UCD3138 Integrated Power Controller Family Errata

This document describes the known exceptions to functional specifications (advisories) for the UCD3138 integrated digital power controller family of devices.

UCD3138 Digital Power Supply Controller Technical Reference Manual

UCD3138 Highly Integrated Digital Controller for Isolated Power Data Sheet

This document may also contain usage notes. Usage notes describe situations where the behavior of the device may not match presumed or documented behavior. This may include behaviors that affect device performance or functional correctness.

Contents

List of Figures

List of Tables

1 Design Exceptions to Functional Specifications

Table 1 lists all usage notes and the applicable silicon revisions.

Table 1. Issues Found

Issue Name	Description	Solution or Workaround	Affected Devices
PMBUS - Clock High Time Out Status Bit	PMBUS does not properly report the “High Clock” status bit in the memory map when a PMBUS message is started by the host and then goes to a PMBUS Idle state.	Use the “Bus Idle “bit status to help detect status of external PMBus.	UCD3138, UCD3138064
ARM - JTAG Port	Disabled JTAG access at ROM boot up.	JTAG access can be enabled through existing Flash program or PMBus Interface command.	UCD3138
DPWM - Fault Turn Off & Burst mode	PSFB application. DPWM outputs can turn off to the wrong logic level when using the edge generation block in the DPWM.	Use firmware versus direct Fault or Analog Comparator turn off DPWM when using the edge generation functionality.	UCD3138
DPWM - LLC Pulse extension	LLC Application. Pulse extension and shoot through can be present when immediate Filter updates are presented to the DPMW.	Use synchronous sampling with 1 sample per cycle and in the first 500 ns of the period.	UCD3138, UCD3138064
PMBus Interface status after accessing reserved address	The PMBus interface hardware will not respond properly if a PMBus message to the UCD3138 tries to access the manufacturer-reserved address of 0x7f (or 0x7E on UCD3138128), then UCD3138 may hold data line low.	PMBUS message to UCD3138 must not try to access manufacturer-reserved address.	UCD3138, UCD3138064, UCD3138A64, UCD3138128, UCD3138A, UCD3138064A, UCD3138A64A, UCD3138128A
PMBus Interface status after Interrupted/Corrupted Messages	The PMBus interface hardware may not respond properly after interrupted messages or certain corrupted messages, if a PMBus message to the UCD3138 is halted in the middle of a byte, the UCD3138 bit counter may not be reset to zero. Therefore, the UCD3138 may not respond correctly to the next message.	If host is configured to retry message communication then that will be sufficient to restore proper PMBUS communication with UCD3138 under **all** circumstances. Normally the bit counter will be re-synchronized after the one incorrect response, so subsequent messages will be handled correctly. Additional transitions on the PMBus clock line will return the PMBus to normal functionality, in general.	UCD3138
SPI Controller	SPI controller in slave mode accepts only 4 bytes at a time. Speed in slave mode is up to 2 MHz.	Most applications use SPI in master mode. In slave mode the SPI is functional, just slower than in Master mode.	UCD3138064, UCD3138A64, UCD3138128, UCD3138064A, UCD3138A64A, UCD3138128A
PMBus clock low timeout	The Clock low timeout is longer than 35 ms, specified by SMBus/PMBus specifications.	Not critical in most systems. In case it is found important, it is recommended to use a timer to set the timeout precisely.	UCD3138, UCD3138064, UCD3138A64, UCD3138128, UCD3138A, UCD3138064A, UCD3138A64A, UCD3138128A
DPWM end of period interrupt firing continuously if DPWM is off	DPWM counter of zero is interpreted as end of period, even if the DPWM module is OFF(disabled).	Either set the DPWM outputs to GPIO-low instead of disabling the DPWM module, or alternatively disable the DPWM period interrupt before disabling the DPWM.	UCD3138, UCD3138064
UART Lockup after changing baud	Causes lockup at UART controller for 8.6 s.	Reset the UART controller after each change in baud rate setting.	UCD3138, UCD3138064, UCD3138A64, UCD3138128, UCD3138A, UCD3138064A, UCD3138A64A, UCD3138128A

2 Description and Solutions

2.1 PMBus Clock High Timeout Status Bit

Description:

The *Clock High Timeout* status bit does not always indicate a *Clock High Timeout* on the bus.

