



ARM Cortex[®]-M0 32-bit Microcontroller

NuVoice[™] Family N572F065 Multi-Algorithm Voice Processor Technical Reference Manual

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuVoice microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com



1 GENERAL DESCRIPTION

The N572F065 is the first Cortex™-M0 based processor for voice process applications, running up to 48MHz and equipped with 64KB flash and 8KB SRAM for high performance process of audio and voice algorithms. Integrating rich analog peripherals, like pre-amplifier, ADC, DAC, hardware mixer, and PA, this chip saves a lot of system design effort and cost. An USB device and flash work together to provide updatable content and scenario downloaded from internet via a PC.

To release the high performance M0 and high density of SRAM, advanced algorithms are designed, optimized, and tested in N572F065 chip. These algorithms include watermark, voice changer, sub-band, sound direction detection, beat detection, and more in developing. In addition to algorithms developed by Nuvoton, we also seek third parties for more interesting software to enrich the applications on N572F065. With the new designed NVIC in M0, the latency of interrupt and response to external events are quick. Multiple algorithms could be run together smoothly and naturally.

The development tools are based on Keil™ MDK using C/C++ for programming language. This is a robust and easy to use environment for development and debug.

Note1: LDO30E naming is changed to Voutx.

Note2: PA-bypass DAC output function is not supported.

Note3: Please reference data sheet for electrical related spec and typical applications.

Following is a brief table of all Part No. of N572 Series:

Part No.	N572F072	N572P072	N572F065
Program ROM	72KB Flash	72KB OTP or 64KB OTP + 8KB Flash	64KB Flash
SRAM	8KB		
CPU freq	48MHz		
SPI Interface	Master/Slave mode 12MHz, 1 set Master mode 24MHz, 1 set		Master mode 12MHz, 2 sets
USB	N/A	N/A	FS/12Mbps



2 FEATURES

- **Core**
 - ARM® Cortex™-M0 core runs up to 48MHz.
 - Support low power sleep mode
 - Single-cycle 32-bit hardware multiplier.
 - NVIC for the 16 interrupt inputs, each with 4-levels of priority.
 - Serial Wire Debug supports with 2 watchpoints/4 breakpoints.
- **Widely operating voltage range from 2.4V to 5.5V**
- **Flash EPROM Memory**
 - 64K bytes Flash EPROM
 - Support ISP for Flash update
 - 512 bytes page erase for Flash
 - Support 2-wire ICP update from ICE interface
 - Support fast parallel programming mode by external programmer.
- **SRAM Memory**
 - 8K bytes embedded SRAM.
- **Clock Control**
 - Flexible selection for different applications.
 - Support PLL, up to 48MHz, for high performance system operation.
 - External 12MHz (or 6MHz) crystal input for USB and precise timing operating mode.
 - External 32KHz crystal input for RTC function and system clock.
 - Internal 24MHz RC oscillator.
- **GPIO**
 - Four I/O modes:
 - ◆ Quasi bi-direction
 - ◆ Push-Pull output
 - ◆ Open-Drain output
 - ◆ Input only with high impedance
 - TTL/Schmitt trigger input selectable.
 - I/O pin can be configured as interrupt source with edge/level setting.
 - High driver and high sink IO mode support.
- **Timers**
 - 3 sets of the timer with 8-bit pre-scalar and 16-bit timer.
 - Counter auto reload.
 - IR carrier generator
 - One fixed frequency timer
- **Watch Dog Timer**
 - Default ON/OFF by configuration setting
 - Multiple clock sources
 - 8 selectable time out period from 6ms ~ 3.0sec (depends on clock source)
 - WDT can wake up power down/sleep.
 - Interrupt or reset selectable on watchdog time-out.
- **RTC**
 - Support time out interrupt
 - Support wake up function.
- **PWM/Capture/Compare Timer**
 - one 16-bit timers and four 16-bit comparators

