EB633/D
Rev. 0, 1/2004

MDASM OPWM Software Considerations to Avoid Missing Pulses

Neil Farnham
TECD Applications

1. Abstract

The MPC5xx[1] family of devices contains a modular I/O sub-system (MIOS). This module consists of a library of flexible I/O and timer functions. One of the library functions is the double action sub-module (MDASM). The MDASM is a versatile 16-bit dual action function that can be configured to generate an output pulse width modulation (OPWM) signal.

In some cases of using the MDASM in OPWM mode, when modifying the leading edge to adjust the duty cycle of the PWM signal, the output of the OPWM channel will remain asserted when it is expected to be negated.

The duty cycle is of the PWM signal is defined as the ratio of signal high time for a given period.

This engineering note explains how this effect occurs and provides potential work around.

Note:
The scope of this engineering bulletin applies to all devices in the MPC5xx[1] family. However, if Errata AR_810 applies to your specific device similar considerations also need to be applied for updates to both leading and trailing edge generation. See Appendix A for Errata description.

This note should be read in conjunction with the User Manual for the specific MPC5xx[1] device.

2. Introduction

2.1. Background Information

The MDASM module is composed of two timing channels (A and B), an output flip-flop, an input edge detector and some control logic as shown in Figure 1.

[1] MPC5xx refers to all members of the MPC5xx family except the MPC505/9

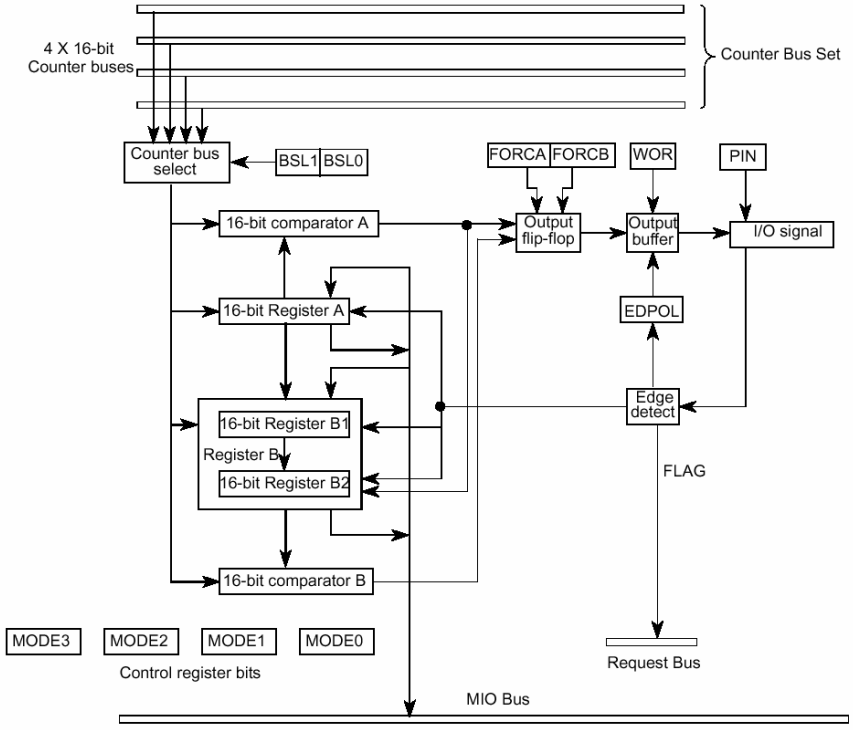

Figure 1

In PWM operation both channels (A and B) are used to generate one PWM output signal on the MDASM signal. Channels A and B define respectively the leading and trailing edges of the PWM output pulse. This can be seen below in Figure 2.

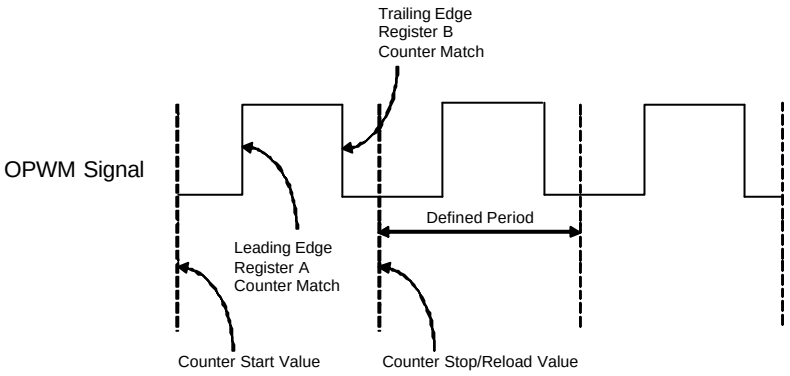

Figure 2

Channel B is doubled buffered and accessed via register B1. Register B2 is not accessible. The value in register B1 is transferred to register B2 each time a match occurs on either Channel A or Channel B.

The value loaded into Channel A match register is compared with the value on the 16-bit counter bus each time the counter bus is updated. When a match on Channel A occurs, the FLAG line is activated and the output flip-flop is set.

The value loaded into Channel B register B2 is compared with the value on the 16-bit counter bus each time the counter bus is updated. When a match occurs on Channel B, the output flip-flop is reset.

It is important to note the distinction between Channel A and Channel B. Channel A is NOT double buffered.

For devices effected by Errata AR_810, Channel B is also NOT double buffered and similar considerations should be applied to both Channel A and Channel B.

If both Channel A and B match registers are loaded with the same value, when a simultaneous match on Channel A and Channel B occurs, the sub-module behaves as if a simple match on Channel B had occurred except for the FLAG line which is activated. The output flip-flop is reset and the value in the Channel B register B1 is transferred to register B2 on the match.

