_COUNT = 30
		Turn On Dependency Masks
2	1	PAGE0 Turn On Dependency Mask
3	2	PAGE1 Turn On Dependency Mask
4	3	PAGE2 Turn On Dependency Mask
5	4	PAGE3 Turn On Dependency Mask
		Stay On Dependency Masks
6	5	PAGE0 Stay On Dependency Mask
7	6	PAGE1 Stay On Dependency Mask
8	7	PAGE2 Stay On Dependency Mask
9	8	PAGE3 Stay On Dependency Mask
		Output Dependency Masks
10	9	Output A Dependency Mask
11	10	Output B Dependency Mask
12	11	Output C Dependency Mask
13	12	Output D Dependency Mask
14	13	Output E Dependency Mask
15	14	Output F Dependency Mask
		Input Pin Configurations
16	15	Input Pin W Configuration
17	16	Input Pin X Configuration
18	17	Input Pin Y Configuration
19	18	Input Pin Z Configuration
		Output Pin Configurations
20	19	Output A Pin Configuration
21	20	Output B Pin Configuration
22	21	Output C Pin Configuration
23	22	Output D Pin Configuration
24	23	Output E Pin Configuration
25	24	Output F Pin Configuration
		Fault-Slaves Masks
26	25	PAGE0 Fault Slaves Mask
27	26	PAGE1 Fault Slaves Mask
28	27	PAGE2 Fault Slaves Mask
29	28	PAGE3 Fault Slaves Mask
		Miscellaneous Configurations
30	29	Input source for SEQ_TIMEOUT
31	30	Shutdown mode configuration

Turn On Dependency Masks

Each of the four pages has its own Turn On Dependency Mask, whose bits are defined as follows:

Bit	7	6	5	4	3	2	1	0
Purpose	PAGE3 POWER GOOD	PAGE2 POWER GOOD	PAGE1 POWER GOOD	PAGE0 POWER GOOD	SEQ_IN PIN 'Z'	SEQ_IN PIN 'Y'	SEQ_IN PIN 'X'	SEQ_IN PIN 'W'

The turn on condition for each page can be dependent on the state of several other pages and/or input pins. The same pages and pins may also be used to control multiple pages. The turn on dependency defines a set of conditions which will allow a page to turn on when met. Note that the logical AND of all conditions must be met. Once the page is on, the turn on dependencies have no further effect on the operating status of the page. Specifically, they will not cause a page to turn off.

Turn-On Dependencies work in parallel with the PMBus defined mechanisms used to enable an output. That is, both the Turn-On Dependencies and the PMBus mechanism must be satisfied. For example if the page responds to an OPERATION command, until ON is specified by the command, the page will remain off even if all of the sequencing Turn-On Dependencies are met. Further, the order in which they are met is irrelevant; the OPERATION command could be issued first and then the sequencing requirements, when met, would turn on the output, or the sequencing requirements could be met first in which case the page would wait for the command. In the ON_OFF_CONFIG command an ALWAYS_CONVERTING setting, is still subject to the specified Turn-On Dependencies.

After a fault, the PMBus specification declares that an OFF/ON sequence occur before the rail is allowed to restart. Note that the toggle of a Turn-On sequencing pin is interpreted to meet this requirement. For example consider a page that responds to the CTRL_PIN and has a Turn-On Dependency which shuts down due to a fault. A toggle low then high on either the CTRL_PIN or the sequencing pin is sufficient to reset the page.

The POWER GOOD bits in the mask will be active when the selected page has reached the threshold defined by POWER_GOOD_ON; the SEQ_IN bits will be active when the selected sequencing input pin is active – defined by its specified polarity.

Stay On Dependency Masks

Each of the four pages has its own Input Stay On Dependency Mask, whose bits are defined as follows:

Bit	7	6	5	4	3	2	1	0
Purpose	PAGE3 POWER GOOD	PAGE2 POWER GOOD	PAGE1 POWER GOOD	PAGE0 POWER GOOD	SEQ_IN PIN 'Z'	SEQ_IN PIN 'Y'	SEQ_IN PIN 'X'	SEQ_IN PIN 'W'

The stay on condition for each page can be dependent on the state of several other pages and/or input pins. The same pages and pins may also be used to control (either Turn-On or Turn-Off) multiple pages.

It is important to note that the absence of a stay-on dependency will not prevent a page from turning on. A transition from active to inactive on one of the stay-on input pins is required to initiate a shut-down. The device will start checking the stay on dependents the moment a rail has been commanded to turn on. If a transition from active to inactive occurs on a stay on dependent during the TON_DELAY time or during the TON_RISE ramp, the rail will be shutdown.

When any selected controlling page drops below the POWER_GOOD_OFF threshold (either due to a fault that triggers shutdown or due to a host command), then this page will turn off.

When any of the selected input pins goes inactive, this page will turn off. The mode (soft or immediate) used for turning off the rail is determined by bits in the Shutdown Mode Settings byte of this command.

Note that the turn-on and stay-on dependencies share the same input pins and have identical bit definitions. One difference in behavior is that all of the Turn-On Dependencies must be met to turn a page on, and loosing any of the Stay-On Dependencies will shut a page down.

Output Dependency Masks
Each status output signal (A through F) has one configuration byte whose bits are defined as follows:

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Purpose	Event Type from Table 9				PAGE3	PAGE2	PAGE1	PAGE0

The Output Dependency Masks are used to select the corresponding pin the Output Pin configuration table for use as a sequencing output and define its function.

Bits 7:4 contain the Event Type, which selects the type of controller event that will change the output status signal. The possible events are shown in Table 9 below.
Bits 3:0 are mask bits that select which page(s) are monitored to create the output status signal. When multiple pages are selected, they may be combined with a logical AND (active only when ALL selected pages are active) or a logical OR (active when ANY selected page is active) depending on the Event Type.

Table 11: Output Dependency Event Types

Event Number	Abbreviation	Description	Multi-Page Logical Combination
0	UNUSED	No sequencing use defined.	N/A
1	SEQ_REG	Sequencing. Selected rail(s) have reached their POWER_GOOD_ON threshold. Any page falling below its POWER_GOOD_OFF threshold will de-assert the pin.	AND
2	PWR_GOOD	Power Good. Note that this event is similar to the response that would be obtained by configuring an output dependent on the SEQ_REG state of all rails, however this setting will respond more quickly. By default the PGOOD pin will be configured for PWR_GOOD as an open drain, active high output. Only one PWR_GOOD output can be defined.	AND
3	OD_VALID	Open-Drain Outputs Valid. When other pins are configured as open-drain outputs, this signal can be monitored by a host to verify that their outputs are meaningful. After a device reset, it is set after all pins have been configured and will remain set thereafter. This output type might also be useful as an indication that the 92xx has completed its power-up configuration.	None
4	OCW	Over-Current Warning. Can be used to alert a host to an impending problem. The SMBus alert line is asserted on a fault – this output may be used to provide some prior warning that a fault is imminent. Only one OCW output can be defined.	OR

Input and Output Pin Configurations
Each of the input pins (W through Z) and output pins (A through F) are configured using one byte whose bits are defined as follows:

Bit	7	6	5	4	3	2	1	0
Purpose	**Pin ID** from Table 10					**Polarity** 0= Active Low 1= Active High	**Mode** 0= Unused 1= Input 2= Actively Driven Output 3= Open-Drain Output	

Bits 1:0 set the mode for the pin.
Bit 2 sets the output polarity.
Bits 7:3 select the Pin ID of the desired I/O pin. The pin IDs are numbers from 0 through 27. Note that the Pin ID numbers are the same for all models of the UCD92xx. This is different from the hardware pin numbers which differ for each package type. Table 10 shows the relationship between Pin ID and hardware Pin Number for each package type, it also shows how the pin may be used, and its state after a reset.

