NuMicro™ NUC100 Series Version A to D Migration Guide

Application Note for 32-bit NuMicro® Family

Rev. 1.00 — Dec. 19, 2012

Document Information

Abstract	This document includes all necessary information to migrate the code of NUC100 series from version A to version D, including a different list between version A and D, performance improvement and migration notice. Users could reference the difference list to check whether they need to modify the code for version D.
Apply to	NUC100 Series

Table of Contents

1 INTRODUCTION 4

2 MIGRATING VERSION A TO D 5

2.1 Version A and D Differences List 5

2.2 Software Migration 10

2.2.1 ADC – DMOF Setting 10

2.2.2 ADC – ADPDMA 10

2.2.3 ADC – STADC 11

2.2.4 ADC – ADST 11

2.2.5 ADC – Calibration 12

2.2.6 ADC – Conversion Rate 12

2.2.7 CLKCTL – RC22M 13

2.2.8 CLKCTL – Clock Switching 13

2.2.9 CLKCTL – PWR_DOWN_EN 14

2.2.10 CLKCTL – HCLK_S 14

2.2.11 CLKCTL – CLKSEL2 15

2.2.12 FMC – FATCON 15

2.2.13 GCR – BOD_LPM 16

2.2.14 GCR – BOD Reset 16

2.2.15 GPIO – Maximum I/O Toggle Rate 17

2.2.16 NMI – NMI Interrupt 17

2.2.17 PWM – PWMIF 17

2.2.18 RTC – TWKE 18

2.2.19 SPI – FIFO Mode 18

2.2.20 SPI – Suspend Interval 19

2.2.21 SPI – Byte Reorder Function 19

2.2.22 SPI – Clock Configuration 20

2.2.23 SPI – Burst Transfer 21

2.2.24 SPI – IF Bit 21

2.2.25 SPI – Tx Register 22

2.2.26 SRAM – Access Timing 22

2.2.27 TIMER – Counter Function 23

2.2.28 TIMER – TDR 23

2.2.29 TIMER – Change Operation Mode 24

2.2.30 UART – BIF/FEF/PEF Flags 24

2.2.31 UART – FIFO Depth 24

2.2.32 WDT – Reset Delay 25

REVISION HISTORY 26

1 Introduction

The NuMicro™ NUC100 series has been updated to version D with better performance and more features. Most of the core system and peripherals in version D are fully compatible with version A. However, in some special cases, the hardware has slight differences that may impact the program behavior designed for version A. In this document, we will highlight all potential impacts for users to migrate the code form version A to version D more easily.

2 Migrating Version A to D

To make it easier to check whether it is necessary to modify the code and show how to migrate the code for version D, the following sections will list all differences between version A and D and describe how to modify the code for version D from version A.

2.1 Version A and D Differences List

The following list shows the compatibility of version A and version D. For not compatible functions, the differences between version A and version D are listed. The improvements between version A and version D are also listed in the list.

Function	Compatible	Differences		Improvement
		Version A	Version D	
ADC	**No (2.2.1)**	N/A	Add a control bit, DMOF, in ADCR[31].	To support 2's complement format in differential mode by setting ADCR[31].
	No (2.2.2)	The count of available bit of ADPDMA register is 12 bits.	The count of available bit of ADPDMA register is 18 bits, including VALID flag, OVERRUN flag and sign-extension bits.	
	No (2.2.3)	When the high level external trigger type is selected, the STADC pin de-bouncing function does not work.	The STADC pin de-bouncing function works on all external trigger types.	
	No (2.2.4)	When software clears ADST bit to 0 in Continuous Scan mode, the A/D conversion will be stopped and will store the conversion result to channel 0 data register after the current conversion is finished.	When software clears the ADST bit to 0 in Continuous Scan mode, the A/D conversion will be stopped immediately. The ADF, VALID, OVERRUN and conversion result will not be updated.	
	No (2.2.5)	Support calibration function.	Not support calibration function.	
	No (2.2.6)	Conversion rate is about 600 kSPS.	Conversion rate is about 700 kSPS.	Improve the conversion rate in version D.
	Yes	After ADC finishes the conversion and clears the ADST bit, software must wait one ADC clock cycle before setting the ADST bit to 1 again.	Software can set ADST bit to 1 immediately after ADC finishes the conversion and clears the ADST bit to 0.	
	Yes	N/A	Add HCLK as a ADC clock source in	HCLK can be selected as