- Four clock selectors
- One 8-bit pre-scalar and one clock dividers
- Two Dead-Zone generator
- Programmable duty control of output waveform (PWM).
- Auto reload mode or one-shot pulse mode
- Capture and compare function.
- **SPI**
 - Two sets of SPI device.
 - Master mode up to 24MHz on system runs on 48MHz.
 - Support MICROWIRE/SPI master mode (SSP)
 - Full duplex synchronous serial data transfer
 - Variable length of transfer data from 1 to 32 bits
 - MSB or LSB first data transfer
 - Rx and Tx on both rising or falling edge of serial clock independently
 - 2 slave/device select lines
 - Two 32-bit buffers
- **USB 2.0 Full-Speed Device**
 - One set of USB 2.0 FS Device 12Mbps
 - On-chip USB Transceiver.
 - Provide 1 interrupt source with 4 interrupt events.
 - Support Control, Bulk In/Out, Interrupt and Isochronous transfers.
 - Auto suspend function when no bus signaling for 3 ms.
 - Provide 6 sets configurable hardware for endpoint configurations.
 - Include 512 Bytes internal SRAM as USB buffer.
 - Provide remote wakeup capability.
- **ADC**
 - 8-ch 12-bit with 200Ksps.
 - Single scan/single cycle scan/continuous scan
 - 8 channels share 8 result registers
 - programmable channel scan sequence
 - Threshold voltage detection
 - Conversion start by S/W, external pins
 - programmable gain control for sound record
- **APU**
 - 13-bit DAC
 - H/W mixer with 2 channel PCM input
 - Embedded power amplifier
 - 7-level volume control
- **Low Voltage Detector**
 - with 2 levels: 3.0V/2.7V
- **LDO**
 - Built-in 2.4V LDO for internal core
 - Built-in 3.3V LDO for USB
- **Voltage Output**
 - Built-in 3.0V power supply Voutx for driving external device
- **Low Voltage Reset**
- **Operating Temperature: -40°C ~85°C**

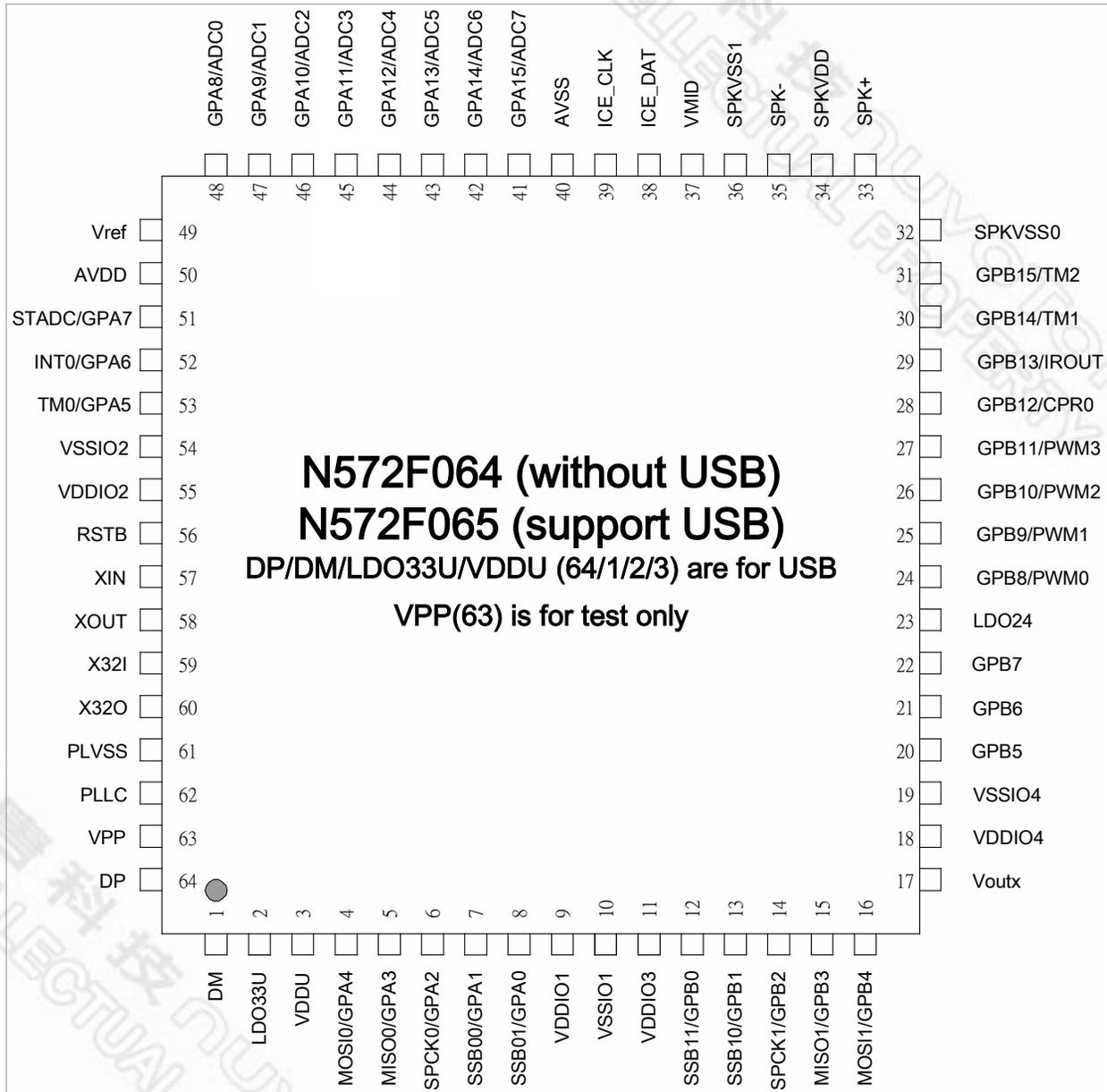
- **Package**
 - All Green package (RoHS)
 - 7mmx7mm LQFP 64-pin



