we innovate
nuvoTon

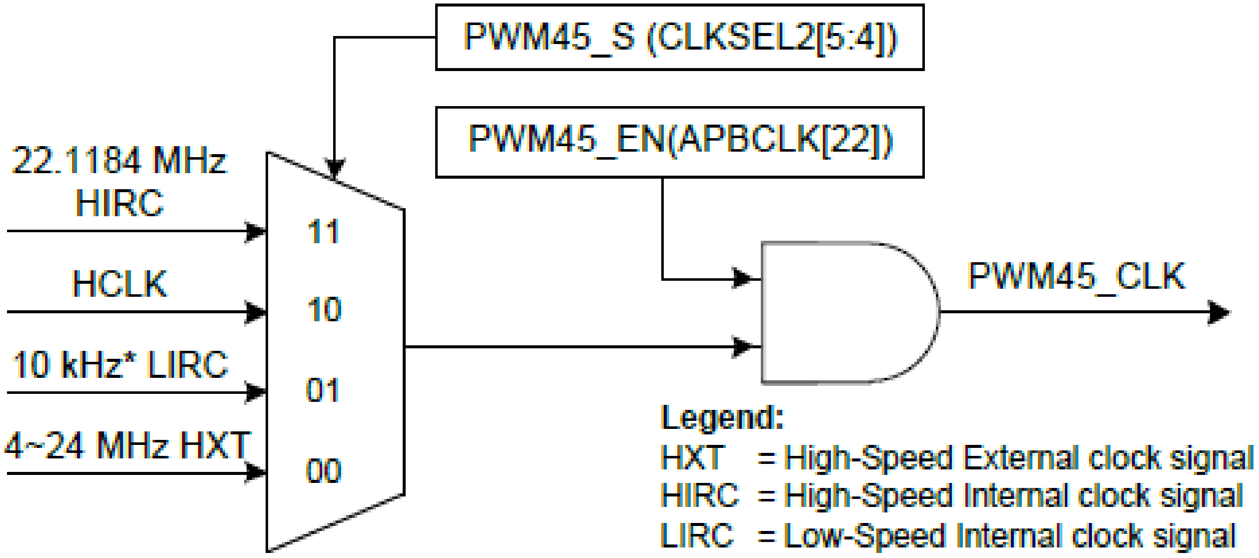
NuMicro^{TM} M051
PWM and Capture

NuMicro@nuvoton.com

Features of PWM

- **Four PWM Generators, each generator supports**
  - One 8-bit prescaler
  - One clock divider
  - Two PWM-timers for two outputs, each timer includes
    - A 16-bit PWM down-counter
    - A 16-bit PWM reload value register (CNR)
    - A 16-bit PWM compare register (CMR)
  - One dead-zone generator
  - Two PWM outputs.
- **8 PWM channels or 4 PWM paired channels.**
- **16 bits resolution.**
- **Support edge and center aligned modes**
- **Single-shot or Continuous mode PWM.**

PWM/Capture Clock Source



PWM45_S (CLKSEL2[5:4])

PWM45_EN(APBCLK[22])