Table 12: Pin ID Definitions

Pin ID	Pin Name	UCD9248			UCD9246			UCD9224			UCD9224
		Pin#	I/O	Reset	Pin#	I/O	Reset	Pin#	I/O	Reset	Alias (*)
0	FLT-1A	15	I/O	hiZ	11	I/O	hiZ	6	I/O	hiZ	--
1	FLT-1B	16	I/O	hiZ	12	I/O	hiZ	7	I/O	hiZ	--
2	FLT-2A	17	I/O	hiZ	13	I/O	hiZ	8	I/O	hiZ	--
3	FLT-2B	18	I/O	hiZ	14	I/O	hiZ	--	--	--	--
4	FLT-3A	29	I/O	hiZ	25	I/O	hiZ	--	--	--	--
5	FLT-3B	41	I/O	hiZ	--	--	--	--	--	--	--
6	FLT-4A	42	I/O	hiZ	34	I/O	hiZ	25	I/O	hiZ	FLT-3A
7	FLT-4B	43	I/O	hiZ	--	--	--	--	--	--	--
8	DPWM-1A	21	I/O	low	17	I/O	low	12	I/O	low	--
9	DPWM-1B	22	I/O	low	18	I/O	low	13	I/O	low	--
10	DPWM-2A	23	I/O	low	19	I/O	low	14	I/O	low	--
11	DPWM-2B	24	I/O	low	20	I/O	low	--	--	--	--
12	DPWM-3A	25	I/O	low	21	I/O	low	--	--	--	--
13	DPWM-3B	26	I/O	low	--	--	--	--	--	--	--
14	DPWM-4A	27	I/O	low	23	I/O	low	16	I/O	low	DPWM-3A
15	DPWM-4B	28	I/O	low	--	--	--	--	--	--	--
16	SRE-1A	12	I/O	hiZ	22	I/O	low	9	I/O	hiZ	--
17	SRE-1B	11	I/O	hiZ	24	I/O	low	18	I/O	hiZ	--
18	SRE-2A	51	I/O	hiZ	33	I/O	hiZ	15	I/O	low	--
19	SRE-2B	37	I/O	hiZ	35	I/O	hiZ	--	--	--	--
20	SRE-3A	38	I/O	hiZ	29	I/O	hiZ	--	--	--	--
21	SRE-3B	52	I/O	hiZ	--	--	--	--	--	--	--
22	SRE-4A	33	I/O	hiZ	30	I/O	hiZ	17	I/O	low	SRE-3A
23	SRE-4B	50	I/O	hiZ	--	--	--	--	--	--	--
24	PGOOD	49	I/O	hiZ	39	I/O	hiZ	26	I/O	hiZ	--
25	SEQ-1	32	I/O	hiZ	36	I/O	hiZ	21	I/O	hiZ	--
26	SEQ-2	53	I	hiZ	41	I	hiZ	22	I/O	hiZ	--
27	SEQ-3	10	I	hiZ	--	--	--	30	I/O	hiZ	--

(*) Package name and pin name are re-numbered for data sheet clarity. Advertised phases are 1A, 1B, 2A and 3A. The firmware (and therefore the command that accesses it) uses the underlying pad, 4A.

Note that in some packages, some of the SEQ-*n* pins use underlying pads which only allow their configuration to be as an input.

For output pins the reset state should be considered if the state of the pin will affect the system at initial power-up. The state shown in the Reset column of the preceding table indicates the behavior of the pin until the firmware has completed its initialization (approximately 15 msec). For example an open drain output assigned to a pin which at reset is driven low, may result in a false indication of that state.

Fault Slaves Masks
Each of the four pages has its own Fault Slaves Mask, whose bits are defined as follows:

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Purpose	Page3 allow retries	Page2 allow retries	Page1 allow retries	Page0 allow retries	Page3 fault slaves	Page2 fault slaves	Page1 fault slaves	Page0 fault slaves

Each page can have multiple slave pages. When a fault occurs on any page, if its response is to shut down; all assigned slave pages will also be shut down. If the allow_retries flag is set, the slave page(s) will remain running until all retries are exhausted by the master. Whether or not retries are allowed, when a fault slave is finally shut down, it will be done with an immediate off response.

Unlike the master rail, after being shut down, slave rails are not latched off. Only the master rail which actually experienced the fault is latched and will require the off-then-on sequence to turn back on. For the fault-slaves, a simple on command is sufficient to restart. When a page is shut down as the result of a fault-slave action, a status bit is set in its MFR_STATUS word indicating the reason it was turned off.

Input source for SEQ_TIMEOUT
The SEQ_TIMEOUT command (D0h) defines a window during which an external event is expected to occur. That event is associated with the pin selected by this byte. The pin's polarity and ID are configured using the Input Pin Configuration described above. 0 selects pin W, 1 selects pin X, 2 selects pin Y, and 3 selects pin Z. Other values are invalid. The pin is ignored by pages which have the SEQ_TIMEOUT window set to 0.

Shutdown Mode Configuration
Rails can be turned off in one of two modes: Soft-off will use the TOFF_DELAY and TOFF_FALL constants while Immediate-off will disable PWM and SRE outputs as soon as commanded. The bits in the shutdown mode determine the behavior on a SEQ_REG initiated shutdown. A value of 0 selects the soft-off mode and 1 selects a shutdown immediate mode.

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Purpose					PAGE3 stay-on-dependent shutdown mode.	PAGE2 stay-on-dependent shutdown mode.	PAGE1 stay-on-dependent shutdown mode.	PAGE0 stay-on-dependent shutdown mode.

Validity Checking:
The host must obey the following rules when issuing the GPIO_SEQ_CONFIG command.

Pin Configuration Rules:

- Zero to four input pins may be assigned.
- The Input Pin Configurations, if used, must start in the Input Pin W Configuration byte and continue in consecutive order with no gaps.
- The Mode bits (1:0) for the four Input Pin Configuration bytes (W-Z) must contain either 0 (unused) or 1 (input).
- Zero to six output pins may be assigned.
- The Output Pin Configurations, if used, must start in the Output Pin A Configuration byte and continue in consecutive order with no gaps.
- The Mode bits for the six Output Pin Configuration (A-F) bytes must contain either 2 (actively driven output) or 3 (open-drain output).
- All outputs used for SEQ_REG sequencing (event type 1 from Table 9) must be assigned first before assigning any other output event types from Table 9).

Pin Usage Conflicts:
It is possible to issue this command with selections that may conflict with other settings. In most cases, this command will take priority over the other setting. The notable exceptions are the DPWM and SRE pins.

DPWM and SRE (8 pins each): The PHASE_INFO command is used to control the assignment of DPWM output to voltage rails. DPWM and SRE outputs that are assigned for normal use by the PHASE_INFO command will not be allowed to be used for GPIO or sequencing. Attempts to do so will generate a MFR_CONFIG_INVALID error (bit 5 of STATUS_MFR).

FLT (8 pins): Some power drivers provide fault reporting back to the UCD92xx controller; others do not. For this reason, the PHASE_INFO command cannot be relied on to determine whether the FLT input pins are to be used for fault detection. If an FLT pin is selected for use by this command, it will no longer be used for fault protection. Note that this configuration may be made intentionally in order to disable the fault response without actually using the configured pin.

If this command is used to configure a pin for a specific GPIO or sequencing purpose and then issued again with the same pin unassigned, the pin may not revert back to its default usage until after the controller has been reset or power cycled.

Because the interactions between settings are so complex and depend on the order in which the PMBus commands are issued, the UCD92xx firmware will not attempt to detect and prevent all possible invalid setting combinations. The Fusion Digital Power Designer GUI provides some additional validity checking, but it is ultimately up to the user to ensure that conflicting GPIO configurations are not selected.

After receiving this command the firmware will latch its outputs off; this forces the user to store the configuration to flash and reboot in order for them to take affect.

12.39 (F5h) TEMP_BALANCE_IMIN (MFR_SPECIFIC_37)

This Read/Write Word command sets the minimum current required before applying temperature balancing compensation.

At low currents, there may be insufficient thermal response to effect a temperature change; setting a minimum current threshold prevents the controller from creating a current imbalance which has no impact on the temperature, but which may result in large currents in some phases when a load begins drawing more current.