3 PART INFORMATION AND PIN CONFIGURATION

3.1 Pin Configuration

3.1.1 Pin Assignment (LQFP64)



Note1: VPP is for embedded flash test; keep it no connection in general application.



3.1.2 Alternate Function List of GPIO

GPIO	Power	Alternate	I/O of Alternate	Function Description
PA.0	VDDIO1	SPI_SSB01	O	SPI0 2 nd chip select pin
PA.1	VDDIO1	SPI_SSB00	O	SPI0 1 st chip select pin
PA.2	VDDIO1	SPI_SPCK0	O	SPI0 serial clock output
PA.3	VDDIO1	SPI_MISO0	I	SPI0 master data input
PA.4	VDDIO1	SPI_MOSI0	O	SPI0 master data output
PA.5	VDDIO2	TM0	I	Timer0 counter external input
PA.6	VDDIO2	INT0	I	External interrupt input pin
PA.7	VDDIO2	STADC	I	ADC external trigger input
PA.8	VDDIO2	ADC0	A	ADC analog input 0
PA.9	VDDIO2	ADC1	A	ADC analog input 1
PA.10	VDDIO2	ADC2	A	ADC analog input 2
PA.11	VDDIO2	ADC3	A	ADC analog input 3
PA.12	VDDIO2	ADC4	A	ADC analog input 4
PA.13	VDDIO2	ADC5	A	ADC analog input 5
PA.14	VDDIO2	ADC6	A	ADC analog input 6
PA.15	VDDIO2	ADC7	A	ADC analog input 7
PB.0	VDDIO3	SPI_SSB11	O	SPI1 2 nd chip select pin
PB.1	VDDIO3	SPI_SSB10	O	SPI1 1 st chip select pin
PB.2	VDDIO3	SPI_SPCK1	O	SPI1 serial clock output
PB.3	VDDIO3	SPI_MISO1	I	SPI1 master data input
PB.4	VDDIO3	SPI_MOSI1	O	SPI1 master data output
PB.5	VDDIO4	-		
PB.6	VDDIO4	-		
PB.7	VDDIO4	-		

PB.8	VDDIO4	PWM0	O	PWM output pin 0
PB.9	VDDIO4	PWM1	O	PWM output pin 1
PB.10	VDDIO4	PWM2	O	PWM output pin 2
PB.11	VDDIO4	PWM3	O	PWM output pin 3
PB.12	VDDIO4	CPR0	I	Capture input
PB.13	VDDIO4	IROUT	O	IR carrier output
PB.14	VDDIO4	TM1	I	Timer1 counter external input
PB.15	VDDIO4	TM2	I	Timer2 counter external input

I: Input, O: Output, A: Analog input



3.1.3 Pad Description

Name	Type	Power	Description
1. GPIO			
PA.0 ~ PA.15	I/O	See 3.1.2	Bidirectional general purpose I/O ports. All of these pins have alternate function; refer to section 3.1.2 for detail.
PB.0 ~ PB.15	I/O	See 3.1.2	Bidirectional general purpose I/O ports. Most of these pins have alternate function; refer to section 3.1.2 for detail.
2. Oscillator			
X32I	I	VDDIO2	32KHz crystal input
X32O	O	VDDIO2	32KHz crystal output
XIN	I	VDDIO2	6MHz or 12MHz crystal input
XOUT	O	VDDIO2	6MHz or 12MHz crystal output
PLVSS	P	-	PLL ground
PLLC	A	-	Capacitor connection for built-in PLL1
3. USB			
DP	I/O	LDO33U	USB data pin, DP.
DM	I/O	LDO33U	USB data pin, DM.
LDO33U	P	-	3.3V output for USB
VDDU	P		USB bus power supply, floating detect input
4. ADC			
VDDA	P	-	Analog power supply
VSSA	P	-	Analog ground
Vref ADCVREF	A	-	Reference voltage input for ADC
5. Speaker Driver			
SPK+	O	SPKVDD	Speaker positive output pin
SPK-	O	SPKVDD	Speaker negative output pin.
SPKVDD	P	-	Analog power supply