22.1184 MHz
HIRC

HCLK

10 kHz* LIRC

4~24 MHz HXT

11

10

01

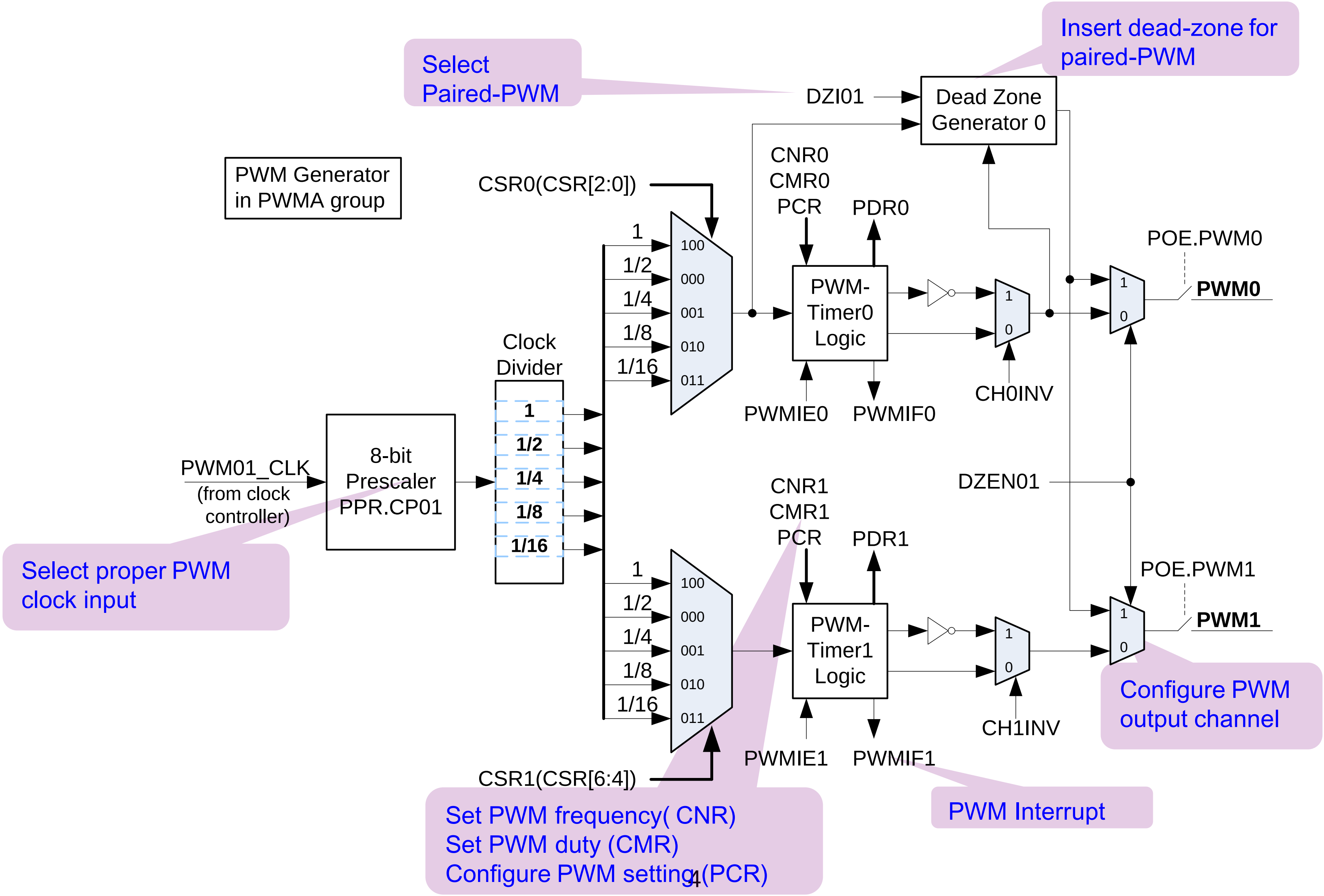
00

PWM45_CLK

Legend:
HXT = High-Speed External clock signal
HIRC = High-Speed Internal clock signal
LIRC = Low-Speed Internal clock signal

Note: 10 kHz selection is only available on M05xxDN/DE.