Temperature balancing can be disabled by specifying a threshold larger than any current the system should ever experience.

This command has two data bytes formatted in the Linear11 Data format. The units are amperes. The default value is 511.5 Amps, which is the largest Linear11 value that may be stored. (*Note: In firmware 3.06, the default was set to 468.75 A. Both values are so large that they will behave the same.*)

12.40 (F6h) LIGHT_LOAD_LIMIT_LOW (MFR_SPECIFIC_38)

This Read/Write Word command specifies the load current below which the controller for the selected page may switch to its Light Load settings. See the LIGHT_LOAD_LIMIT_HIGH command in section 12.29 for details.

This command has two data bytes formatted in the Linear11 Data format. The units are amperes.

12.41 (F7h) FAST_OC_FAULT_LIMIT (MFR_SPECIFIC_39)

This Read/Write Word command specifies the threshold used by the fast analog comparators used to quickly detect over-current conditions on some output power phases.

This command has two data bytes formatted in the Linear11 Data format. The units are amperes, and range from 0.0 to 511.5 A. The default value is 0.0 A.

A setting of 0.0 A is treated as a special case that disables the analog comparators. This may be used to prevent false triggers from inadvertently shutting off the outputs.

FAST_OC_FAULT_LIMIT vs. IOUT_OC_FAULT_LIMIT

The UCD9240 uses a combination of 'A' and 'B' phases for its output. The analog comparators only monitor the 'A' phases. The IOUT_OC_FAULT_LIMIT command (46h) sets a similar threshold that is used for all phases (A or B). The IOUT_OC_FAULT_LIMIT includes additional digital smoothing to remove noise spikes, but it responds more slowly than the analog comparators. The table below shows the difference between the two

	FAST_OC_FAULT_LIMIT(F7h)	**IOUT_OC_FAULT_LIMIT (46h)**
Type	Hardware Analog Comparator	Firmware
Output Phases Monitored	'A' phases only	All phases (A and B)
Speed	Fast: 3 switching cycles + ~14-30 us	Medium: ~150-200 us.
Input Smoothing	No	Yes
Fault Responses	Immediate shutdown with possible retries. Set by IOUT_OC_FAULT_RESPONSE	Immediate or delayed shutdown, with possible retries. Set by IOUT_OC_FAULT_RESPONSE (49h) command.

Range of Voltage from Current Sense Circuit	0.03 to 2.00 V	0 to 2.50V

False Trigger Prevention
The analog comparators that quickly detect over-current conditions have two key controls:

1) Threshold: This level is compared against the voltage from the current sense circuit, Visense.
2) Duration: A hardware counter is incremented for each consecutive switching cycle where Visense exceeds the threshold. When the counter reaches the Duration limit, an over-current interrupt is generated.

The Threshold is set using the FAST_OC_FAULT_LIMIT command. This threshold is used during all modes except IDLE.

While operating in REGULATE mode, the comparator responds quickly (3 switching cycles). To prevent short current spikes during turn-on and turn-off from triggering false OC faults, a longer Duration is used (50 switching cycles).

MODE	Threshold	Duration (Switching Cycles)	Comment
IDLE	2.0V (Maximum Possible)	255	Basically ignore over-currents.
REGULATE	FAST_OC_FAULT_LIMIT	3	Respond quickly to over-currents.
Others: RAMP_UP, RAMP_DOWN, TRACKING_RAMP, START_DELAY, STOP_DELAY	FAST_OC_FAULT_LIMIT	50	Ignore short spikes when power stages first turn on, but respond to sustained over-current conditions.

12.42 (F8h) POWER_GOOD_CONFIG (MFR_SPECIFIC_40) (UCD9222A, '9244A, '9244N)

This Read/Write Byte command configures the PowerGood output pin. This command is only supported on the UCD9244A and UCD9222A, which have most of their I/O pins used as VID inputs. The PowerGood pin on non-VID devices can be configured using the more general GPIO_SEQ_CONFIG command.

The PowerGood pin output is the logical-AND of the power-good signals from every selected rail. If any selected rail is not good, the PowerGood output will turn off. By default, all rails affect the PowerGood status. This command allows the status from some rails to be ignored.

The data for this command is shown in this table.

Byte Number (Write)	Byte Number (Read)	Description
0		CMD = 0xF8
1	0	Data Bit 7-4: Reserved Bit 3: PG mask for Page 3 (Rail 4) Bit 2: PG mask for Page 2 (Rail 3) Bit 1: PG mask for Page 1 (Rail 2) Bit 0: PG mask for Page 0 (Rail 1)

For each PG mask bit
1 = Include rail's power-good status into PowerGood output
0 = Ignore rail's power-good status.

The default value is 0x0F (all rails enabled), regardless of the number of rails configured for use by the PHASE_INFO setting.

12.43 (F9h) Undefined (MFR_SPECIFIC_41)

12.44 (FAh) PHASE_DROP_CAL (MFR_SPECIFIC_42)

This Read/Write Word command specifies a delay for the specified page to be used between turning off the DPWM output and the SRE when dropping phases. The delay in switching cycles is equal to the phase current in Amps * PHASE_DROP_CAL. Ex. Consider dropping one phase of a four phase output which is drawing 60 Amps (equally divided among the four phases): If PHASE_DROP_CAL is set to 1.2, then the 15 Amps on the phase times 1.2 will result in a delay of 18 switching cycles.

Experimentally this has been shown to significantly reduce the transient when dropping phases. The optimal value is highly dependent on the power stage parameters.

This command has two data bytes formatted in the Linear11 Data format. The units are switching cycles / Ampere. The default setting is 1.0.

12.45 (FBh) SYNC_OFFSET (MFR_SPECIFIC_43)

This Read/Write Word command defines the offset from the SYNC_IN pin to the rising edge of the first phase assigned to the selected page. (Subsequent phases of the page will be evenly distributed across the switching period.)

See the application note "UCD9224, UCD9246, UCD9248 Sync Application Report" for details on the implications of using this command; while precisely defined phase relationships between multiple pages can be set with this command, there is a cost in phase margin proportional to the delay specified.

This command has two data bytes formatted in the Linear11 Data format. The units are nsec.

12.46 (FCh) EADC_TRIM (MFR_SPECIFIC_44)

(Supported by UCD9240, '9220)
This Read/Write Byte command is an optional calibration parameter used to null any offsets that may be present between any two AFE_GAIN settings. The factory calibrates the EADC_TRIM to achieve absolute accuracy at a single AFE_GAIN setting. This calibration may introduce a small offset when switching between two different AFE_GAIN settings. For example, if the AFE_GAIN is set to 2x for RAMP and 4x for REGULATE there may be a slight offset in output voltage may be seen when switching from RAMP to REGULATE gains. This offset can be minimized by adjusting EADC_TRIM at the expense of absolute accuracy, which can be adjusted for by calibration VOUT_CAL_OFFSET after adjusting EADC_TRIM.

This command has one data byte. The parameter has not units and has a range of 0 to 63. Values greater than 63 will cause the firmware to use the default factory value for EADC_TRIM.

12.47 (FCh) VID_RESTART (MFR_SPECIFIC_44) (UCD9222A, '9244A, '9244N)

This paged Send Byte command restarts the VID startup sequence on the selected page(s) that are configured for VID operation. Setting PAGE=0xFF before issuing this command will restart the VID sequence on all VID-controlled rails. See section 11.4 for additional details of the VID_CONFIG command.

Actions that are taken:

- The output voltage set-point is recalculated using the VID Code Init setting from VID_CONFIG.
- If the 6-bit serial VID format was enabled by the VID_CONFIG setting, any partially-received message will be cleared. The VID_S edge-detection hardware will be configured to start looking for the first half-word of the VID message.
- If the VID Lockout Interval was enabled in the VID_CONFIG setting, the VID interface will be ignored for the selected time interval. If a lockout interval was in progress when the VID_RESTART command was issued, the interval will restart for its full duration with no regard for any partial interval that may have been completed.

