Application Report
SPRA647 - May 2000

Power Drive Protection Interrupt on the TMS320C24x

Hareesh Janakiraman *Digital Signal Processing Solutions*

ABSTRACT

The Power Drive Protection Interrupt ($\overline{\text{PDPINT}}$) pin is a safety feature that is built into all TMS320C24x™ DSPs. This feature protects the electrical drive from damage due to any electrical, mechanical or thermal abnormalities. When this pin is driven low externally, all the PWM outputs are put in a high impedance state, thereby protecting the electrical drive from further damage. A proper understanding of the functioning and limitations of the $\overline{\text{PDPINT}}$ module is necessary to realize the benefits of this feature.

Contents

List of Figures

1 Design Problem

How do I use the $\overline{\text{PDPINT}}$ pin in my design?

2 Introduction

The Power Drive Protection Interrupt ($\overline{\text{PDPINT}}$) pin is a safety feature that is built into all TMS320C24x devices. This feature protects the electrical drive from damage due to any electrical, mechanical or thermal abnormalities. When this pin is driven low externally, all the PWM outputs are put in a high impedance state, thereby protecting the electrical drive from further damage. A proper understanding of the functioning and limitations of the $\overline{\text{PDPINT}}$ module is necessary to realize the benefits of this feature.

3 Principle of Operation

The $\overline{\text{PDPINT}}$ is typically connected to a current/voltage sensor or a temperature sensor to check for abnormalities in the system. Whenever the $\overline{\text{PDPINT}}$ interrupt is asserted (by an external active-low signal), the FCOMPOE bit in COMCON register changes to zero. This action puts the PWM outputs in a high impedance state. To enable the PWM outputs again, this bit must be changed to one. The PWM outputs can also be forced into a high impedance state by the user writing a zero to the FCOMPOE bit. Therefore, there are two ways to put the PWM outputs into high impedance. $\overline{\text{PDPINT}}$ has a clock independent path. This means that the PWM channels are put in the high impedance state the moment $\overline{\text{PDPINT}}$ is taken low and remains in this state as long as $\overline{\text{PDPINT}}$ is maintained low.

TMS320C24x is a trademark of Texas Instruments.

4 State of $\overline{\text{PDPINT}}$ During Reset

It is important to maintain $\overline{\text{PDPINT}}$ in high state during a power-on reset or a warm reset. i.e., the DSP should not be powered on or reset with $\overline{\text{PDPINT}}$ low. If this happens, subsequent $\overline{\text{PDPINT}}$ interrupts are not recognized. That is, even if the $\overline{\text{PDPINT}}$ pin is taken high after the device has come up and subsequently driven low, the interrupt is not recognized. In this circumstance, clearing the $\overline{\text{PDPINT}}$ interrupt flag will not help either.

It must be emphasized that the action of putting the PWM outputs in high impedance when $\overline{\text{PDPINT}}$ is active always takes place. For example, even if PDPINT is low during power-up, the PWM outputs will be still be driven into high impedance since $\overline{\text{PDPINT}}$ has a clock independent path. However, subsequent valid $\overline{\text{PDPINT}}$ interrupts will not be recognized by the core.

5 $\overline{\text{PDPINT}}$ Behavior in TMS320C24x Silicon

$\overline{\text{PDPINT}}$ is level sensitive in TMS320F243/F241 (Revision 2.1) silicon. Therefore multiple interrupts will be asserted if $\overline{\text{PDPINT}}$ is held low on a continuos basis. This is not the case with TMS320C242 silicon, whose $\overline{\text{PDPINT}}$ pin is (falling) edge sensitive. The example program at the end of this application note aids in understanding the edge sensitive (or level sensitive) nature of the $\overline{\text{PDPINT}}$ pin.