SPKVSS1	P	-	Analog ground.
SPKVSS0	P	-	Analog ground.
VMID	A	-	Connect a capacitor to SPKVSS1.
6. Power			
VDDIO4	P	-	Power supply for I/O port, and source of LDO/Voutx
VSSIO4	P	-	Ground pin, connect to 0V.
VDDIO2	P	-	Power supply for I/O port, and source of oscillator. Note that VDDIO2 voltage cannot be lower than voltage of LDO24.
VSSIO2	P	-	Ground pin, connect to 0V.
VDDIO1	P	-	Power supply for I/O port.
VSSIO1	P	-	Ground pin, connect to 0V.
VDDIO3	P	-	Power supply for I/O port.
LDO24	P	-	2.4V LDO output
VOUTX	P	-	3.0V regulated power for driving out
VPP	P	-	Power supply for embedded flash programming test and keep it no connection in general application.
7. SWD			
ICE_CLK	I	VDDIO2	Serial Wired Debugger Clock pin
ICE_DAT	I/O	VDDIO2	Serial Wired Debugger Data pin
8. Other			
RSTB	I	VDDIO2	Reset input pin, low active. Internal pull-high.

Total: 64 pads

4 BLOCK DIAGRAM

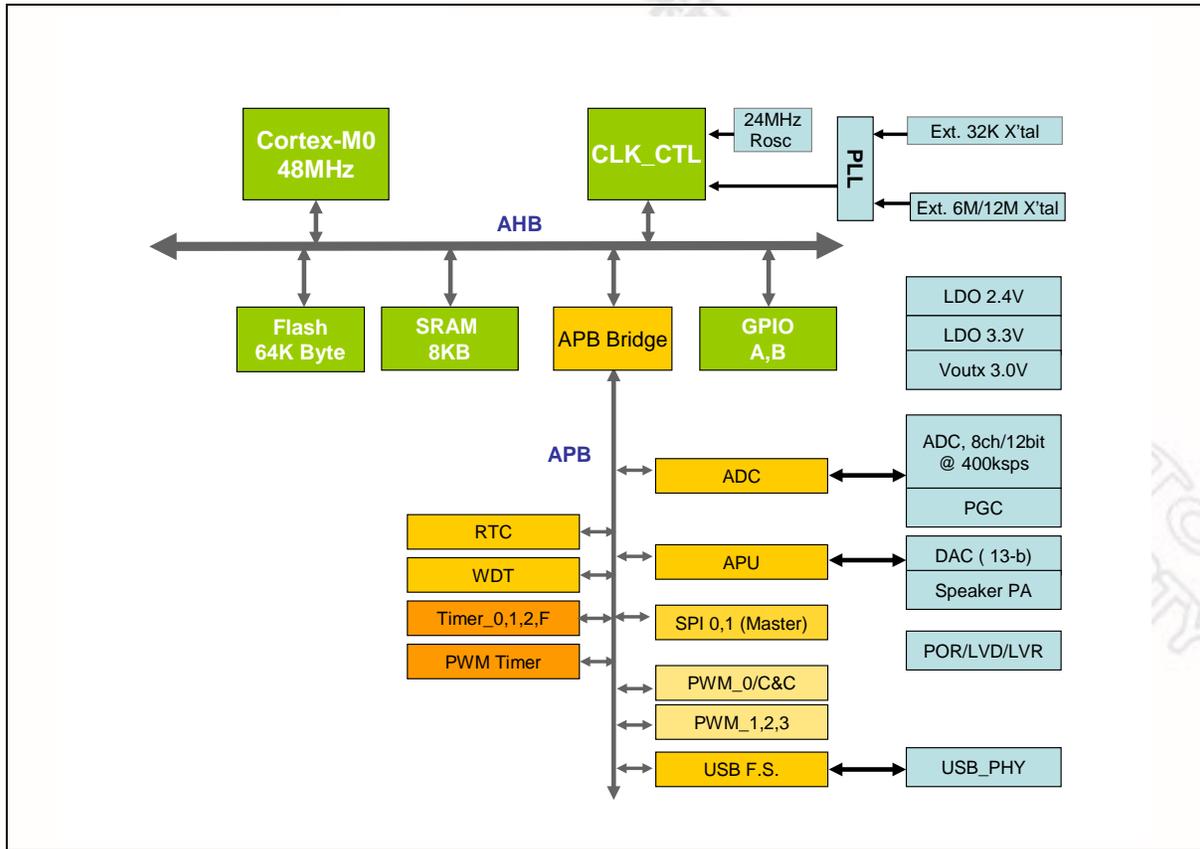


Figure 4-1 Functional Block Diagram

新唐科技 NUVOTON
 INTELLECTUAL PROPERTY

5 FUNCTIONAL DESCRIPTION

5.1 ARM® Cortex™-M0 core

The Cortex™-M0 processor is a configurable, multistage, 32-bit RISC processor. It has an AMBA AHB-Lite interface and includes an NVIC component. It also has optional hardware debug functionality. The processor can execute Thumb code and is compatible with other Cortex-M profile processor.

Figure 5-1 shows the functional blocks of processor.

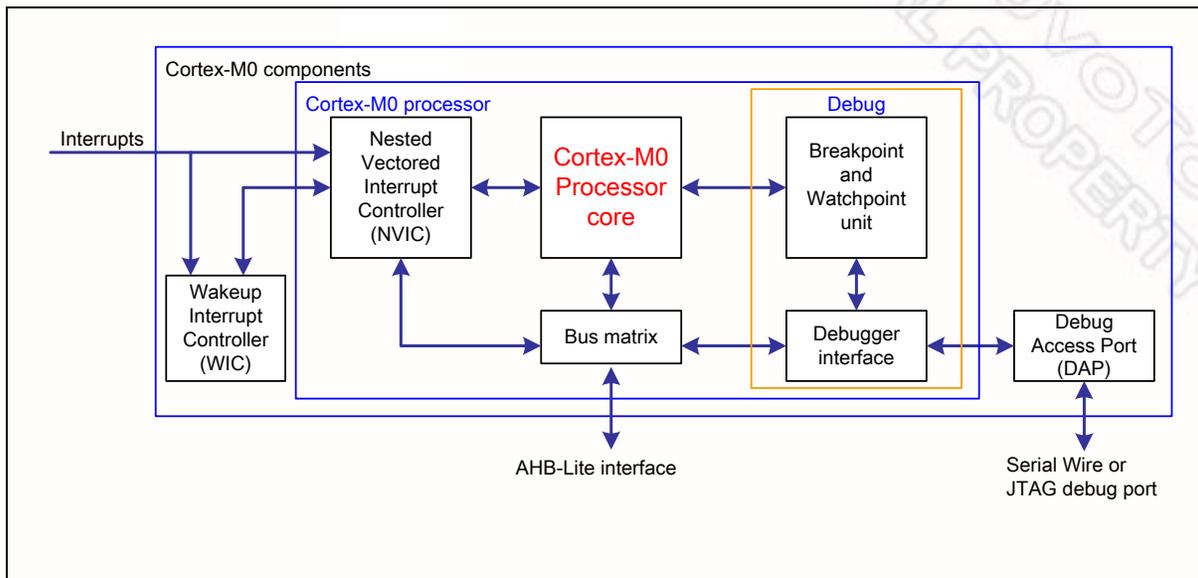


Figure 5-1 MCU Block Diagram