12.48 (FDh) DEVICE_ID (MFR_SPECIFIC_45)

This Read-only Block Read command returns an ASCII string up to 32 characters in length. It is broken into three or four sections, separated by the vertical bar character ('|'). The format within each section may change in future releases so support tools should not rely on specific byte alignment; instead they should identify the sections and sub-sections using the vertical bar and the periods that separate them.

1. The first section is the hardware device ID (e.g. 'UCD9240-64").

2. The second section contains the firmware version information.
   Its format is "A.BB.C.DDDD", where
   A = Major Release Level (1 character)
   BB = Minor Release Level (1 or 2 characters)
   C = Sub-Release (1 character)
   DDDD = Build Number (4 characters)

   The major and minor release numbers will be incremented immediately after each official firmware release.

   The sub-release field allows for branching off the main development path to build updates based on older versions.

The build number is automatically updated every time firmware is compiled. The value does not reset to zero when the release level is updated. Several pre-release versions of firmware could have the same major, minor, and sub-release numbers. These different pre-release versions may be distinguished by the build number.

3. The third section contains the firmware compilation date.
   The date is reported in YYMMDD format, similar to the MFR_DATE command.

4. The optional fourth section may contain device-specific info.

Example: A typical DEVICE_ID string would be “UCD9240|2.4.0.0069|070509”. In this example,
Hardware device = UCD9240 (in Release 3.8 and later, the hardware device would have the number of package pins appended to it and would be, for example, “UCD9240-64”)
Firmware Major Release = 2
Firmware Minor Release = 4
Firmware Sub-Release = 0
Firmware Build Number = 69
Firmware Build Date = May 9, 2007

13 Range Checking and Limits

The following table shows the numerical limits for all the supported PMBus commands.

Table 13: Range and Limits for PMBus commands

Code	Command	Minimum	Maximum	Hardcoded Default	Comments
00h	PAGE	0	Num pages or 255	0	
01h	OPERATION	See comments	See comments	0x40	There are seven meaningful values for this parameter (0x00, 0x40, 0x80, 0x94, 0x98, 0x98, 0xA4 and 0xA8).
02h	ON_OFF_CONFIG	See comments	See comments	0x1A	>0x20 invalid, all others accepted
03h	CLEAR_FAULTS	n/a	n/a	n/a	WRITE ONLY
04h	PHASE	0	Num phases or 255	0	If PAGE = 255 then PHASE must be 255 for any PHASE related commands to be accepted.
05h-0Fh	Reserved				
10h	WRITE_PROTECT				NOT SUPPORTED
11h	STORE_DEFAULT_ALL	n/a	n/a	n/a	WRITE ONLY
12h	RESTORE_DEFAULT_ALL	n/a	n/a	n/a	WRITE ONLY
13h	STORE_DEFAULT_CODE				NOT SUPPORTED
14h	RESTORE_DEFAULT_CODE				NOT SUPPORTED
15h	STORE_USER_ALL				NOT SUPPORTED
16h	RESTORE_USER_ALL				NOT SUPPORTED
17h	STORE_USER_CODE				NOT SUPPORTED
18h	RESTORE_USER_CODE				NOT SUPPORTED
19h	CAPABILITY	n/a	n/a	0xB0	READ ONLY
1Ah	QUERY				NOT SUPPORTED
1Bh-1Fh	Reserved				
20h	VOUT_MODE	n/a	n/a	0x14	READ ONLY
21h	VOUT_COMMAND	0	15.9995	0	
22h	VOUT_TRIM				NOT SUPPORTED
23h	VOUT_CAL_OFFSET	-8	7.9995	0	Note this parameter is treated as a SIGNED variable
24h	VOUT_MAX	0	15.9995	1.6	1.6V / VOUT_SCALE_LOOP
25h	VOUT_MARGIN_HIGH	0	15.9995	0	
26h	VOUT_MARGIN_LOW	0	15.9995	0	
27h	VOUT_TRANSITION_RATE	0	1.92 / (AFEgain*VOUT_SCALE_LOOP)	0.24	
28h	VOUT_DROOP				NOT SUPPORTED
29h	VOUT_SCALE_LOOP	0	1.5996	1	
2Ah	VOUT_SCALE_MONITOR	0	1.5996	1	
2Bh-2Fh	Reserved				
30h	COEFFICIENTS				NOT SUPPORTED
31h	POUT_MAX				NOT SUPPORTED

Code	Command	Minimum	Maximum	Hardcoded Default	Comments
32h	MAX_DUTY	-100.125	100.125	100	Negative values are allowed but may cause unpredictable results. Due to LINEAR11 rounding and resolution the max number read back is 100.125. Internally the maximum number is 0.99.
33h	FREQUENCY_SWITCH	15.25	2000	500	All outputs must be OFF before this command can be updated
34h	Reserved				
35h	VIN_ON	0	31.9688	6	
36h	VIN_OFF	0	31.9688	5	
37h	INTERLEAVE				NOT SUPPORTED
38h	IOUT_CAL_GAIN	0.6113	20000	0	Number from 20000 to 40031 will result in 20000 because of internal resolution.
39h	IOUT_CAL_OFFSET	-511.5	511.5	0	
3Ah	FAN_CONFIG_1_2	n/a	n/a	0	valid values [0x00, 0x10, 0x20, 0x30, 0x80, 0x90, 0xA0, 0xB0]
3Bh	FAN_COMMAND_1	0	100	0	
3Ch	FAN_COMMAND_2				NOT SUPPORTED
3Dh	FAN_CONFIG_3_4				NOT SUPPORTED
3Eh	FAN_COMMAND_3				NOT SUPPORTED
3Fh	FAN_COMMAND_4				NOT SUPPORTED
40h	VOUT_OV_FAULT_LIMIT	0	15.9995	0	
41h	VOUT_OV_FAULT_RESPONSE	0x00	0xFF	0x80	
42h	VOUT_OV_WARN_LIMIT	0	15.9995	0	
43h	VOUT_UV_WARN_LIMIT	0	15.9995	0	
44h	VOUT_UV_FAULT_LIMIT	0	15.9995	0	
45h	VOUT_UV_FAULT_RESPONSE	0x00	0xFF	0x00	
46h	IOUT_OC_FAULT_LIMIT	-511.5	511.5	0	
47h	IOUT_OC_FAULT_RESPONSE	0x00	0xFF	0x80	
48h	IOUT_OC_LV_FAULT_LIMIT	0	15.9995	0	
49h	IOUT_OC_LV_FAULT_RESPONSE	0x00	0xFF	0x80	
4Ah	IOUT_OC_WARN_LIMIT	-511.5	511.5	0	
4Bh	IOUT_UC_FAULT_LIMIT	-511.5	511.5	0	
4Ch	IOUT_UC_FAULT_RESPONSE	0x00	0xFF	0x00	
4Dh	Reserved				
4Eh	Reserved				
4Fh	OT_FAULT_LIMIT	-255.75	255.75	80	
50h	OT_FAULT_RESPONSE	0x00	0xFF	0x80	
51h	OT_WARN_LIMIT	-255.75	255.75	75	
52h	UT_WARN_LIMIT				NOT SUPPORTED
53h	UT_FAULT_LIMIT				NOT SUPPORTED
54h	UT_FAULT_RESPONSE				NOT SUPPORTED
55h	VIN_OV_FAULT_LIMIT	0	15.984	15.5	
56h	VIN_OV_FAULT_RESPONSE	0x00	0xFF	0x80	
57h	VIN_OV_WARN_LIMIT	0	15.984	15	
58h	VIN_UV_WARN_LIMIT	0	15.984	6	
59h	VIN_UV_FAULT_LIMIT	0	15.984	5	
5Ah	VIN_UV_FAULT_RESPONSE	0x00	0xFF	0x00	
5Bh	IIN_OC_FAULT_LIMIT				NOT SUPPORTED
5Ch	IIN_OC_FAULT_RESPONSE				NOT SUPPORTED
5Dh	IIN_OC_WARN_LIMIT				NOT SUPPORTED
5Eh	POWER_GOOD_ON	0	15.9995	0	
5Fh	POWER_GOOD_OFF	0	15.9995	0	