			CLKSEL1[3:2].	ADC clock source.
	Yes	N/A	Add a new hardware trigger source in ADCR[5:4].	The A/D conversion can be started by PWM center-aligned trigger.
CLKCTL	**No (2.2.7)**	Before executing ISP, user should enable RC22M in PWRCON[2]	RC22M clock is enabled automatically when ISPEN (ISPCON[0]) is set to 1.	
	No (2.2.8)	1. Need to delay about 5ms to wait for clock source ready after enabling some clock sources. Users must make sure the clock source is ready before they can use it. 2. Clock switching always take effect after setting it.	1. Add CLKSTATUS register to monitor the clock source status. User need to make sure the clock source status is ready before using it. 2. Clock switching will fail if relative clock source is not ready.	
	No (2.2.9)	The power-down enable bit (PWR_DOWN_EN) is cleared by hardware if any GPIO interrupt occurred before WFI executed.	The power down enable bit (PWR_DOWN_EN) will not be cleared by hardware even if any GPIO interrupt occurred before WFI executed.	
	No (2.2.10)	Software can set HCLK_S=0x1xx to switch HCLK clock source to RC22M.	Software can only set HCLK_S=0x111 to switch HCLK clock source to RC22M, HCLK_S=100/101/110 was saw as invalid clock source.	N/A
	No (2.2.11)	I2S & clock divider default value is XTL12M	I2S & clock divider default value is RC22M	N/A
	Yes	N/A	Add EBI clock control bit in EBI_EN(AHBCLK[3])	Add EBI function.
COMP	Yes	N/A	N/A	N/A
CRC	Yes	N/A	Support CRC function	Add CRC function to version D.
EBI	Yes	N/A	Support EBI function	Add EBI function to version D.
FMC	**No (2.2.12)**	N/A	No longer support FATS and FPSEN of FATCON control register.	
	Yes	N/A	Add APUEN to ISPCON[3] to support APROM update in APROM.	To support AP update AP function.
GCR	**No (2.2.13)**	BOD_LPM will be reset only by pin reset & POR.	BOD_LPM will be reset by any reset source	N/A
	No (2.2.14)	When system voltage is lower than BOD setting voltage and BOD reset function is enabled, system will assert a signal to reset chip. System will not be held in reset state even if voltage is still lower than BOD setting voltage.	When system voltage is lower than BOD setting voltage and BOD reset function is enabled, system will assert a signal to reset chip. The system will be held in reset state until system voltage is higher than BOD setting voltage.	N/A
GPIO	**No (2.2.15)**	N/A	N/A	Improve the maximum I/O toggle speed from 7 HCLK to 4 HCLK.
	Yes	N/A	N/A	Add GPIO bit control registers.