The implemented device provides:

- A low gate count processor that features:
 - The ARMv6-M Thumb® instruction set.
 - Thumb-2 technology.
 - ARMv6-M compliant 24-bit SysTick timer.
 - A 32-bit hardware multiplier.
 - The system interface supports little-endian data accesses.
 - The ability to have deterministic, fixed-latency, interrupt handling.
 - Load/store-multiples and multicycle-multiplies that can be abandoned and restarted to facilitate rapid interrupt handling.
 - C Application Binary Interface compliant exception model. This is the ARMv6-M, C Application Binary Interface(C-ABI) compliant exception model that enables the use of pure C functions as interrupt handlers.
 - Low power sleep-mode entry using Wait For Interrupt(WFI), Wait For Even(WFE) instructions, or the return from interrupt sleep-on-exit feature.

- NVIC features
 - 16 external interrupt inputs, each with four levels of priority.
 - Dedicated non-Maskable Interrupt (NMI) input.
 - Support for both level-sensitive and pulse-sensitive interrupt lines
 - Wake-up Interrupt Controller (WIC), providing ultra-low power sleep mode support.
- Debug support
 - Four hardware breakpoints.
 - Two watchpoints.
 - Program Counter Sampling Register (PCSR) for non-intrusive code profiling.
 - Single step and vector catch capabilities.
- Bus interfaces
 - Single 32-bit AMBA-3 AHB-Lite system interface that provides simple integration to all system peripherals and memory.
 - Single 32-bit slave port that supports the DAP.