Solution:

- Use UNIT_BUSY bit in PMBST register instead
- If UNIT_BUSY stays high with no PMBus message traffic, reset hardware and firmware:
  - Set up firmware timer in timer interrupt to detect lockup
  - If it times out, reset PMBus peripheral, firmware
  - PMBus firmware clears timer whenever data is transmitted or received on bus, and when UNIT_BUSY bit is not active
  - Timer interval depends on PMBus host speed
  - 100 ms is generally good

2.2 Disabled JTAG Access at Power Up

Description:

The JTAG pins are multiplexed with other pin functionalities and can be selected through IOMUX.

Specifically, the value of JTAG_CLK_MUX_SEL bit-field inside the IOMUX register needs to be set to 0.

Table 2. Bits 7-6: JTAG_CLK_MUX_SEL - TCK Pin MUX Select

I/O Pin	0	1	2	3
TCK	TCK	TCAP	SYNC	PWM-0

The hardware sets the default value of JTAG_CLK_MUX_SEL to 2; therefore, JTAG is not accessible. This is done to provide code security when jumping straight to flash mode in a programmed device. In this case, the customer program can choose whether to enable JTAG or not.

On all devices except the UCD3138, when the checksum is invalid, during the ROM boot loader execution, the value of JTAG_CLK_MUX_SEL will be set to 0. This permits JTAG debugging and download in ROM mode.

Solution:

The value of JTAG_CLK_MUX_SEL can be set to zero using PMBus, either by sending the write word command, or by using the memory debugger in the GUI.

2.3 Fault and Burst Turn Off for PSFB

Description:

With Fault or Burst mode, sometimes 1 out of 4 DPWM pins may not shut off. The bridge leg with a fixed pulse will always shut off, but one of the moving leg switches may not turn off.

So after a fault, the power supply is safely shut off. Fault logic is before Edge Generation logic, as Figure 1 shows.

Some PSFB signals go low on a falling edge input. Fault will make all Edge Generation inputs go low If the relevant input is high, the output will go low, so fault is handled as expected – Figure 2.

If the relevant input is low, there is no falling edge to make the output go low, so it will stay high. – Figure 3