	Yes	User needs to disable de-bounce function of GPIO before power down to avoid double GPIO interrupts when system wake-up by GPIO.	No need to care about whether the de-bounce function is enabled or not when power down.	No need to disable the de-bounce function when power down.
HIRC	Yes	N/A	Offer a HIRC accurate frequency control of 22.1184 MHz or 24 MHz.	Support the HIRC auto trim function.
I2C	Yes	N/A	Support power down wake-up function.	Version D adds the wake-up function
I2S	Yes	N/A	Add a control bit, RXLCH, in I2SCON[23].	When monaural format is selected, I2S controller can receive data from either right channel or left channel.
NMI	**No (2.2.16)**	NMI is always enabled and assigned to IRQ0 (BOD Interrupt) by default.	NMI can be disabled and default to be disabled.	Add a protect bit of NMI_EN in NMI_SEL register to enable or disable the NMI interrupt.
PDMA	Yes	N/A	N/A	N/A
PS2	Yes	PS2 I/O pins (PS2DAT, PS2CLK) are dedicated for PS2.	PS2 I/O pins (PS2DAT, PS2CLK) can be configured as GPIO or PS2 I/O pins. In default, they are configured as PS2 I/O pins to compatible with version A.	Version D supports multi-function pin option for PS2.
PWM	**No (2.2.17)**	PWMIFx (x = 0~3) will not be set if PWMIEx is disabled.	No matter PWMIEx is enabled or not, PWMIFx(x=0~3) is set by hardware when PWMx down counter reaches zero	
	Yes	N/A	CCR[6]/[7]/[22]/[23] can be option to write 1/0 clear by setting PBCR register. Default to write 0 clear	Add write one clear option to clear rising/falling indicator.
	Yes	N/A	Setting PCR[1]/[9]/[17]/[25](CHxPINV) can inverse PWM output pin waveform.	Add PWM polar inverse function.
	Yes	N/A	Setting PCR[30]/[31](PWMxxTYPE) can select PWM timer to center-aligned type or edge-aligned type	Add Center-aligned type function.
	Yes	N/A	Set PIER[8]/[9]/[10]/[11](PWMDIEx) enable duty interrupt	Add PWM timer duty interrupt enable function.
	Yes	N/A	Setting PIER[16]/[17](INTxxTYPE) can select two kinds of PWM interrupt period type.	Add PWM interrupt period type selection function.
	Yes	N/A	PIIR[8]/[9]/[10]/[11](PWMDIFx) can indicate duty interrupt	Add PWM timer duty interrupt flag.
	Yes	N/A	Setting TCON[0]/[1]/[2]/[3](PWMxTEN) can enable PWM trigger ADC function at center-aligned type.	Add PWM trigger ADC function.
	Yes	N/A	TSTATUS[0]/[1]/[2]/[3](PWMxTF) can indicate PWM trigger ADC at center-aligned type.	Add PWM trigger ADC flag.
	Yes	N/A	SYNCBUSYx can indicate hardware	Add Synchronous busy bit

			synchronous status when software writes CNRx/CMRx/PPR or switches PWMx operation mode.	function.
	Yes	N/A	PWM clock source can be selected from internal 10KHz by setting CLKSEL2[8]/[9]/[10]/[11]/[4]/[5]/[6]/[7] and CLKSEL1[28]/[29]/[30]/[31]	PWM source can be from internal 10 kHz.
RTC	**No (2.2.18)**	Set TTR[3](TWKE) to enable/disable tick interrupt wake-up function, but the alarm interrupt wake-up function is forced on.	Remove TTR[3](TWKE); the tick/alarm interrupt wake-up function is controlled by setting RIER[1](TIER)/RIER[0](AIER).	Alarm interrupt wake-up function can be disabled.
SC	Yes	N/A	Support the Smart Card function.	Add Smart card function to version D.
SPI	**No (2.2.19)**	No FIFO mode	Support FIFO mode.	Add 8-layer depth FIFO. The SPI controller can send and receive consecutive serial clock transactions.
	No (2.2.20)	N/A	The calculation formula of suspend interval is different from version A.	
	No (2.2.21)	Two control bits configuration in byte reorder function	One control bit configuration in byte reorder function	
	No (2.2.22)	Both clock dividers of variable clock function are 16-bit dividers and the second clock source of variable clock function is system clock.	Both clock dividers of variable clock function are 8-bit dividers and the second clock source of variable clock function is SPI engine clock.	
	No (2.2.23)	Support Burst Transfer mode.	Not support Burst Transfer mode.	FIFO mode can support better performance and flexibility than Burst mode.
	No (2.2.24)	The IF bit will be set to 1 if slave is configured as level trigger and slave select pin goes to inactive state.	Add a slave select inactive interrupt option, SS_INT_OPT, in SPI_CNTRL2[16]. This option decides that if slave is configured as level trigger and slave select pin goes to inactive state, whether the IF bit will be set or not. In default, the IF bit will not be set to 1.	
	No (2.2.25)	Slave can update Tx register no matter the GO_BUSY bit is 0 or 1.	Slave can update Tx register only when the GO_BUSY bit is 0.	
	Yes	N/A	Add a SPI engine clock backward compatible option, BCn, in SPI_CNTRL2[31] and it is default to be backward compatible.	The SPI clock rate of Master mode can be up to 32 MHz and Slave mode can be up to 20 MHz at 5V.
	Yes	N/A	Add two Dual I/O Transfer mode control bits, DUAL_IO_EN and DUAL_IO_DIR, in SPI_CNTRL2[13:12].	Support Dual I/O Transfer mode.
	Yes	N/A	Slave supports 3-wire mode.	Add slave 3-wire mode.
	Yes	Not support multi-slave application. The SPI clock pin must be kept at idle state when the slave select pin is at inactive state.	Support multi-slave application.	