Configure PWM Generator



Select Paired-PWM

Insert dead-zone for paired-PWM

Select proper PWM clock input

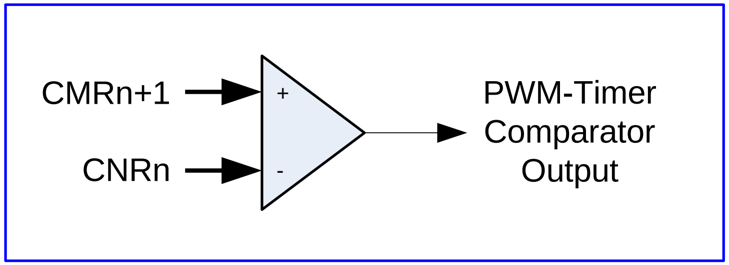
Configure PWM output channel

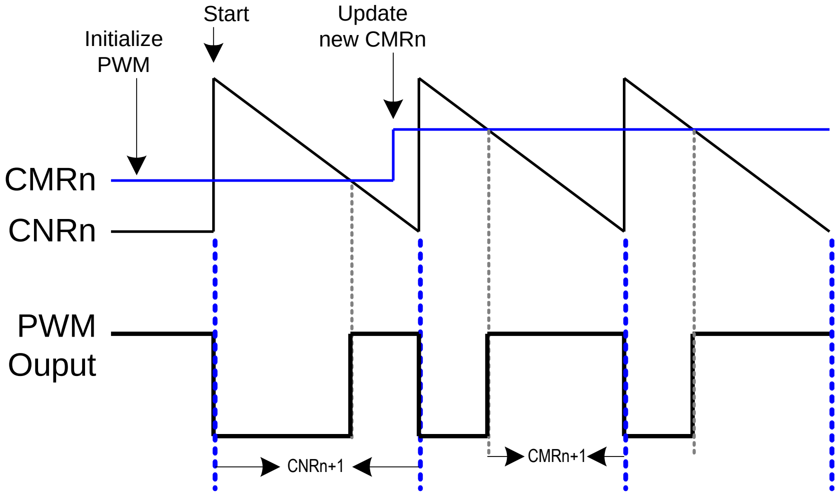
Set PWM frequency(CNR)
Set PWM duty (CMR)
Configure PWM setting (PCR)

PWM Interrupt

PWM Edge Align Mode

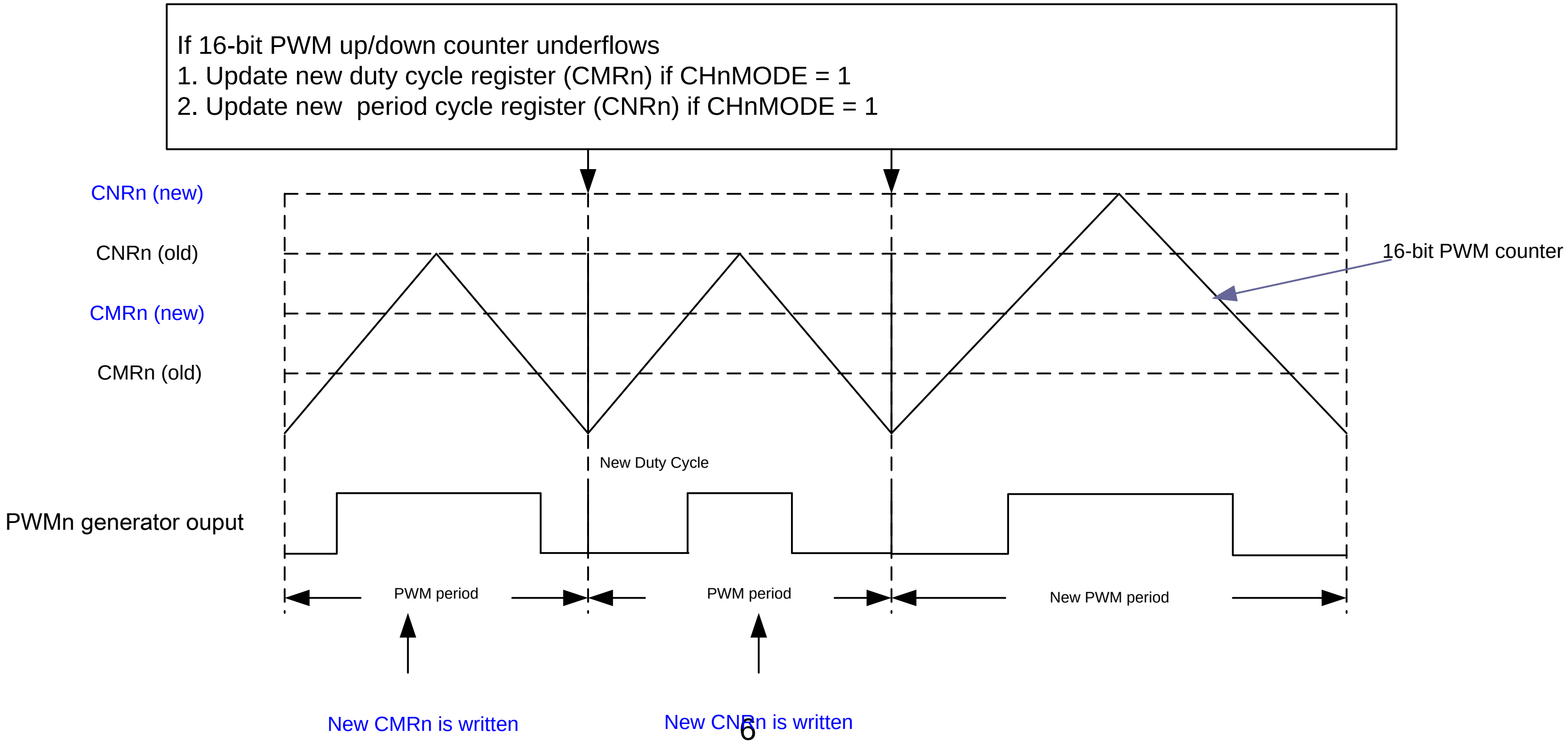
- Duty ratio = (CMR+1) / (CNR+1)
- Duty = (CMR+1) x (clock period)
- Period = (CNR+1) x (clock period)