5.2 System Manager

5.2.1 Overview

The following functions are included in system manager section

- System Memory Map
- System Interrupt Map
- System management registers for product ID
- System management registers for chip and module functional reset and multi-function pin control
- Brown-Out and chip miscellaneous Control Register
- Combined peripheral interrupt source identify



5.2.2 System Memory Map

N572F065 provides a 4G-byte address space for programmers. The memory locations assigned to each on-chip modules are shown in

Table 5-1. The detailed register and memory addressing and programming will be described in the following sections for individual on-chip modules. N572F065 series only supports little-endian data format.

Table 5-1 Address Space Assignments for On-Chip Modules

Address Space	Token	Modules	Reference
Flash & SRAM Memory Space			
0x0000_0000 – 0x0000_FFFF	FLASH_BA	FLASH Memory Space (64KB)	6.3
0x0030_0000 _ 0x0030_0003	CFG_BA	User Configuration Memory (4B)	6.4
0x2000_0000 – 0x2000_1FFF	SRAM_BA	SRAM Memory Space (8KB)	
AHB Modules Space (0x5000_0000 – 0x501F_FFFF)			
0x5000_0000 – 0x5000_01FF	SYS_BA	System Global Control Registers	5.2.3
0x5000_0200 – 0x5000_02FF	CLK_BA	Clock Control Registers	5.3.5
0x5000_0300 – 0x5000_03FF	INT_BA	Interrupt Multiplexer Control Registers	5.2.5.5
0x5000_4000 – 0x5000_7FFF	GPIO_BA	GPIO Control Registers	5.6.2
0x5000_C000 – 0x5000_FFFF	FMC_BA	Flash Memory Control Registers	6.6
APB Modules Space (0x4000_0000 ~ 0x400F_FFFF)			
0x4000_4000 – 0x4000_7FFF	WDT_BA	Watch-Dog Timer Control Registers	5.11.1
0x4000_8000 – 0x4000_BFFF	RTC_BA	Real Time Clock (RTC) Control Register	5.8.2
0x4001_0000 – 0x4001_3FFF	TMR_BA	Timer0/Timer1/Timer2/TimerF Control Registers	5.10.4
0x4003_0000 – 0x4003_3FFF	SPI0_BA	SPI0 Serial Interface Control Registers	5.9.6
0x4003_4000 – 0x4003_7FFF	SPI1_BA	SPI1 Serial Interface Control Registers	5.9.6
0x4004_0000 – 0x4004_3FFF	PWM_BA	PWM Control Registers	5.7.11
0x4006_0000 – 0x4006_3FFF	USB_BA	USB FS device Controller Registers	
0x400C_0000 – 0x400C_3FFF	APU_BA	APU Controller Registers	
0x400E_0000 – 0x400E_FFFF	ADC_BA	Analog-Digital-Converter (ADC) Registers	5.13.5
System Control Space (0xE000_E000 ~ 0xE000_EFFF)			
0xE000_E100 – 0xE000_E40F	SCS_BA	NVIC Control Registers	5.2.4.4



5.2.3 System Manager Control Registers

Register	Offset	R/W	Description	Reset Value
SYS Base Address: SYS_BA = 0x5000_0000				
SYS_PDID	SYS_BA+0x00	R	Product Identifier Register	0x0000_F065
SYS_RSTSTS	SYS_BA+0x04	R/W	System Reset Source Register	0x0000_00XX
SYS_IPRST0	SYS_BA+0x08	R/W	IP Reset Control Resister0	0x0000_0000
SYS_IPRST1	SYS_BA+0x0C	R/W	IP Reset Control Resister1	0x0000_0000
SYS_BODCTL	SYS_BA+0x18	R/W	Brown-Out Detector Control Register	0x0000_0080
SYS_PORCTL	SYS_BA+0x1C	R/W	Power-On-Reset Controller Register	0x0000_0000
SYS_GPA_MFP	SYS_BA+0x30	R/W	GPIO PA Multiple Alternate Functions and Input Type Control Register	0x0000_0000
SYS_GPB_MFP	SYS_BA+0x34	R/W	GPIO PB Multiple Alternate Functions and Input Type Control Register	0x0000_0000
SYS_REGLCTL	SYS_BA+0x100	R/W	Register Lock Control Register	0x0000_0000



Product Identifier Register (PDID)

This register provides specific read-only information for software to identify this chip.