Code	Command	Minimum	Maximum	Hardcoded Default	Comments
60h	TON_DELAY	0	3276	0	
61h	TON_RISE	0	3276	0	
62h	TON_MAX_FAULT_LIMIT	0	3276	0	
63h	TON_MAX_FAULT_RESPONSE	0x00	0xFF	0x00	
64h	TOFF_DELAY	0	3276	0	
65h	TOFF_FALL	0	3276	0	
66h	TOFF_MAX_WARN_LIMIT	0	3276 or 0x7FFF	0x7FFF	0x7FFF is a special value meaning there is no limit. See section 16.7 of the PMBus Specification.
67h	Reserved				
68h	POUT_OP_FAULT_LIMIT				NOT SUPPORTED
69h	POUT_OP_FAULT_RESPONSE				NOT SUPPORTED
6Ah	POUT_OP_WARN_LIMIT				NOT SUPPORTED
6Bh	PIN_OP_WARN_LIMIT				NOT SUPPORTED
6Ch-77h	Reserved				
78h	STATUS_BYTE				READ ONLY
79h	STATUS_WORD				READ ONLY
7Ah	STATUS_VOUT				READ ONLY
7Bh	STATUS_IOUT				READ ONLY
7Ch	STATUS_INPUT				READ ONLY
7Dh	STATUS_TEMPERATURE				READ ONLY
7Eh	STATUS_CML				READ ONLY
7Fh	STATUS_OTHER				READ ONLY
80h	STATUS_MFR_SPECIFIC				READ ONLY
81h	STATUS_FANS_1_2				READ ONLY
82h	STATUS_FANS_3_4				READ ONLY
83h-87h	Reserved				
88h	READ_VIN				READ ONLY
89h	READ_IIN				READ ONLY
8Ah	READ_VCAP				NOT SUPPORTED
8Bh	READ_VOUT				READ ONLY
8Ch	READ_IOUT				READ ONLY
8Dh	READ_TEMPERATURE_1				READ ONLY
8Eh	READ_TEMPERATURE_2				READ ONLY
8Fh	READ_TEMPERATURE_3				NOT SUPPORTED
90h	READ_FAN_SPEED_1				READ ONLY
91h	READ_FAN_SPEED_2				NOT SUPPORTED
92h	READ_FAN_SPEED_3				NOT SUPPORTED
93h	READ_FAN_SPEED_4				NOT SUPPORTED
94h	READ_DUTY_CYCLE				READ ONLY
95h	READ_FREQUENCY				NOT SUPPORTED
96h	READ_POUT				READ ONLY
97h	READ_PIN				READ ONLY
98h	PMBUS_REVISION				READ ONLY
99h	MFR_ID	n/a	n/a	MFR_ID	
9Ah	MFR_MODEL	n/a	n/a	MFR_MODEL	
9Bh	MFR_REVISION	n/a	n/a	MFR_REVISION	
9Ch	MFR_LOCATION	n/a	n/a	MFR_LOCATION	
9Dh	MFR_DATE	n/a	n/a	YYMMDD	
9Eh	MFR_SERIAL	n/a	n/a	000000	
9Fh	Reserved				
A0h	MFR_VIN_MIN				NOT SUPPORTED
A1h	MFR_VIN_MAX				NOT SUPPORTED
A2h	MFR_IIN_MAX				NOT SUPPORTED
A3h	MFR_PIN_MAX				NOT SUPPORTED
A4h	MFR_VOUT_MIN				NOT SUPPORTED
A5h	MFR_VOUT_MAX				NOT SUPPORTED
A6h	MFR_IOUT_MAX				NOT SUPPORTED

Code	Command	Minimum	Maximum	Hardcoded Default	Comments
A7h	MFR_POUT_MAX				NOT SUPPORTED
A8h	MFR_TAMBIENT_MAX				NOT SUPPORTED
A9h	MFR_TAMBIENT_MIN				NOT SUPPORTED
AAh-AFh	Reserved				
B0h-BFh	USER_DATA_00 - USER_DATA_15				
C0h-CFh	Reserved				
D0h	SEQ_TIMEOUT (MFR_SPECIFIC_00)	0	3276	0	
D1h	VOUT_CAL_MONITOR (MFR_SPECIFIC_01)	-8	7.9995	0	Note this parameter is treated as a SIGNED variable
D2h	PHASE_INFO (MFR_SPECIFIC_02)	n/a	n/a	0	See section 11 of this document.
D3h	VIN_SCALE_MONITOR (MFR_SPECIFIC_03)	0	1.998	1	
D4h	CLA_BANK (MFR_SPECIFIC_04)	0	3	0	0xFE and 0xFF are also valid.
D5h	CLA_GAINS (MFR_ SPECIFIC_05)	n/a	n/a	ZOH	Default value is Zero-Order Hold.
D6h	PAGE_ISOLATED (MFR_SPECIFIC_06)	0	1	1	
	DRIVER_CONFIG (MFR_SPECIFIC_06)	0	7	0	
D7h	EADC_SAMPLE_TRIGGER (MFR_SPECIFIC07)	224 to 239	(1e6/FREQUENCY_SWITCH) + min - 16	224	The minimum value depends on FREQUENCY_SWITCH. It will vary between 224 and 239 but be consistent for a given switching frequency.
D8h	ACTIVATE_CLA_BANK (MFR_ SPECIFIC_08)	n/a	n/a	n/a	
D9h	ROM_MODE (MFR_SPECIFIC_09)	n/a	n/a	n/a	WRITE ONLY
DAh	USER_RAM_00 (MFR_SPECIFIC_10)	0	255	0	
DBh	SOFT_RESET (MFR_SPECIFIC_11)	n/a	n/a	n/a	WRITE ONLY
DCh	IIN_SCALE_MONITOR (MFR_SPECIFIC_12)	0	1.998	0	
DDh	THERMAL_COEF (MFR_SPECIFIC_13)	-50	50	0.394	
DEh	PHASE_ENABLE (MFR_SPECIFIC_14)	0	Num phases	Enabled	
DFh	DRIVER_MIN_PULSE (MFR_SPECIFIC_15)	0	32736	0	
E0h	MIN_DUTY (MFR_SPECIFIC_16)	-100.125	100.125	-50	Due to LINEAR11 rounding and resolution the max number read back is 100.125. Internally the maximum number is 0.99.
E1h	SYNC_IN_OUT (MFR_SPECIFIC_17)	n/a	n/a	0x00FF	
E2h	PARM_INFO (MFR_SPECIFIC_18)	n/a	n/a	n/a	Index is checked to verify that it points to a valid base address
E3h	PARM_VALUE (MFR_SPECIFIC_19)	n/a	n/a	n/a	