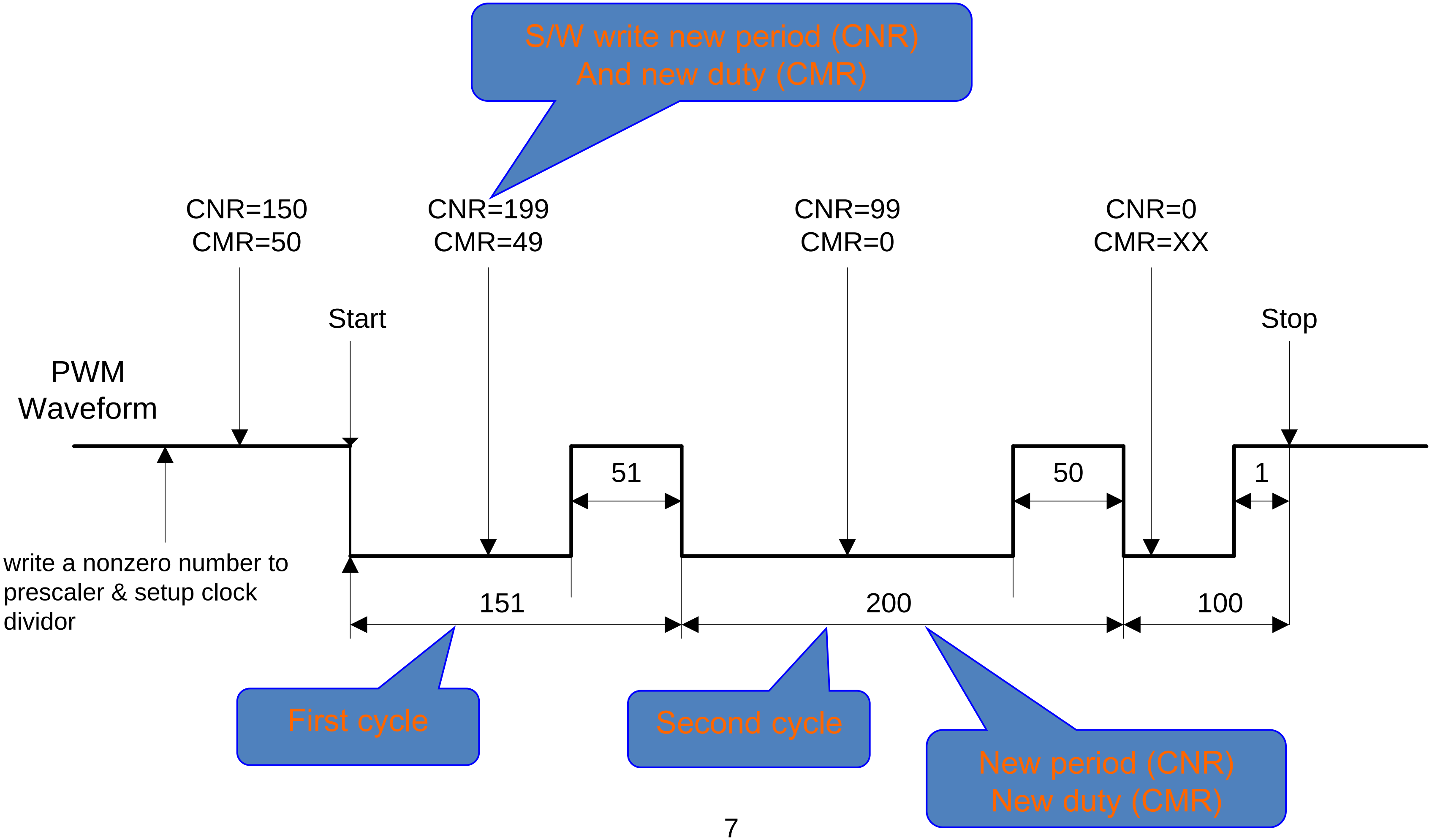
PWM Center Align Mode

- Duty ratio = (CNR - CMR) / (CNR+1)
- Duty = (CNR - CMR) x 2 x (clock period)
- Period = (CNR+1) x 2 x (clock period)



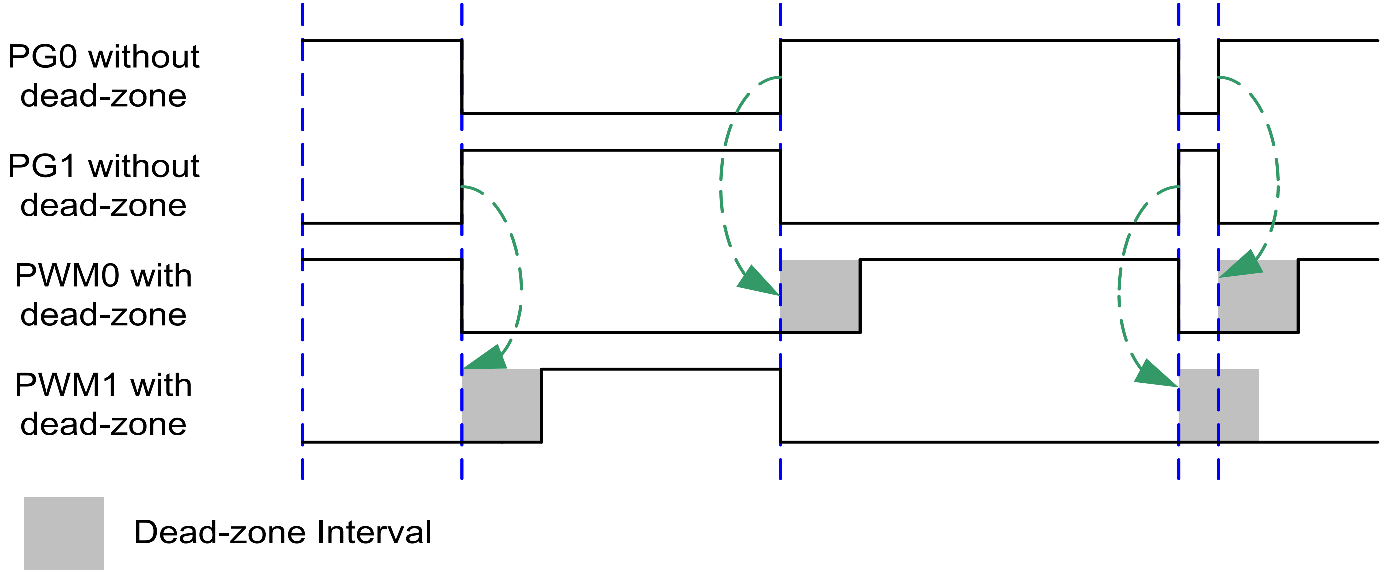
If 16-bit PWM up/down counter underflows
1. Update new duty cycle register (CMRn) if CHnMODE = 1
2. Update new period cycle register (CNRn) if CHnMODE = 1

PWM Double Buffering Illustration



Operation of Dead Zone Generator

- **Why need the dead zone control?**
  - To avoid a paired-PWM outputs overlapping on duty-on duration.
  - For example, in Motor Driver application, it needs to avoid the upper and lower power switch turn on simultaneously.
- **Insert a delay time (dead zone) before duty on at each channel of paired-PWM.**
- **8-bit dead-zone timer from PWM clock.**