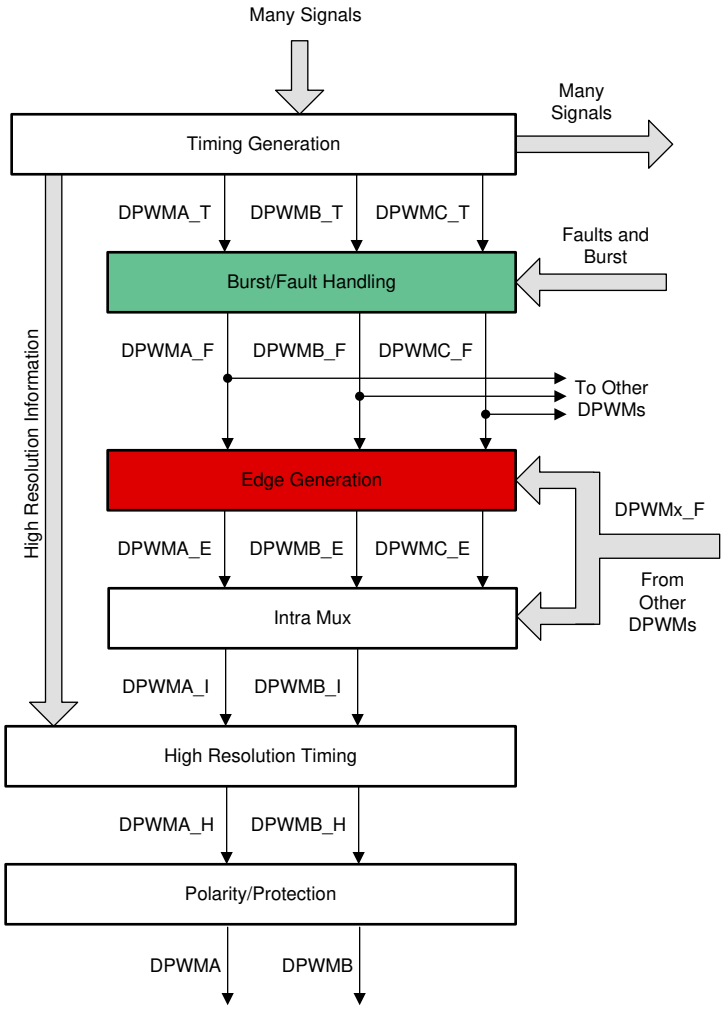

Figure 1. PWM Block Diagram

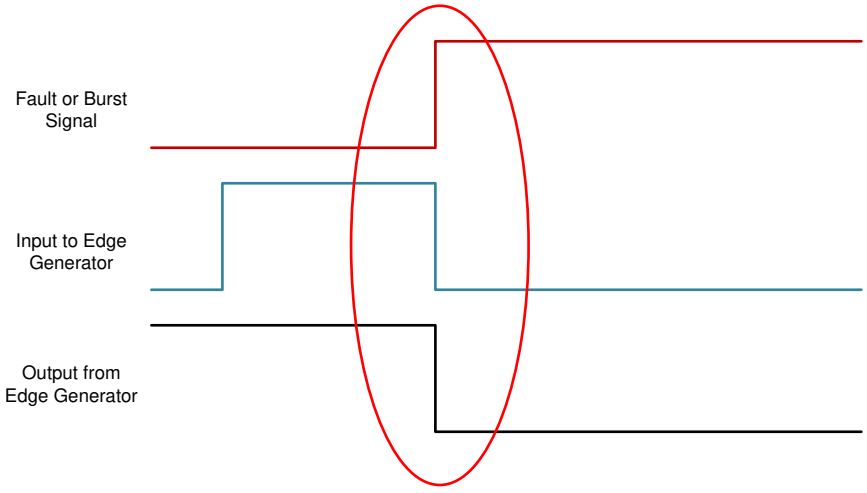

Figure 2. Fault or Burst Turns DPWM OFF

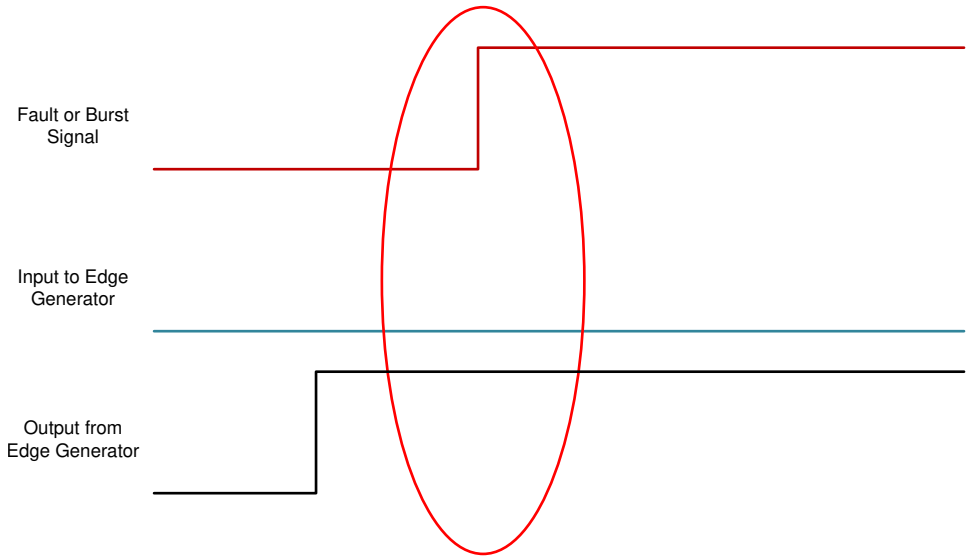

Figure 3. Fault or Burst Does not Turn DPWM OFF

Solution:

Use Firmware Interrupt to shut other DPWM pin off.

2.4 DPWM Pulse Extension

Description:

DPWM pulse extension in the LLC converter can occur under the following conditions, (1) the sample trigger is within 500 ns of the end of the frame, and (2) there is a large change in Y_N such that the converter transitions from multi-mode to resonant mode operation.

Solution:

Place the sample trigger (TRIG) near the beginning of the period or no later than 500 ns from the end of the shortest period that must be supported.

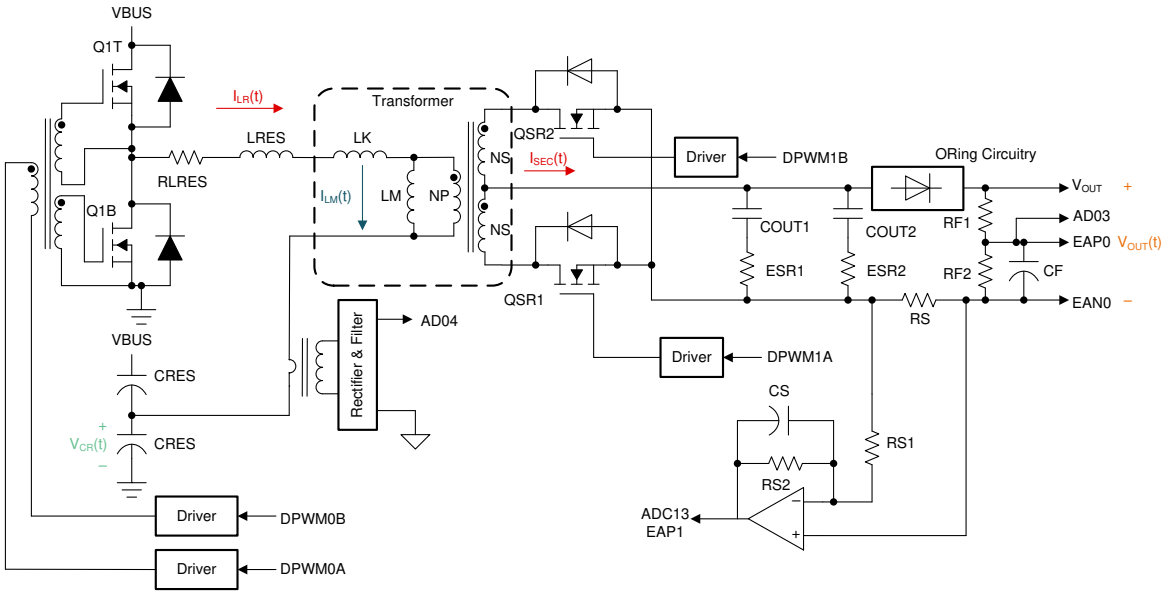

Figure 4. LLC Representative Schematic Diagram