Code	Command	Minimum	Maximum	Hardcoded Default	Comments
E4h	TEMPERATURE_CAL_GAIN (MFR_SPECIFIC_20)	-1638	1638	0	
E5h	TEMPERATURE_CAL_OFFSET (MFR_SPECIFIC_21)	-255.75	255.75	0	
E6h	TRACKING_SOURCE (MFR_SPECIFIC_22)	0x00	0xFF	0xFF	
E7h	TRACKING_SCALE_MONITOR (MFR_SPECIFIC_23)	0	7.9922	1	
E8h	FAN_SPEED_FAULT_LIMIT (MFR_SPECIFIC_24)	-32736	32736	0	
E9h	LOGGED_PEAKS (MFR_SPECIFIC_25)	n/a	n/a	n/a	Only valid write is all zeroes.
EAh	LOGGED_FAULTS (MFR_SPECIFIC_2	n/a	n/a	n/a	Only valid write is all zeroes.
EBh	LIGHT_LOAD_LIMIT_HIGH (MFR_SPECIFIC_27) (formerly called LIGHT_LOAD_LIMIT)	-511.5	511.5	0	
ECh	DEVICE_ID / ROM_VER (MFR_SPECIFIC_28)	n/a	n/a	n/a	READ ONLY
EDh	LIGHT_LOAD_CONFIG (MFR_SPECIFIC_29)	n/a	n/a	0	
EEh	PREBIAS_OFFSET	-100.125	100.125	0	
EFh	PREBIAS_GAIN	-3.9961	3.9961	1	
F0h	EXECUTE_FLASH (MFR_SPECIFIC_32)	n/a	n/a	n/a	WRITE_ONLY
F1h	MFR_SETUP_PASSWORD (MFR_SPECIFIC_33)	n/a	n/a	n/a	Default password is 0xFFFFFFFFFFFF
F2h	DISABLE_SECURITY (MFR_SPECIFIC_34)	n/a	n/a	n/a	Security is disabled by default
F3h	GPIO_SEQ_CONFIG (MFR_SPECIFIC_35)	n/a	n/a	n/a	Powergood pin is configured by default
F4h	MFR_SECURITY_BIT_MASK (MFR_SPECIFIC_36)	n/a	n/a	n/a	Default bit mask is to have no commands secured.
F5h	TEMP_BALANCE_IMIN (MFR_SPECIFIC_37)	-511.5	511.5	511.5	
F6h	LIGHT_LOAD_LIMIT_LOW (MFR_SPECIFIC_38)	-511.5	511.5	0	
F7h	FAST_OC_FAULT_LIMIT (MFR_SPECIFIC_39)	0	511.5	0	
F8h	POWER_GOOD_CONFIG (MFR_SPECIFIC_40	0	15	0x0F	
F9h	MFR_SPECIFIC_41				
FAh	PHASE_DROP_CAL (MFR_SPECIFIC_42)	-127	127	1.0	
FBh	SYNC_OFFSET (MFR_SPECIFIC_43)	0	32751	0	
FCh	EADC_TRIM (MFR_SPECIFIC_44)	0	63	128	Values >63 will be ignored and replaced by factory-trimmed value from INFO_FLASH.
	VID_RESTART (MFR_SPECIFIC_44)	n/a	n/a	n/a	
FDh	DEVICE_ID (MFR_SPECIFIC_45)				
FEh	Mfr_Specific_Extended_Command				
FFh	PMBUS_Extended_Command				

14 Usage: Assigning DPWM Outputs to Phases and Pages

The UCD9240 contains four DPWM units, each with two output phases. Multiple DPWM units may be bridged together to form output rails with higher output current and reduced ripple. This section describes the command used to define the relationship between the DPWM units (the phases) and the output rails (the pages).

While there is significant flexibility in terms of mapping power stages to output rails, the differential voltage feedback signals (EAP/EAN) cannot be re-mapped through any commnds, and therefore, must be connected to the proper input on the circuit board. Because the EADC sample trigger for a given front end stage is derived from the ramp timer of the first (lowest numbered) DPWM on the rail, the system must ensure that the number of the EADC and the number of the first DPWM match. For example, consider a two rail configuration in which 4 power stages (1A, 2A, 1B and 2B) are assigned to the first rail and 2 power stages (3A and 4A) to the second. The first DPWM on the first rail is 1; its voltage feedback must be through EAP1/EAN1. The first DPWM on the second rail is 3; its voltage feedback must be through EAP3/EAN3. (In this configuration EAP2/EAN2 and EAP4/EAN4 are unused and are disabled to reduce unnecessary power consumption.)

14.1 (D2h) PHASE_INFO (MFR_SPECIFIC_02)

The PHASE_INFO is used to define the relationship between the DPWM units (the phases) and the output rails (the pages).

Byte Number	Description
0	CMD = D2
1	BYTE_COUNT=4
2	Page 0 Phase Info
3	Page 1 Phase Info
4	Page 2 Phase Info
5	Page 3 Phase Info

Each data byte in the PHASE_INFO string is a bit-field that contains the phase mapping information for one page. Setting a bit to '1' assigns that DPWM output to that page.

Bit	7	6	5	4	3	2	1	0
DPWM Output	4B*	4A	3B*	3A	2B	2A	1B	1A

(*) DPWM outputs 3B and 4B are not supported on the 64-pin version of the UCD9240.

Many system parameters may be determined from the PHASE_INFO string.

The number of pages can be determined by counting the number of non-zero bytes in the string.

The number of phases for each page can be determined by counting the number of non-zero bits within each byte.

Note: This command affects many internal register settings, many of which could damage the power supply load if not set correctly. To ensure that they are all initialized correctly after a PHASE_INFO command has been issued all output voltage rails are locked in the OFF setting until after a power-cycle. The STORE_DEFAULTS_ALL command should be issued first so that the new settings are stored for use after the power-cycle.

14.1.1 Validity Checking:

The PHASE_INFO command will perform some validity checks on the phase info string, looking for invalid combinations that the hardware cannot support. These are the rules that are enforced:

1. No phase may be controlled by more than one page.
2. Each DPWM unit has two outputs (A and B). It is not required that both outputs be used, but if they are, they both must be controlled by the same page.
3. Page numbering starts from zero, and increases in consecutive order with no gaps allowed.
4. DPWM outputs 3B and 4B do not exist on the 64-pin version of the UCD9240. Selecting these outputs is not allowed.

An invalid configuration will be rejected, and will be indicated by the INVALID_DATA bit in the STATUS_CML register and the CML bit in the STATUS_BYTE and STATUS_WORD registers.

14.1.2 Examples of Valid Configurations

Example 1 (2+2+1+1):
Four pages, consisting of two dual-phase and two single-phase pages
PhaseInfo = [000 0011, 0000 1100, 0001 0000, 0100 0000]

DPWM Output	Page	Phase
1A	0	0
1B	0	1
2A	1	0
2B	1	1
3A	2	0
3B	Unused	Unused
4A	3	0
4B	Unused	Unused

Example 2 (4+4+0+0):
Two pages with four phases per page
PhaseInfo = [0000 1111, 1111 0000, 0000 0000, 0000 0000]

DPWM Output	Page	Phase
1A	0	0
1B	0	1
2A	0	2
2B	0	3
3A	1	0
3B	1	1
4A	1	2
4B	1	3

Example 3 (8+0+0+0):
One page with eight phases
PhaseInfo = [1111 1111, 0000 0000, 0000 0000, 0000 0000]

DPWM Output	Page	Phase
1A	0	0
1B	0	1
2A	0	2
2B	0	3
3A	0	4
3B	0	5
4A	0	6
4B	0	7

14.1.3 Examples of Invalid Configurations

Shaded text indicates the invalid portion of the command string.

Example 4 (4+2+2+2):
One quad-phase and three dual-phase outputs
PhaseInfo = [0000 1111, 0000 1100, 0011 0000, 1100 0000]

DPWM Output	Page	Phase
1A	0	0
1B	0	1
2A	0 & 1	2 or 0
2B	0 & 1	3 or 1
3A	2	0
3B	2	1
4A	3	0
4B	3	1

This configuration is invalid because two different pages are trying to control the same DPWM outputs, 2A and 2B, in violation of Rules #1 and #2. In addition, the total number of phases exceeds the number of outputs available.

Example 5 (3+3+1+1):
Two three-phase outputs and two single-phase outputs
PhaseInfo = [0000 0111, 0011 1000, 0100 0000, 1000 0000]

DPWM Output	Page	Phase
1A	0	0
1B	0	1
2A	0	2
2B	1	0
3A	1	1
3B	1	2
4A	2	0
4B	3	0

This configuration is invalid because DPWM Units 2A and 2B are being controlled by two different pages, which is not allowed by Rule #2.