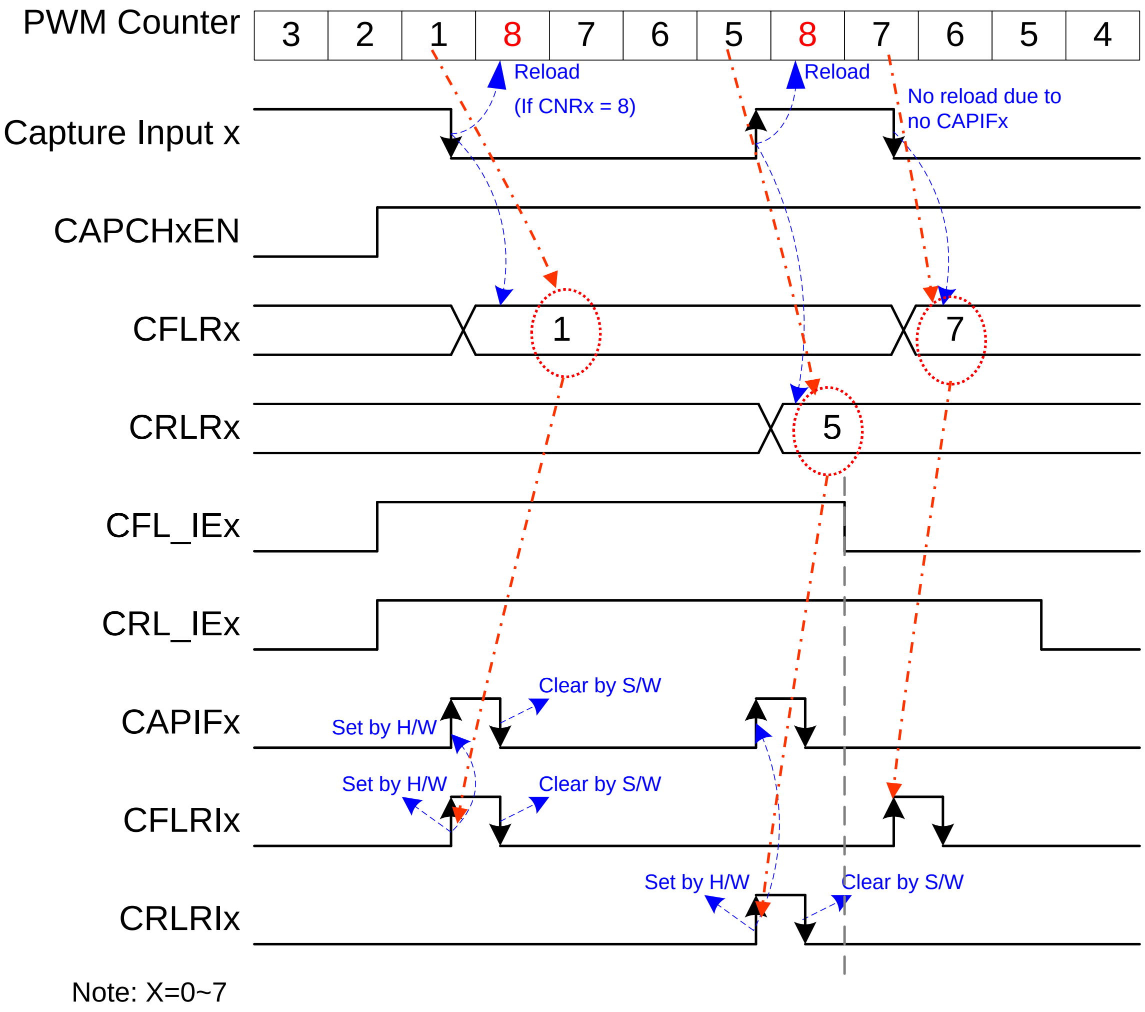


Effect of dead-zone for complementary pairs

Features of Capture Input

- Timing control logic shared with PWM Generators. (therefore up to 16 bits data length)
- 8 Capture inputs shared with PWM outputs
- Each channel supports
  - One rising latch register (CRLR)
  - One falling latch register (CFLR)
  - Capture interrupt flag (CAPIFx)