SRAM	**No (2.2.26)**	N/A	N/A	SRAM access is zero wait state in version D.
	Yes	Response error only if access range is larger than 16K, no matter chip SRAM size is 16K/8K/4K	Response error if access address is out of chip SRAM size. For example, if the SRAM size is 8KB, system will be "HardFault" when accessing size is larger than 8KB.	
TIMER	**No (2.2.27)**	Use external clock source to achieve counter function	Add counter mode register to count event from pin correctly. Remove external clock from timer clock sources.	
	No (2.2.28)	TDR cannot be read sequentially (only read even value). TDR value equal to real count value minus 3.	TDR can read value sequentially (if software reads speed fast enough). New counter mode can read the exact count value from TDR.	
	No (2.2.29)	When software switches TIMER operation mode before TIMER start running, it may cause TIMER operating error	Software can switch TIMER operation mode before TIMER starts running.	
	Yes	N/A	N/A	Add toggle mode output function.
	Yes	N/A	TIMER clock source can be selected from internal 10KHz by setting CLKSEL1[10:8]/CLKSEL1[14:12]/ CLKSEL1[18:16]/CLKSEL1[22:20] to 0x001 or 0x101	Add internal 10 kHz clock source selection for TIMER wake-up function.
	Yes	N/A	Add TCSR[23](WAKE_EN) to enable/disable TIMER wake-up function.	Add the TIMER wake-up function.
	Yes	N/A	Add TCAP, TEXCON and TISR registers to support TIMER input capture function	Add the TIMER input capture function.
UART	**No (2.2.30)**	When break or error interrupt occurs, one of BIF, FEF, PEF bits will set to 1. But software cannot write 1 to clear these bits. User needs to flush FIFO to clear the error flag.	Support to write 1 to clear BIF, FEF, and PEF bits.	
	No (2.2.31)	UART FIFO built-in 63/15/15 (UART0/UART1/UART2) bytes.	UART FIFO built-in 64/16/16 (UART0/UART1/UART2) bytes.	Version D adds two bits of TX_FULL and RX_FULL to indicate the FIFO status.
USB	Yes	N/A	N/A	N/A
WDT	**No (2.2.32)**	WDT time-out reset delay period is fixed as 1024*WDT_CLK	WDT time-out reset delay period is configurable by setting WTCRALT[1:0] and default to be (1024 + 2) * WDT_CLK	Add new register WTCRALT to select WDT time-out reset delay period time.
WWDT	Yes	N/A	Support the Window Watch Dog Timer function.	Add the WWDT function in version D.

Table 2.1 NUC100 Version A to D difference List

2.2 Software Migration

This section provides the migration information and necessary notification about the no compatible functions, which are listed in Table 2.1.

2.2.1 ADC – DMOF Setting

Description:

In version A, ADC does not support 2's complement output format. In version D, the ADC output format can be configured by software as 2's complement by setting the DMOF bit to 1 when ADC analog input type is configured as differential input mode. If the differential input mode is disabled, the DMOF bit should be cleared to 0.

Affected Registers:

ADCR[31]

Migration Note:

This is a new feature in version D.

2.2.2 ADC – ADPDMA

Description:

The ADPDMA register stores the current PDMA transfer data. In version A, the ADPDMA register only contains the 12-bit conversion result. In version D, it includes VALID flag, OVERRUN flag and 16-bit sign-extension conversion result.

Affected Registers:

ADPDMA register

Migration Note:

In version A, the conversion result is stored in ADPDMA[11:0]. In version D, the VALID flag is placed in ADPDMA[17], the OVERRUN flag is placed in ADPDMA[16], the 16-bit sign-extension conversion result is placed in ADPDMA[15:0]. If user's application does not need to read the

VALID flag and OVERRUN flag, configure the PDMA transfer width as 16 bits.

2.2.3 ADC – STADC

Description:

When the STADC pin is selected as the ADC trigger source, the pin de-bouncing mechanism can protect from the bounce on STADC pin. If level trigger condition is selected, the STADC pin must be kept at defined state at least 8 PCLK cycles to start an A/D conversion. If edge trigger condition is selected, the high and low state must be kept at least 4 PCLK cycles. Pulses shorter than this specification will be ignored. But in version A, when high level trigger type is selected, the de-bouncing function does not work.