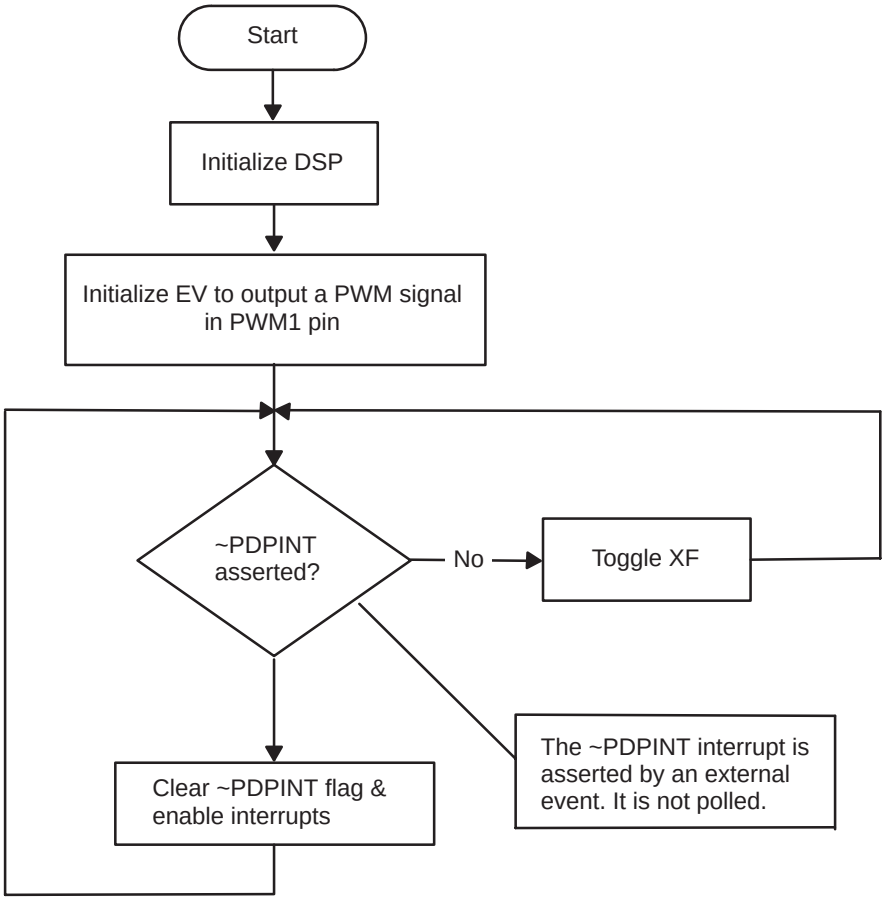

Figure 1. Flowchart for $\overline{\text{PDPINT}}$ Validating Program

6 Program to Check the $\overline{\text{PDPINT}}$ & PWM1 OF TMS320C24x

This program illustrates the behavior of $\overline{\text{PDPINT}}$ pin in 'C24x DSP. It produces an output in PWM1 Channel and aids in observing the effect of $\overline{\text{PDPINT}}$ on PWM1.$\overline{\text{PDPINT}}$ need not be enabled at the peripheral level, since it is done automatically at reset. Program normally keeps toggling the XF bit. If $\overline{\text{PDPINT}}$ is level sensitive and is asserted (driven and held low), XF toggling frequency will reduce due to the overhead in servicing the interrupt. If $\overline{\text{PDPINT}}$ is edge sensitive and is asserted (driven and held low), it will not alter the XF toggling frequency, since the interrupt will be recognized only once. However, in both cases when $\overline{\text{PDPINT}}$ is asserted, the PWM output should be put in a high impedance state, which can be verified by pulling the output high/low and monitoring the state of the pin by an oscilloscope.

```
      .title " pdpint"

      .include 24x.h    ; 24x.h contains all the register definitions

;-------------------------------------------------------------------------
; Vector address declarations
;-------------------------------------------------------------------------
      .sect  "vectors"

RSVECT      B    START        ; Reset Vector
INT1        B    GISR1        ; Interrupt Level 1
INT2        B    GISR2        ; Interrupt Level 2
INT3        B    GISR3        ; Interrupt Level 3
INT4        B    GISR4        ; Interrupt Level 4
INT5        B    GISR5        ; Interrupt Level 5
INT6        B    GISR6        ; Interrupt Level 6
RESERVED    B    PHANTOM      ; Reserved
SW_INT8     B    PHANTOM      ; User S/W Interrupt
SW_INT9     B    PHANTOM      ; User S/W Interrupt
SW_INT10    B    PHANTOM      ; User S/W Interrupt
SW_INT11    B    PHANTOM      ; User S/W Interrupt
SW_INT12    B    PHANTOM      ; User S/W Interrupt
SW_INT13    B    PHANTOM      ; User S/W Interrupt
SW_INT14    B    PHANTOM      ; User S/W Interrupt
SW_INT15    B    PHANTOM      ; User S/W Interrupt
SW_INT16    B    PHANTOM      ; User S/W Interrupt
TRAP        B    PHANTOM      ; Trap vector
NMI         B    NMI          ; Non-maskable Interrupt
;-------------------------------------------------------------------------
; M A C R O - Definitions
;-------------------------------------------------------------------------
KICK_DOG    .macro                ; Watchdog reset macro
            LDP   #00E0h          ; DP-->7000h-707Fh
            SPLK  #05555h, WDKEY
            SPLK  #0AAAAh, WDKEY