Example 6 (4+0+4+0):
Two quad-outputs with a gap in the spacing
PhaseInfo = [0000 1111, 0000 0000, 1111 0000, 0000 0000]

DPWM Output	Page	Phase
1A	0	0
1B	0	1
2A	0	2
2B	0	3
3A	2	0
3B	2	1
4A	2	2
4B	2	3

This configuration is invalid because it has gaps in the page numbering

15 Usage: Setting Control Law Accelerator Gains

The UCD92xx digital power supply controller contains dedicated Control Law Accelerator (CLA) hardware to calculate the PWM output duty cycle for a switching power supply. The CLA uses a mathematical formula that is based on the mismatch between the target voltage and the measured voltage, as well as previous output duty cycles. Each of these terms is multiplied by an appropriate control gain and combined to form a PWM duty cycle.

The control gains that set the relative weighting of each of these terms are stored in banks. The CLA hardware has two banks, one active and the other inactive. By switching one bit it is possible to switch control from one bank to the other.

It is possible for a unit to have more than two banks of gains, for use during different operating conditions. All the gain banks are stored in the controller's RAM memory. To use a gain bank that was stored in RAM, the values are copied from RAM into the inactive CLA hardware bank and then the bit is switched to make that new bank active. The UCD92xx firmware handles this automatically.

The UCD92xx supports up to 4 independent output voltage rails, and each rail has its own CLA. The PMBus PAGE command is used to select the rails to be addressed by subsequent PMBus commands.

Special manufacturer-specific commands are used to move one bank at a time into RAM or the CLA hardware.

(D4h) CLA_BANK (MFR_SPECIFIC_04)
(D5h) CLA_GAINS (MFR_SPECIFIC_05)
(D8h) ACTIVATE_CLA_BANK (MFR_SPECIFIC_08)

The PMBus commands STORE_DEFAULT_ALL and RESTORE_DEFAULT_ALL are used to copy the values from RAM to Flash and vice-versa.

15.1 (D4h) CLA_BANK (MFR_SPECIFIC_04)

This Read/Write Byte command sets the bank number for the CLA gains. Different gains can be used for different modes of operation.

For example, one set of gains could be used for regulating at a constant output voltage (Run Mode). These gains would typically be chosen to have very high gain at high frequencies to reject noise and voltage disturbances.

A separate set of gains might be used during soft start when the target voltage is slowly stepping upward or during soft stop when the target voltage is stepping down (Ramp Mode). These gains would typically be chosen with lower bandwidth so that the steps in the target voltage blur together into a smooth ramp. The use of lower bandwidth gains in Ramp Mode also reduces the likelihood of saturating the PWM duty cycle beyond its range of 0 to 100%. Saturation in the duty cycle can lead to large slow transients in the output voltage, which is highly undesirable.

A third set of gains may be needed when operating in Light Load mode when some output phases are disabled.
Additional sets of gains could be used for responding to large transients, or manufacturer testing operations, or for experimentation.

The CLA_BANK value must be set before the CLA_GAINS (MFR_SPECIFIC_05) command is issued so that the CLA gains will be placed in the proper RAM or hardware register location.

Byte Number	Description
0	CMD = D4
1	Bank

BANK	Typical Usage
0x00	Gains used during Run Mode
0x01	Gains used during Ramp-up and Ramp-down Modes.
0x02	Gains used during Light Load Mode.
0x03 to 0xFD	Reserved for future use. Not valid for initial release of UCD92xx due to memory constraints.
0xFE	CLA hardware active bank. Not valid for writes.
0xFF	CLA hardware inactive bank.

15.2 (D5h) CLA_GAINS (MFR_SPECIFIC_05)

This Read/Write Block command is used to write or read the settings to be loaded into the CLA registers of the UCD92xx.

If the CLA_BANK is 0, 1, or 2, then the CLA_GAINS will be written directly to the appropriate location in RAM corresponding to the current PAGE and CLA_BANK on a CLA_GAINS write. Similarly, a CLA_GAINS read command will return the current values stored in RAM for the current PAGE and CLA_BANK if CLA_BANK is 0, 1 or 2.

If the CLA_BANK is 0xFF, then the CLA_GAINS will be written or read from the inactive bank of hardware registers for the current PAGE.

If the CLA_BANK is 0xFE, then the CLA_GAINS will be read from the active bank of hardware registers for the current PAGE. A CLA_GAINS write command with CLA_BANK set to 0xFE is invalid and will return a PMBUS_INVALID_DATA error NACK.

Byte Number	Description	Hardware Register	
		Register	Bits
0	CMD = D5		
1	BYTE_COUNT = 24		
2	B01 high byte	FLTRCOEF1	31:24
3	B01 low byte	FLTRCOEF1	23:16
4	B11 high byte	FLTRCOEF1	15:8
5	B11 low byte	FLTRCOEF1	7:0
6	B21 high byte	FLTRCOEF2	31:24
7	B21 low byte	FLTRCOEF2	23:16
8	COEF_SCALER high byte	FLTRCOEF2	15:8
9	COEF_SCALER low byte	FLTRCOEF2	7:0
10	A11 high byte	FLTRCOEF3	15:8
11	A11 low byte	FLTRCOEF3	7:0
12	A21 high byte	FLTRCOEF3	31:24
13	A21 low byte	FLTRCOEF3	23:16
14	B12 high byte	FLTRCOEF4	31:24
15	B12 low byte	FLTRCOEF4	23:16
16	A12 high byte	FLTRCOEF4	15:8
17	A12 low byte	FLTRCOEF4	7:0
18	FLTRNLR1 byte3 (*)	FLTRNLR1	31:24
19	FLTRNLR1 byte2 (*)	FLTRNLR1	23:16
20	FLTRNLR1 byte1 (*)	FLTRNLR1	15:8
21	FLTRNLR1 byte0 (*)	FLTRNLR1	7:0
22	LIMIT3	FLTRNLR2	31:24
23	LIMIT2	FLTRNLR2	23:16
24	LIMIT1	FLTRNLR2	15:8
25	LIMIT0	FLTRNLR2	7:0

(*) FLTRNLR1 is a 32-bit word whose fields do not land on convenient byte boundaries. It is used to control the gain of the EADC. See section 15.5 for more details.

Name	Size	Bit Numbers
AFE_GAIN	2	31:30
NOM_GAIN_MULT	6	29:24
POS_MID_GAIN_MULT	6	23:18
POS_LRG_GAIN_MULT	6	17:12
NEG_MID_GAIN_MULT	6	11:6
NEG_LRG_GAIN_MULT	6	5:0

15.3 (D8h) ACTIVATE_CLA_BANK (MFR_SPECIFIC_08)

This Write Word command selects which bank of CLA gains currently stored in RAM to load into the hardware registers of the UCD92xx. If bank is set to 0xFF, this command will not access CLA gains from RAM, but instead swap between the active and inactive gains currently loaded in the hardware registers.

Byte Number	Description
0	CMD = D8
1	Bank

Note:
The Bank parameter in this command does NOT modify the global PMBus CLA_BANK variable.

15.4 CLA Usage Examples

15.4.1 Example 1: Writing and then Selecting the Run Mode CLA gains for Page 0

Writing the CLA gains to RAM

1. Issue a PAGE command to set page = 0.
2. Issue a CLA_BANK command to set bank = 0 (Run Mode).
3. Issue a CLA_GAINS command to write the values of the CLA gains table into the appropriate location of RAM for Page 0, Mode 0.

Selecting the CLA gains to be loaded into the hardware registers.

1. Issue an ACTIVATE_CLA_BANK command with page = 0 and bank = 0.

15.4.2 Example 2: Writing and then Selecting the Ramp Mode CLA gains for Page 2

Writing the CLA gains to RAM

1. Issue a PAGE command to set page = 2.
2. Issue a CLA_BANK command to set bank = 1 (Ramp Mode).
3. Issue a CLA_GAINS command to write the values of the CLA gains table into the appropriate location of RAM for Page 2, Mode 1.

Selecting the CLA gains to be loaded into the hardware registers.

1. Issue an ACTIVATE_CLA_BANK command with page = 2 and bank = 1.

15.4.3 Example 3: Setting the CLA gains for Page 3 Directly to Hardware without Storing in RAM

Writing the CLA gains to Hardware Registers

1. Issue a PAGE command to set page = 3.
2. Issue a CLA_BANK command to set bank = FF (Inactive Hardware Bank).
3. Issue a CLA_GAINS command to write the values of the CLA gains table into the hardware registers for the inactive bank of Page 3.