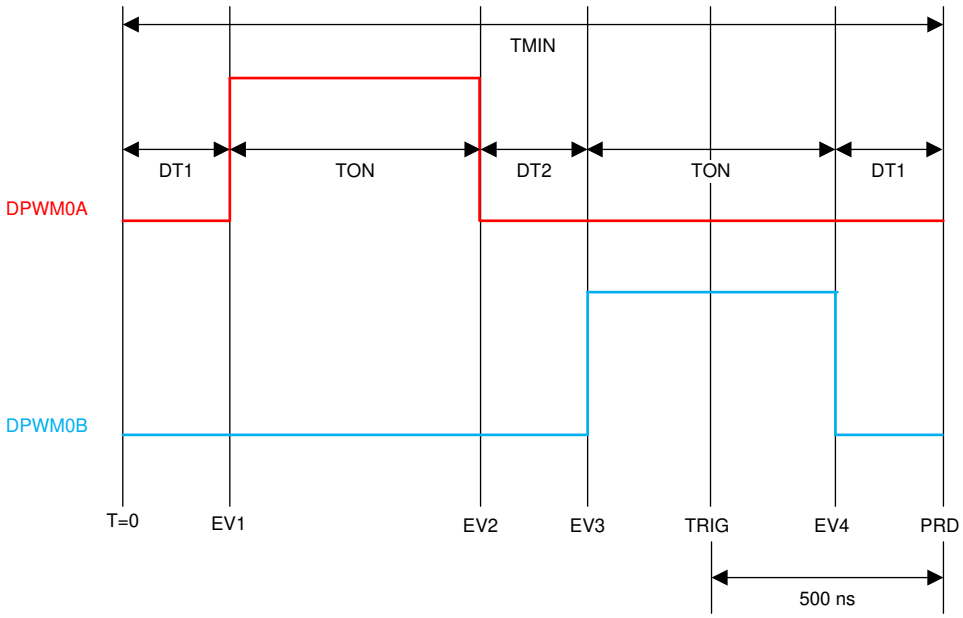

Figure 5. DPWM Representation

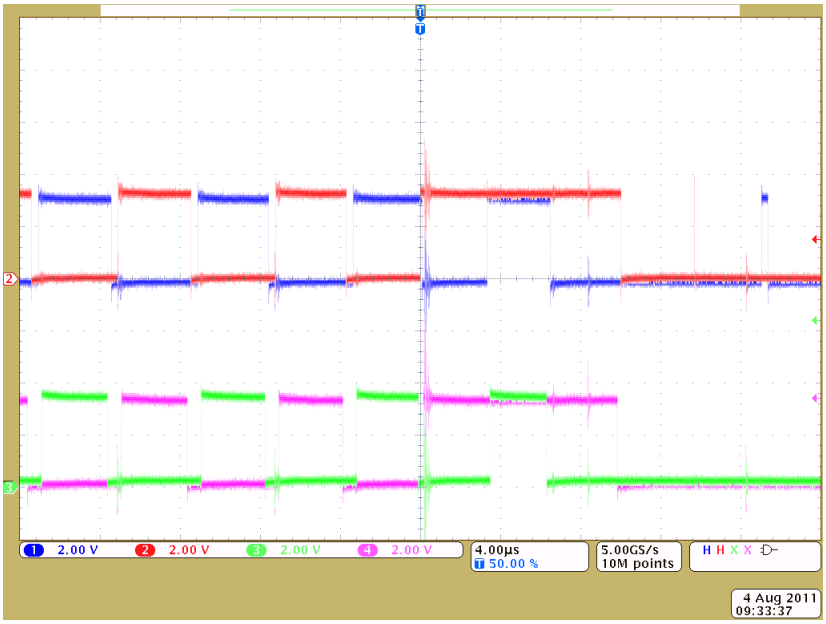

Figure 6. DPWM Scope Capture

2.5 PMBus Communication Accidental Access to Reserved Address

Description:

The PMBus interface hardware will not respond properly if a PMBus message to the UCD3138 tries to access the manufacturer-reserved address of 0x7F (or 0x7E on UCD3138128), then UCD3138 may hold the data line low. In this case, additional transitions on the PMBus clock line will return the PMBus to normal functionality.

Solution:

The PMBUS message to the UCD3138 device must not try to access the manufacturer-reserved address.