            LDP   #0h         ; DP-->0000h-007Fh
            .endm
            .text

START:      SETC  INTM            ; Disable interrupts
            SPLK  #0000h,IMR      ; Mask all core interrupts
            LACC  IFR             ; Read Interrupt flags
            SACL  IFR             ; Clear all interrupt flags
            LDP   #0E0h
```

```
                SPLK  #006Fh, WDCR      ; Disable WD if VCCP=5V
                KICK_DOG

                SPLK  #0h,60h           ; Set wait state generator for:
                OUT   60h,WSGR        ; Program Space, 0 wait states

                LDP   #0E1h             ; Set dual purpose pins for
                SPLK  #0FFFFh,OCRA      ; PWM function
                LDP   #0h
                SPLK  #0001h,IMR        ; Enable INT1 (for ~PDPINT)

                LDP   #EVIMRA >> 7      ; Clear all flags in EV (if set)
                LACL  EVIFRA            ;
                SACL  EVIFRA            ;
                CLRC  INTM              ; Enable interrupts globally

                LDP   #T1CON >>7        ; Generate a PWM signal in PWM1 pin
                SPLK  #0500, T1PR
                SPLK  #0250, CMPR1
                SPLK  #1000001000000000b, COMCON
                SPLK  #0000000000000010b, ACTR
                SPLK  #1001000001000010b, T1CON
                LDP   #0h

LOOP2           SETC  XF          ; Program normally keeps looping here
                CALL  DELAY       ; in LOOP2, unless interrupted by
                CLRC  XF          ; ~PDPINT. If PDPINT interrupt is
                CALL  DELAY       ; continuously asserted, then the
                B     LOOP2       ; frequency at which XF is toggled will
                                  ; be reduced, due to the ISR overhead

DELAY           LAR   AR0,#0FFFFh; The DELAY loop merely provides a delay
LOOP            RPT   #10h        ; for XF toggling.
                NOP
                BANZ LOOP
                RET

GISR1:          LDP   #PIVR >>7   ; ISR for PDPINT. Merely clears the
                LACL  PIVR        ; PDPINT flag
                LDP   #EVIMRA >> 7
                LACL  EVIFRA
                SACL  EVIFRA
                CLRC  INTM
                LDP   #0h
                RET

GISR2:          RET
GISR3:          RET
GISR4:          RET
GISR5:          RET
GISR6:          RET

PHANTOM         KICK_DOG          ;Resets WD counter
                B     PHANTOM

                .END
```

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgment, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Customers are responsible for their applications using TI components.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.