Affected Registers:

ADCR[8] / ADCR[7:6] / ADCR[5:4]

Migration Note:

In version D, if the STADC pin high level trigger is selected as the ADC trigger source, the STADC pin must be kept at high state at least 8 PCLK cycles to start a conversion. In version A, the de-bouncing function does not work. The A/D conversion will start as the STADC pin is at high state.

2.2.4 ADC – ADST

Description:

In version A, when the ADST bit is cleared to 0 by software in Continuous Scan mode, the A/D conversion will be stopped and will store the conversion result to channel 0 data register after the current conversion is finished. In version D, when the ADST bit is cleared to 0 by software in Continuous Scan mode, the A/D conversion will be stopped immediately. The ADF, VALID, OVERRUN and conversion results will not be updated.

Affected Registers:

ADCR[11] / ADCR[3:2]

Migration Note:

In version A, the contents of data register 0 is unpredictable after clearing the ADST bit in Continuous Scan mode. There is no such limitation in version D.

2.2.5 ADC – Calibration

Description:

Version A is provided with the calibration function to minimize the conversion error, but version D does not support the calibration function.

Affected Registers:

ADCALR register

Migration Note:

Version D has no ADCALR register.

2.2.6 ADC – Conversion Rate

Description:

In version A, ADC needs about 27 ADC clock cycles to complete an A/D conversion. In version D, ADC only needs about 21 ADC clock cycles. Therefore, if the ADC clock rate is 16 MHz in version D, the conversion rate is about 700 kSPS (Samples per Second).

Affected Registers:

None

Migration Note:

In most cases, there is no need to make any modification for this change.

2.2.7 CLKCTL – RC22M

Description:

It is necessary to enable RC22M before using ISP function of NUC100. In NUC100 version A, users need to enable it by themselves before using the ISP function. In version D, the RC22M will be enabled automatically when starting to use ISP function, i.e. setting ISPEN of ISPCON control register to 1 will cause RC22M to be enabled.

Affected Registers:

ISPCON[0]

PWRCON[2]

Migration Note:

In general, the program designed in version A does not require any change for this difference.

2.2.8 CLKCTL – Clock Switching

Description:

There are different clock sources that can be used in the NuMicro™ NUC100 Family. In version A, the switching control is always taking effect after setting the relative control register. However, in version D, the hardware will monitor relative clock source status. If the clock source is not ready, the switching will fail. Therefore, user must make sure the clock source status is ready for use before switching the clock source.

Affected Registers:

CLKSTATUS

Migration Note:

User can use delay to wait for clock source ready in both version A and version D. However, if the delay time is not enough for clock ready, the switching setting may work in version A but fail in version D.

There are two options to migrate the code from version A to version D to avoid clock switching fail, as described below.

1. Use the same way as version A, i.e. using delay loop to wait clock source ready.

By using delay, the code would be the same in both version A and D. However, user must make sure the delay time is long enough for clock source ready.

2. Use the CLKSTATUS register to check clock source ready status.

This method is only supported in version D. In version D, if clock switching is performed before clock source ready, the switching will fail and the clock setting is kept in original state.

2.2.9 CLKCTL – PWR_DOWN_EN

Description:

In version A, the power down enable bit (PWR_DOWN_EN) is cleared by hardware if any GPIO interrupt or specific wake-up source interrupt occurred before WFI executed. the power-down enable bit needs to be enabled by software again in GPIO interrupt handler and specific wake-up interrupt handler to make sure power down enable bit is enabled. In version D, power down enable bit will not be cleared even if GPIO interrupt or specific wake-up interrupt occurred before WFI executed. The power-down enable bit does not need to be enabled in GPIO interrupt handler.

Affected Registers:

PWRCON[7]

Migration Note:

In general, the program designed at version A does not require any change for this difference.

2.2.10 CLKCTL – HCLK_S

Description:

In version A, users can set HCLK_S of CLKSEL0 control register as **0x1xx** to configure the system clock source as 22.1184 MHz. However, in version D, only **0x111** can be used in HCLK_S to configure the system clock source as 22.1184 MHz.