Register	Offset	R/W	Description	Reset Value
SYS_PDID	SYS_BA+0x00	R	Product Identifier Register	0x0000_F065

31	30	29	28	27	26	25	24
PDID[31:24]							
23	22	21	20	19	18	17	16
PDID[23:16]							
15	14	13	12	11	10	9	8
PDID[15:8]							
7	6	5	4	3	2	1	0
PDID[7:0]							

Bits	Description
[31:0]	PDID Product Identifier Chip identifier (part number) for N572F065 series.



System Reset Source Register (SYS_RSTSTS)

This register provides specific information for software to identify this chip's reset source from last operation.

Register	Offset	R/W	Description	Reset Value
SYS_RSTSTS	SYS_BA+0x04	R/W	System Reset Source Register	0x0000_00XX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
RESERVED	PMURSTF	MCURF	Reserved	LVRF	WDTRF	PINRF	PORF

Bits	Description
[31:7]	Reserved Reserved.
[6]	PMURSTF Reset Source From PMU The PMURSTF flag is set by the reset signal from the PMU module to indicate the previous reset source. 0= No reset from PMU. 1= The PMU has issued the reset signal to reset the system. Note: Write 1 to clear this bit to 0.
[5]	MCURF MCU Reset Flag The MCURF flag is set by the "reset signal" from the MCU Cortex_M0 module to indicate the previous reset source. 0 = No reset from MCU. 1 = The MCU Cortex_M0 had issued the reset signal to reset the system. Note: Write 1 to clear this bit to 0.
[4]	Reserved Reserved.
[3]	LVRF LVR Reset Flag The LVR reset flag is set by the "Reset Signal" from the Low Voltage Reset Controller to indicate the previous reset source. 0 = No reset from LVR. 1 = LVR controller had issued the reset signal to reset the system. Note: Write 1 to clear this bit to 0.

[2]	WDTRF	<p>Reset Source From WDG</p> <p>The WDTRF flag is set if pervious reset source originates from the Watch-Dog module.</p> <p>0= No reset from Watch-Dog. 1= The Watch-Dog module issued the reset signal to reset the system.</p> <p>Note: Write 1 to clear this bit to 0.</p>
[1]	PINRF	<p>nRESET Pin Reset Flag</p> <p>The nRESET pin reset flag is set by the “Reset Signal” from the nRESET Pin to indicate the previous reset source.</p> <p>0 = No reset from nRESET pin. 1 = Pin nRESET had issued the reset signal to reset the system.</p> <p>Note: Write 1 to clear this bit to 0.</p>
[0]	PORF	<p>POR Reset Flag</p> <p>The POR reset flag is set by the “Reset Signal” from the Power-on Reset (POR) Controller to indicate the previous reset source.</p> <p>0 = No reset from POR. 1 = Power-on Reset (POR) Controller had issued the reset signal to reset the system.</p> <p>Note: Write 1 to clear this bit to 0.</p>



IP Reset Control Register1(SYS_IPRST0)

Register	Offset	R/W	Description	Reset Value
SYS_IPRST0	SYS_BA+0x08	R/W	IP Reset Control Resister0	0x0000_0000

To program these bits needs an open lock sequence, write “59h”, “16h”, “88h” to register SYS_REGLCTL to un-lock these bits. Refer to the register SYS_REGLCTL at address SYS_BA+0x100.

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved				RAMWS	CPUWS	CPURST	CHIPRST

Bits	Description
[31:4]	Reserved Reserved.
[3]	RAMWS Wait State Control For CPU Access RAM 0 = 1 HCLK clock wait-state. 1 = zero wait-state.
[2]	CPUWS CPU Wait-State Control For Flash Memory Access 0 = 1 HCLK clock wait-state. 1 = zero wait-state. Note: that CPUWS cannot be set as “1” when CPU runs the program to do Flash ISP operation.
[1]	CPURST CPU Kernel One Shot Reset Setting this bit will reset the CPU kernel and Flash Memory Controller(FMC), this bit will automatically return to “0” after the 2 clock cycles 0 = Normal. 1 = Reset CPU.
[0]	CHIPRST CHIP One Shot Reset Set this bit will reset the whole chip, this bit will automatically return to “0” after 2 clock cycles. The CHIPRST is almost the same as the POR reset, all the chip modules are reset but the chip settings from flash are not reloaded. 0 = Normal. 1 = Reset CHIP.



IP Reset Control Register1 (SYS_IPRST1)

Setting these bits “1” will generate an asynchronous reset signal to the corresponding peripheral block. The user needs to set bit to “0” to release block from the reset state.