Operation Timing of Capture Input



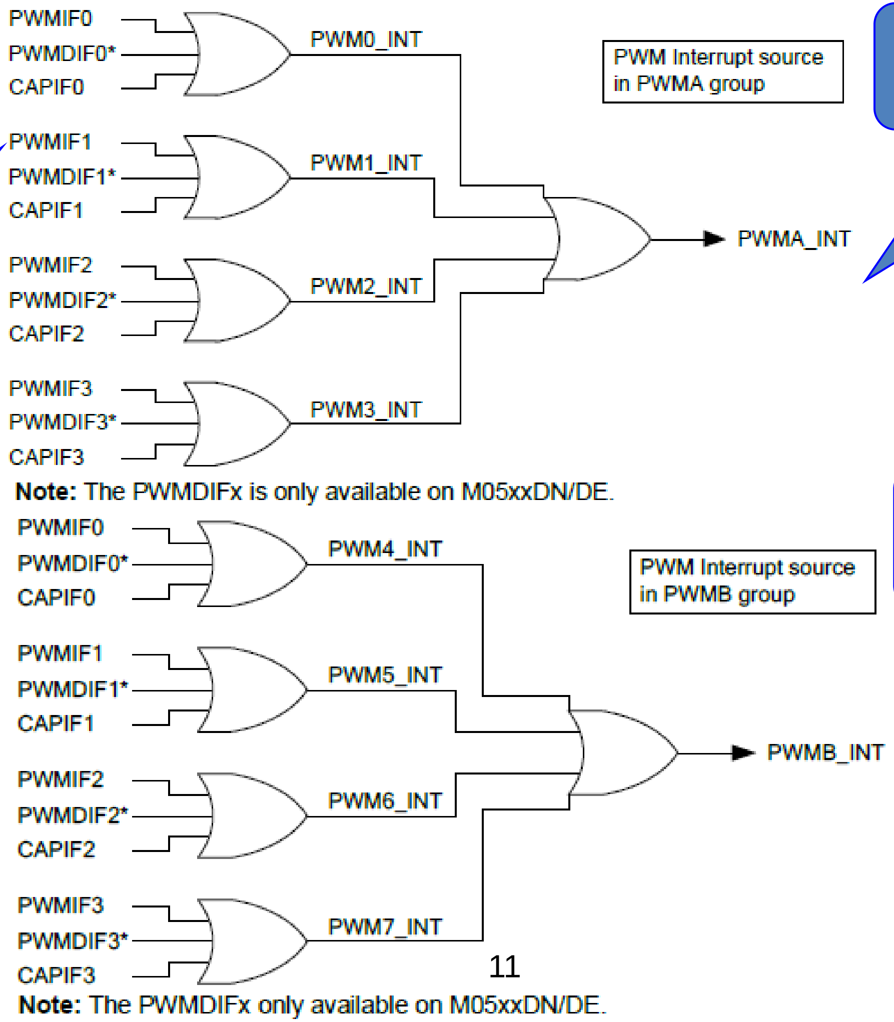
The PWM counter will be reloaded with CNRx when a capture interrupt flag (CAPIFx) is set

The channel low pulse width is (CNR+1 - CRLR).

The channel high pulse width is (CNR+1 - CFLR).

Interrupt Architecture

- **Two PWM/Capture Interrupt Vectors**



Note: The PWMDIFx is only available on M05xxDN/DE.

Note: The PWMDIFx only available on M05xxDN/DE.