Affected Registers:

CLKSEL0[2:0]

Migration Note:

Users must make sure their HCLK_S setting is **0x111** when they want to use internal 22.1184 MHz clock as system clock source. **0x100**, **0x101** and **0x110** are all reserved for further usage in version D.

2.2.11 CLKCTL – CLKSEL2

Description:

The reset default of CLKSEL2 is changed from **0x000000F0** to **0x000200FF** in version D, which means the default clock source of I2S (CLKSEL2[1:0]) and FRQDIV(CLKSEL2[3:2]) are changed to 22.1184 MHz and the WWDT default clock source set to HCLK/2048.

Affected Registers:

CLKSEL2[3:0]

Migration Note:

Users must add the code to set the clock source of I2S and FRQDIV in version D to make sure the clock source is what they want.

2.2.12 FMC – FATCON

Description:

In version D, FATS and FPSEN of FATCON control register are no longer supported. The settings for FATCON[3:1] and FATCON[0] are not effective.

Affected Registers:

ISPCON[3]

Migration Note:

The settings for FATS and FPSEN will be invalid in version D.

2.2.13 GCR – BOD_LPM

Description:

The BOD_LPM bit of BODCR control register is changed to be reset only when pin reset or power on reset.

Affected Registers:

BODCR[5]

Migration Note:

It is hardware behavior of reset and no modification is required for software with regard to it.

2.2.14 GCR – BOD Reset

Description:

In version A, when system voltage is lower than BOD setting voltage and the BOD reset function is enabled, system will assert a signal to reset chip. The system will not be held in reset state even if voltage is still lower than BOD setting. This will cause chip reset and run continuously when system voltage is lower than BOD setting voltage.
In version D, when system voltage is lower than BOD setting voltage and the BOD reset function is enabled, system will assert a signal to reset chip. The system will be held in reset state until system voltage is higher than BOD setting voltage.

Affected Registers:

N/A

Migration Note:

In most cases, it is hardware behavior of reset and no modification is required for software with regard to this change.

2.2.15 GPIO – Maximum I/O Toggle Rate

Description:

In version A, the maximum I/O toggle rate is 7 HCLK.

In version D, the maximum I/O toggle rate is improved to 4 HCLK.

Affected Registers:

N/A

Migration Note:

I/O toggle rate is improved to 4 HCLK in version D.

2.2.16 NMI – NMI Interrupt

Description:

In version D, a NMI_EN bit is added in the NMI_SEL register to enable or disable NMI interrupt. This bit is a protect bit. Programming this bit needs to write “59h”, “16h”, 88h” to the REGWRPROT register (0x5000_0100) to unlock bit protection.

Affected Registers:

NMI_SEL[8]

REGWRPROT

Migration Note:

This is a new feature in version D. There is no need to modify the code for version D if NMI is not used.

2.2.17 PWM – PWMIF

Description:

In version A, PWMIFx(x=0~3) will not be set if the specified PWMIEx is disabled. In version D, no matter the specified PWMIEx is enabled or not, PWMIFx(x=0~3) is set by hardware when PWMx down counter reaches zero.

Affected Registers:

PIIR[3:0]

Migration Note:

In general, the program designed at version A doesn't need to do any change for this difference.

2.2.18 RTC – TWKE

Description:

In version A, RTC alarm interrupt wake-up function is always enabled and the tick interrupt wake-up function can be enabled/disabled by setting TWKE bit to 1 or 0.

In version D, the TWKE bit has been removed and RTC alarm and tick interrupt wake-up function are corresponding to the RTC AIER bit and RTC TIER bit settings. If this bit is set to 1, the corresponding interrupt and wake-up function are all enabled; otherwise, the interrupt and wake-up function are all disabled.

Affected Registers:

TTR[3], RIER[1:0]

Migration Note:

The RTC alarm interrupt wake-up function can be disabled in version D, but it is always enabled in version A.

2.2.19 SPI – FIFO Mode

Description:

In version A, if Burst Transfer mode is enabled by software, SPI controller will transfer 2 data stored in SPI_TX0 and SPI_TX1 registers when the GO_BUSY bit is set to 1. In version D, SPI controller can increase the efficiency with the 8-layer depth FIFO buffer. If FIFO mode is enabled, once the transmit FIFO buffer is not empty, SPI controller will set the GO_BUSY bit automatically. Software can write the next data to transmit FIFO buffer whenever the FIFO buffer is not full. In the meantime, the receive data will be stored in receive FIFO buffer. The

receive data can be read by software whenever the receive FIFO buffer is not empty.