2.2. Specific Issue

The problem arises when the Channel A match register of an OPWM is updated in the same system clock that a match on Channel A is expected, the match will be ignored, and the OPWM output will remain asserted until the match on the new value of A. (Note that the Channel A is not double buffered like the Channel B).

The problem sequence is:
(1) When the OPWM is enabled, the OPWM output is asserted when a match occurs on Channel A.
(2) This match may trigger a software task (e.g., via an interrupt on Channel A match) that then updates the Channel A match register. i.e., host writes a new compare value to both Channel A and Channel B match registers with a store word command.
(3) If the write to Channel A match register occurs during the same system clock that a match on Channel A is expected, the match will not be recognized.
(4) In this case the OPWM output will remain asserted until the next Channel A match that follows the next Channel B match.

This effect is shown in Figure 3.

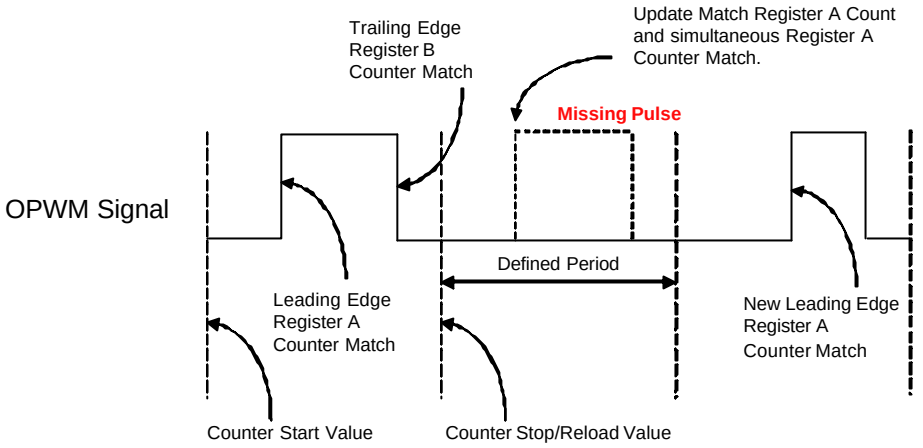

Figure3

This effect will not occur on the Channel B for devices with the Errata AR_810 fix as Channel B is doubled buffered with the B1 and B2 registers. This hardware ensures that the update only occurs after a match has been flagged.

3. Workarounds

Follow one of the following procedures:

(1) Keep the value in the Channel A match register (MDASMAR) constant and only perform writes to the Channel B match register (MDASMBR) to alter the pulse width and use a 16 bit data write access, to update only the Channel B match register (MDASMBR).

(2) Read the relevant counter value (such as MCSM) before doing the MDASMAR write. If the counter value is "just below" the old Channel A match register value, then the Channel A match register (MDASMAR) update should be delayed.

(3) Write to Channel A match register (MDASMAR), and then check if the value of the relevant counter (such as MCSM) is bigger than the OLD Channel A match register value. If so, it is required to force the pin value to it's desired state (FORCA).

(4) When using the MDASM interrupt to update the Channel A match register (MDASMAR), verify that the PWM pulse width is larger than the interrupt latency.

4. Appendix A

CDR_AR_810 Customer Erratum MIOS14.CDR3IMB3_01_0

MIOS: Synchronize writes to DASM B channel in OPWM mode.

DESCRIPTION: In some cases of using DASM in OPWM mode, when writing B register, the output of the OPWM channel will remain asserted when it was expected to be negated. When B register of an OPWM channel is updated in the same system clock that a match on B is expected, the match will be ignored, and the OPWM output will remain asserted until the match on the new value of B. The problem sequence is: (1) OPWM output is asserted when A match occurs. (2) This match may trigger a SW task (e.g., via interrupt on A match) that updates B register (B is double buffer in this mode, i.e., host writes a new compare value to B, the OPWM output negates when the timer matches to the original value of B, then the new value is copied to the comparator to be used in the next PWM cycle) (3) If the write to B occurs at the same system clock that B match is expected, the match will not be recognized. (4) In this case, the OPWM output will remain asserted until the next B match that follows the next A match.

WORKAROUND: Follow one of the following procedures: (1) read the relevant counter value (such as MCSM) before doing the DASM write. If the counter value is "just below" the old DASM B value, then the B register update should be delayed. (2) write to B register, and then check if the value of the relevant counter (such as MCSM) is bigger than the OLD B value. If so, it is required to force the pin value to it's desired state (FORCB). (3) When using DASM interrupt to update B register, verify that the PWM pulse width is larger than the interrupt latency. (4) keep the B value constant and only perform writes to the A register to alter the pulse width. In this case the value of A is updated after the interrupt from the previous channel A match. For this to work the pulse period needs to be greater than the interrupt latency, so that the new A value is written before the next A channel compare is enabled. (Note that the A register is not double buffered).

THIS PAGE INTENTIONALLY LEFT BLANK

THIS PAGE INTENTIONALLY LEFT BLANK

HOW TO REACH US:

USA / EUROPE / Locations Not Listed:

Motorola Literature Distribution
P.O. Box 5405
Denver, Colorado 80217
1-800-521-6274 or 480-768-2130

JAPAN:

Motorola Japan Ltd.
SPS, Technical Information Center
3-20-1, Minami-Azabu Minato-ku
Tokyo, 106-8573 Japan
81-3-3440-3569

ASIA/PACIFIC:

Motorola Semiconductors H.K. Ltd.
Silicon Harbour Centre
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
852-26668334

HOME PAGE:

http://motorola.com/semiconductors

MOTOROLA

EB633/D, Rev. 0