Each PWM/ Capture channel has its own interrupt flag

Channel0~Channel3 share one interrupt vector

Channel4~Channel7 share one interrupt vector

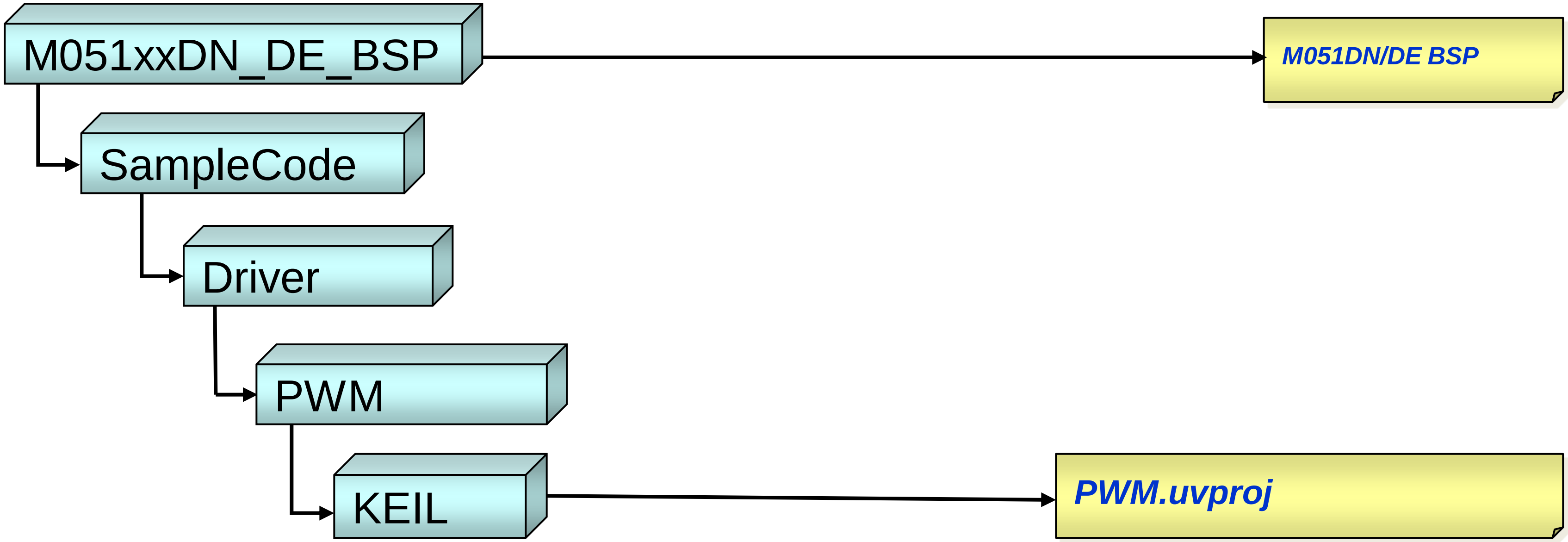
PWM Sample Code (1/2)

```
/*----------------------------------------------------------------------------------------*/
/*  Main Function                                                                         */
/*----------------------------------------------------------------------------------------*/
int32_t main (void)
{
    /* Enable IP clock */
    SYSCLK->APBCLK |= SYSCLK_APBCLK_PWM01_EN_Msk;
    /* Set P2 multi-function pins for PWMB Channel0~3  */
    SYS->P2_MFP = SYS_MFP_P21_PWM1;

    /*Set Pwm mode*/
    _PWM_SET_TIMER_AUTO_RELOAD_MODE(PWMA,PWM_CH1);
    /*Set PWM Timer clock prescaler*/
    _PWM_SET_TIMER_PRESCALE(PWMA,PWM_CH1, 1); // Divided by 2

    /*Set PWM Timer clock divider select*/
    _PWM_SET_TIMER_CLOCK_DIV(PWMA,PWM_CH1,PWM_CSR_DIV1);

    /*Set PWM Timer duty*/
    PWMA->CMR1 = 0x1FF;

    /*Set PWM Timer period*/
    PWMA->CNR1 = 0x3FF;

    /* Enable PWM Output pin */
    _PWM_ENABLE_PWM_OUT(PWMA, PWM_CH1);

    /* Disable PWMB NVIC */
    NVIC_DisableIRQ((IRQn_Type)(PWMB_IRQn));

    /* Enable PWM Timer */
    _PWM_ENABLE_TIMER(PWMA, PWM_CH1);
}
```

PWM Sample Code (2/2)

```
/*---------------------------------------------------------------------------------------------*/
/* PWM Timer function                                                                      */
/*---------------------------------------------------------------------------------------------*/
void PWMB_IRQHandler(void)
{
      PWMA->PIIR = PWM_PIIR_PWMIF1_Msk;
      PWMA->CMR1++;
     if(PWMA->CMR1 > PWMA->CNR1)
     {
            PWMA->CMR1;
     }
}
```

Run PWM Sample Code


M051xxDN_DE_BSP
SampleCode
Driver
PWM
KEIL
M051DN/DE BSP
PWM.uvproj