Affected Registers:
SPI_CNTRL[21]

Migration Note:
This is a new feature in version D.

2.2.20 SPI – Suspend Interval

Description:
The definition of the suspend interval is the interval between the last clock edge of the preceding transaction word and the first clock edge of the following transaction word. In version A, the default value is 0 (but 3 in version D). The calculation of suspend interval is different in both versions.

Affected Registers:
SPI_CNTRL[15:12]

Migration Note:
The period of the suspend interval is obtained according to the following equation.
Version A:
(SPI_CNTRL[15:12] + 2) period of SPI clock cycle
Version D:
(SPI_CNTRL[15:12] + 0.5) period of SPI clock cycle

2.2.21 SPI – Byte Reorder Function

Description:
In version A, SPI controller uses SPI_CNTRL[20:19] to configure the byte reorder function and byte suspend function. In version D, only SPI_CNTRL[19] is used to control the byte reorder function. If byte reorder function is disabled, the byte suspend function will be disabled as well.

Affected Registers:

SPI_CNTRL[20:19]

Migration Note:

In version A, the configuration of byte reorder function and byte suspend function is as follows.

SPI_CNTRL[20:19] = 2'b00

Both byte reorder and byte suspend functions are disabled.

SPI_CNTRL[20:19] = 2'b01

Byte reorder function is enabled, and a byte suspend interval will be inserted among each byte. The setting of TX_BIT_LEN must be configured as 0.

SPI_CNTRL[20:19] = 2'b10

Byte reorder function is enabled, but byte suspend function is disabled.

SPI_CNTRL[20:19] = 2'b11

Byte reorder function is disabled, but a byte suspend interval will be inserted among each byte. The setting of TX_BIT_LEN must be configured as 0.

In version D, the configuration is as follows.

SPI_CNTRL[19] = 1

Byte reorder function is enabled. A byte suspend interval will be inserted among each byte. The period of the byte suspend interval depends on the setting of SP_CYCLE (SPI_CNTRL[15:12]).

SPI_CNTRL[19] = 0

Both byte reorder and byte suspend functions are disabled.

2.2.22 SPI – Clock Configuration

Description:

In version A, SPI engine clock source is system clock. The SPI engine clock divider is 16-bit width. In version D, the clock source can be PLL or system clock. By default, system clock is selected as the SPI engine clock source. The engine clock divider is 8-bit width. The clock calculation of both versions is different.

Affected Registers:

CLKSEL1[7:4] / SPI_DIVIDER / SPI_CNTRL2[31]

Migration Note:

In version A, the SPI engine clock calculation of master mode is as follows.

SPI engine clock rate = system clock rate / (SPI_DIVIDER[15:0]+1)*2

In slave mode, the SPI engine clock is the system clock. The maximum available frequency of SPI clock is the fifth of the slave's system clock rate.

In version D, both the master's and slave's engine clock are configurable. The slave's engine clock rate must be larger than the input SPI clock rate. The engine clock calculation is as follows.

If SPI_CNTRL2[31] is cleared to 0,

SPI engine clock rate = system clock rate / ((SPI_DIVIDER[7:0]+1)*2)

If SPI_CNTRL2[31] is set to 1,

SPI engine clock rate = system clock rate or PLL clock rate / (SPI_DIVIDER[7:0]+1)

2.2.23 SPI – Burst Transfer

Description:

In version A, if Burst Transfer mode is enabled through software, SPI controller will transfer 2 data stored in SPI_TX0 and SPI_TX1 registers when GO_BUSY bit is set to 1. Version D does not support burst transfer mode. In version D, SPI controller can get better efficiency with the 8-layer depth FIFO buffer.

Affected Registers:

SPI_CNTRL[9:8]

Migration Note:

In version D, SPI_CNTRL[9:8] are reserved bits.

2.2.24 SPI – IF Bit

Description:

SPI_CNTRL[16], IF bit, will be set to 1 when a SPI transaction is done.

Affected Registers:

SPI_CNTRL[16] / SPI_CNTRL2[16]

Migration Note:

In version A, the IF bit will be set to 1 if slave is configured as level trigger and slave select pin goes to inactive state.