Swap the active and inactive hardware banks.

1. Issue an ACTIVATE_CLA_BANK command with page = 2 and bank = FF.

15.4.4 Example 4: Swapping Between Active and Inactive Banks of Page 2 without Loading New Gains

1. Issue an ACTIVATE_CLA_BANK command with page = 2 and bank = FF.

15.5 AFE Gain

The analog gain of the EADC input can be adjusted to trade off resolution vs. range when measuring the feedback voltage. The analog front end (AFE) gain is set in the top two bits of the FLTRNLR1 register. The resolution and range of the EADC depend on the AFE gain setting, as shown in this table. A high AFE gain setting will have the best resolution but may saturate during a large transient. Lower AFE gain settings will prevent this saturation, at the expense of fine resolution. Typically, the higher-resolution settings (larger setting numbers) are used while regulating near the commanded setpoint voltage, while the wider-range settings (smaller setting number) are used during soft-start and soft-stop.

Setting	Analog Gain	Resolution (*)	Range(*)
0	1x	8 mV	-256 to + 248 mV
1	2x	4 mV	-128 to +124 mV
2	4x	2 mV	-64 to +62 mV
3	8x	1 mV	-32 to +31 mV

(*) The resolution and range are quoted at the feedback voltage sense inputs. The resolution and range at the output terminals will be larger by a factor equal to the feedback resistor divider ratio.

15.6 Non-Linear EADC Gains

The digital gain of the EADC input may be configured as a non-linear function of the feedback error voltage. This allows the controller to respond differently to large transients than it would for small deviations near the set-point. The dynamic range of the output of the error ADC is divided into 5 programmable segments where each segment can have a unique programmed gain. The 5 error value segments are defined by 4 limit registers. Note that the limit register values do not have to be symmetric around zero. A schematic diagram of the function is shown in Figure 2.

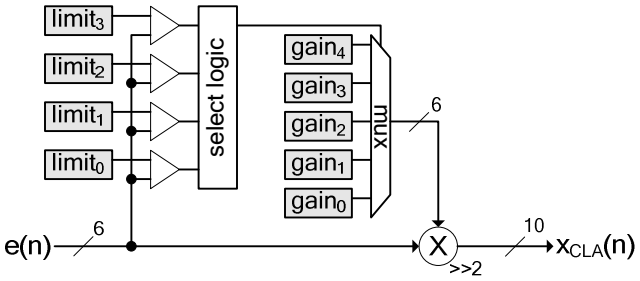

Figure 2: Nonlinear Gain Boost Implementation

The non-linear digital gains are controlled by the FLTRNLR1 and FLTRNLR2 registers. FLTRNR2 holds four 6-bit boundary values that mark the edge of each of the five ranges. FLTRNLR1 contains five 6-bit gain values used in each input range. Each 6-bit gain value is stored in a format with 4 bits of integer and 2 bits of fraction. This, a gain of 1.0 would be stored using a value of 4. Each gain value may range from 0.25X to 15.75X.

The following table shows the relationship between the non-linear input voltage limits and the non-linear input gain multiplier.

Name	Register	Bits	Name	Register	Bits
AFE_GAIN	FLTRNLR1	31:30			
			POS_LRG_GAIN_MULT	FLTRNLR1	17:12
LIMIT3	FLTRNLR2	29:24			
			POS_MID_GAIN_MULT	FLTRNLR1	23:18
LIMIT2	FLTRNLR2	21:16			
			NOM_GAIN_MULT	FLTRNLR1	29:24
LIMIT1	FLTRNLR2	13:8			
			NEG_MID_GAIN_MULT	FLTRNLR1	11:6
LIMIT0	FLTRNLR2	5:0			
			NEG_LRG_GAIN_MULT	FLTRNLR1	5:0

16 Glossary

Term	Meaning
ACK	Acknowledge – Indicates that the PMBus has received the message correctly.
ADC	Analog to Digital Converter – Converts analog voltages to digital counts that may be used for monitoring or control. The UCD92xx contains two ADCs: o eADC – Measures the differential voltage feedback error at high speed over a narrow range around the setpoint. o ADC12: Measures wide ranging signals at a much slower rate.
AFE	Analog Front End – A variable-gain differential amplifier used to sense the feedback error voltage. The output of the AFE is fed to the eADC for use in controlling the output voltage.
CLA	Control Law Accelerator – A mathematical engine in the UCD92xx controller that performs the digital control law calculations.
DAC	Digital-to-Analog Converter – A DAC is used to control the output setpoint voltage in the UCD92xx.
dPWM	Digital PWM – The dPWM output pins drive the power stages to generate output voltages that are regulated by the UCD92xx controller. The UCD92xx also has some general purpose PWM outputs that may be configured for other purposes, such as controlling a fan. The 'd' in dPWM helps to distinguish between the general purpose outputs and the power stage control outputs.
DFlash	Data Flash Memory – Non-volatile memory used for storing PMBus settings. The values in DFlash are automatically copied to RAM during wakeup.
eADC	Error A/D Converter – Measures the voltage feedback error signal, the difference between the output voltage and the reference voltage. This signal is fed to the CLA once per switching cycle.
NACK	Non-Acknowledge – An error has occurred in the PMBus message transfer.
PFlash	Program Flash Memory – Non-volatile memory used for the UCD92xx main firmware.
PMBus	Power Management Bus – An open-standard protocol that defines a means of communicating with power conversion devices using an I2C physical interface.
PWM	Pulse Width Modulation or Pulse Width Modulator
RAM	Random Access Memory- Volatile memory used to hold PMBus settings and internal variables. PMBus settings will be lost after a reset unless they are stored to Data Flash.
ROM	Read-Only Memory – Non-volatile memory used for the UCD92xx boot algorithms and some common data tables.

17 Document Revision History

Rev	Date	Initials	Description of Document Change
1.01	07/10/2008	Hem	• Added Revision number to document footer. • Removed Page argument from ACTIVATE_CLA_BANK command. Changed command format from R/W Word to R/W byte and made it a paged command. • Added SEQ_TIMEOUT to paged fault log. • Changed fault log to make VIN_OV and VIN_UV be paged faults. • Indicate that the MFR_Ratings commands are no longer supported. • Indicate that the VOUT_TRIM command is no longer supported.
1.03	08/07/2008	Hem	Described difference between temperature and current for peak logs.
1.05	08/20/2008	KKN	Added Alert Response Protocol section, MFR_STATUS section, comments about Response code support and a table of ranges and limits. Fixed minor errors and clarified some wording in preparation for release.
1.06	10/02/2008	KKN	Added large table with range-checking and limits for each command
1.07	10/29/2008	KKN	Miscellaneous cleanup
1.08	09/25/2009	Hem	• Updated Table 1 with details about which products support each command. • Added stub sections for DRIVER_CONFIG, PHASE_DROP_CAL, and SYNC_OFFSET commands. Still needs to be filled in. • Broke GPIO_SEQ_CONFIG command into two sections: First-generation and Second-generation. Put stub text into second-gen section. Needs to be filled in.
1.09	1/22/2010	KKN	Filled in DRIVER_CONFIG, PHASE_DROP_CAL and SYNC_OFFSET sections. Filled in the second generation GPIO_SEQ_CONFIG section.
1.11	4/27/2010	KKN	Added section about Clearing a Shutdown due to a Fault
1.12	1/10/2011	KKN	Added UCD9211, UCD9212, UCD9222 and UCD9244 command descriptions
1.13	3/8/2011	Hem	Added VID_RESTART command.
1.14	6/29/2011	Hem	Added POWER_GOOD_CONFIG command. Updated VID_CONFIG to include 7- and 8-bit parallel VID modes. Removed references to UCD9244H. Added references to ‘9222A, ‘9244A, and ‘9244N.