2.6 PMBus Communication Fault Handling

Description:

The UCD3138(064) PMBus interface hardware and firmware is designed to handle communication faults caused by glitches, interruptions, and stuck signals on the PMBus lines. In many cases, the system will recover immediately. In some cases, for example if one or both of the lines are stuck high or low, no communication will be possible. There are a few cases in which the system may require an additional message transmission for recovery. Two of cases follow:

1. If a PMBus message to the UCD3138(064) is halted in the middle of a byte, the UCD3138(064) bit counter may not be reset to zero. This occurs because the PMBUS Reset Bit does not effectively reset the bit counter in the UCD3138(064) PMBus hardware. Therefore, the UCD3138(064) may not respond correctly to the next message. Normally the bit counter will be resynchronized after this one incorrect response, so subsequent messages will be handled correctly.
2. The other case is when a communication frame is interrupted, and no stop bit was detected. In this case as well the UCD3138(064), and additional messages must be issued for recovery.

Solution:

If the host is configured to retry message communication, that will be sufficient to restore proper PMBUS communication with the UCD3138 device under **all** circumstances.

Normally the bit counter will be re-synchronized after the one incorrect response, so subsequent messages will be handled correctly.

In general, additional transitions on the PMBus clock line will return the PMBus to normal functionality.

2.7 PMBus and I2C Controllers Limitations

Description:

PMBus as Master cannot support the Block process call.

This is due to the fact that not enough of the bytes can be sent prior to the repeated start (up to 2 total supported by setting the EXT_CMD). The very same limitation will also prevent the implementation of combined random-read from bigger EEPROMs where there is need for more than 2 address bytes.

Solution:

For I2C to EEPROM connection, since you can not send a write/repeated start/read message with enough bytes in the write message, you send a write message with the proper number of bytes, including a stop at the end. Then you send a read message, and the EEPROM will start reading at whatever address was provided in the write message. Essentially, you replace the repeated start with a stop and a start.

There is no current solution to the previously mentioned limitations when acting as a master PMBus.

2.8 SPI Controller

Description:

The SPI controller in slave mode accepts only 4 bytes at a time. The speed in slave mode is up to 2 MHz.

Solution:

Most applications use SPI in master mode. In slave mode the SPI is functional, just slower than in the Master mode.

2.9 PMBus Clock Low Timeout

Description:

The Clock low timeout is longer than 35 ms, specified by SMBus and PMBus specifications.

Solution:

This is not relevant at the normal operation of the system and only happens at the time an exception occurred; therefore, it is not critical in most systems.

In case this is found important use a timer interrupt to set the timeout precisely.

2.10 DPWM end-of-Period Firing all Time if DPWM Turned off

Description:

If the DPWM counter is equal to zero, this is interpreted as end of period, and the DPWM will constantly invoke the end-of-period interrupt. This is the case even if the DPWM module is OFF (disabled).

In this case, the ISR consumes all the MIPS, and the threads lower the priority to starve and not be executed.

Solution:

Either set the DPWM outputs to GPIO-low instead of disabling the DPWM module.

For example:

```
Dpwm0Regs.DPWMCTRL1.bit.GPIO_A_EN = 1;
```

Or alternatively disable the DPWM period interrupt each time before disabling the DPWM module.

2.11 UART Lockup After Changing Baud Rate

Description:

Changing the UART baud rate in the middle of an operation is essential for implementing an auto-baud rate detection mechanism.

But baud rate change at the time during communication may cause a lockup of the UART controller for 8.6 s.

Solution:

Reset the UART controller after each change in the baud rate setting.

Make sure to time the resetting of the controller so it is executed at the times that there is no communication (transmission or reception).

If you cannot ensure that resetting may not happen during reception, a robust error handling mechanism (such as retransmission) needs to be in place.

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Original (January 2020) to A Revision	Page
• Changed TRIG period to 500 ns in the *Description* and *Solution* sections of the *DPWM Pulse Extension section*, and in *Figure 5*.	5

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265