In version D, a slave select inactive interrupt option, SS_INT_OPT, is added in SPI_CNTRL2[16]. This option decides whether the IF bit will be set or not when slave is configured as level trigger and the slave select pin goes to inactive state. By default, the IF bit will not be set to 1.

2.2.25 SPI – Tx Register

Description:

The SPI_TX registers are used to store the transmit data. In version A, slave can update Tx register no matter the GO_BUSY bit is 0 or 1. In version D, slave can update the Tx register only when the GO_BUSY bit is 0.

Affected Registers:

SPI_CNTRL[0] / SPI_TX0 register / SPI_TX1 register

Migration Note:

Before updating the TX register, make sure the GO_BUSY bit is 0.

2.2.26 SRAM – Access Timing

Description:

In version A, the SRAM access timing is one wait state. In version D, the SRAM access timing is improved to be zero wait state.

Affected Registers:

None

Migration Note:
This is a hardware improvement and no modification is required for software. However, the performance would be better in version D.

2.2.27 TIMER – Counter Function

Description:
In version A, external counter pins TM0~3 are usually as external clock input source for timer clock source. In version D, a CTB bit (TCSR[24] Counter Mode Enable Bit) is added. When the CTB bit is set to 1, the TMx pin is used as an event counter pin to count the precise external input events and the timer clock source must be selected as HCLK.

Affected Registers:
TCSR[24]

Migration Note:
In version D, a CTB bit is added to enable the external event counting function.

2.2.28 TIMER – TDR

Description:
In version A, software can only read 0, 2, 4 … from TDR control register. In version D, software can read the exact current timer count value from TDR, e.g. 0,1,2…

Affected Registers:
TDR[23:0] and CLKSEL2[17:16]

Migration Note:
In most cases, there is no need to make any modification for this change.

2.2.29 TIMER – Change Operation Mode

Description:

In version A, CEN (Timer Enable Bit) needs to be enabled by software before changing operation mode to One-Shot, Periodic, Toggle or Continuous-Counting. In version D, this limitation has been removed. Software can change mode before or after enabling CEN.

Affected Registers:

TCSR[28:27]

Migration Note:

In most cases, there is no need to make any modification for this change.

2.2.30 UART – BIF/FEF/PEF Flags

Description:

In version A, when error or break interrupts occur, one of BIF, FEF, PEF flags will be set to 1. User needs to clear the flags by reset FIFO. However, in version D, user needs to write '1' to the specified flag to clear it. It means we need to write 1 to BIR, FEF or PEF to clear it.

Affected Registers:

UA_FSR[6:4]

Migration Note:

In version D, users need to write 1 to clear BIF, FEF, PEF.

2.2.31 UART – FIFO Depth

Description:

In version A, the FIFO depth is 63/15/15(UART0/UART1/UART2) bytes, which means user should not fill FIFO over 63/15/15(UART0/UART1/UART2) bytes. In version D, the FIFO depth is 64/16/16(UART0/UART1/UART2) bytes. Besides, version D offers two bits of TX_FULL and RX_FULL in UA_FSR to indicate the FIFO full status.

Affected Registers:

UA_FSR[15]

UA_FSR[23]

Migration Note:

This is a new feature in version D.

2.2.32 WDT – Reset Delay

Description:

In version A, WDT time-out reset delay period is fixed as 1024*WDT_CLK.

In version D, WDT time-out reset delay period can be selected to (1024+2)*WDT_CLK, (128+2)*WDT_CLK, (16+2)*WDT_CLK or (1+2)*WDT_CLK by setting WTCRALT[1:0].

Affected Registers:

WTCRALT[1:0]

Migration Note:

This is a new feature in version D.

Revision History

Rev.	Date	Description
1.00	12-19-2012	Initially issued.

Important Notice

Nuvoton products are not designed, intended, authorized or warranted for use as components in systems or equipment intended for surgical implantation, atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, or for other applications intended to support or sustain life. Furthermore, Nuvoton products are not intended for applications wherein failure of Nuvoton products could result or lead to a situation wherein personal injury, death or severe property or environmental damage could occur.

Nuvoton customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nuvoton for any damages resulting from such improper use or sales.

Please note that all data and specifications are subject to change without notice. All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.