Register	Offset	R/W	Description	Reset Value
SYS_IPRST1	SYS_BA+0x0C	R/W	IP Reset Control Register1	0x0000_0000

31	30	29	28	27	26	25	24
Reserved			ADCRST	USBDRST	Reserved		
23	22	21	20	19	18	17	16
Reserved			PWMRST	Reserved			
15	14	13	12	11	10	9	8
Reserved		SPI1RST	SPI0RST	Reserved			
7	6	5	4	3	2	1	0
Reserved	TMRFRST	APURST	TMR2RST	TMR1RST	TMR0RST	GPIORST	Reserved

Bits	Description	
[31:29]	Reserved	Reserved.
[28]	ADCRST	ADC Controller Reset 0 = Normal Operation. 1 = IP reset
[27]	USBDRST	USB Device Controller Reset 0 = Normal operation 1 = IP reset
[27:21]	Reserved	Reserved.
[20]	PWMRST	PWM Controller Reset 0 = Normal Operation. 1 = IP reset
[19:14]	Reserved	Reserved.
[13]	SPI1RST	SPI1 Controller Reset 0 = Normal Operation. 1 = IP reset
[12]	SPI0RST	SPI0 Controller Reset 0 = Normal Operation. 1 = IP reset
[11:7]	Reserved	Reserved.



[6]	TMRFRST	TimerF Controller Reset 0 = Normal operation. 1 = IP reset
[5]	APURST	APU Controller Reset 0 = Normal operation. 1 = IP reset
[4]	TMR2RST	Timer2 Controller Reset 0 = Normal operation. 1 = IP reset
[3]	TMR1RST	Timer1 Controller Reset 0 = Normal Operation. 1 = IP reset
[2]	TMR0RST	Timer0 Controller Reset 0 = Normal Operation. 1 = IP reset
[1]	GPORST	GPIO Controller Reset 0 = Normal operation. 1 = IP reset
[0]	Reserved	Reserved.



Brown-Out Detector Control Register (SYS_BODCTL)

Register	Offset	R/W	Description	Reset Value
SYS_BODCTL	SYS_BA+0x18	R/W	Brown-Out Detector Control Register	0x0000_0080

Partial of the SYS_BODCTL control register values are initiated by the flash configuration. After the power on initialization, these bits are protected by the lock circuit. To program these bits needs an open lock sequence, write "59h", "16h", "88h" to address 0x5000_0100 to un-lock these bits. Refer to the register SYS_REGLCTL at address SYS_BA+0x100.

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
LVR_EN	BOD_OUT	Reserved				BOD_VL	BOD_EN

Bits	Description
[31:8]	Reserved Reserved.
[7]	LVR_EN Low Voltage Reset (LVR) Enable (Protected Bit) The LVR function resets the chip when the input power voltage is lower than LVR circuit setting. LVR function is enabled in default. 0 = Disable LVR function. 1 = Enable LVR function – After enable the bit, the LVR function will active with 100uS delay for LVR output stable.
[6]	BOD_OUT The Status For Brown-Out Detector Output It's a read only bit. 0 = The detected voltage is lower than BOD_VL setting. If the BOD_EN is "0", this bit always responses "0". 1 = The detected voltage is higher than BOD_VL setting.
[5:2]	Reserved Reserved.
[1]	BOD_VL Brown-Out Detector Threshold Voltage Selection (Initiate & Protected Bit) The default value is set by flash controller user configuration register CONFIG[21]. 0 = Threshold voltage is 2.7V. 1 = Threshold voltage is 3.0V.
[0]	BOD_EN Brown-Out Detector Enable (Initiated & Protected Bit) The default value is set according to the flash controller User Configuration Register CONFIG[23], BOD_EN = ~CONFIG[23]. 0 = Brown-Out Detector function is disabled. 1 = Brown-Out Detector function is enabled,.



Power-On-Reset Controller Register (SYS PORCTL)

Register	Offset	R/W	Description	Reset Value
SYS_PORCTL	SYS_BA+0x1C	R/W	Power-On-Reset Controller Register	0x0000_0000

This register is a protected register. To program this needs an open lock sequence, write “59h”, “16h”, “88h” to address 0x5000_0100 to un-lock this bit. Refer to the register SYS_REGLCTL at address SYS_BA+0x100.

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
POROFF[15:8]							
7	6	5	4	3	2	1	0
POROFF[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	POROFF	<p>Power-On Reset Enable Bit (Write Protect)</p> <p>When powered on, the POR circuit generates a reset signal to reset the whole chip function, but noise on the power may cause the POR active again. User can disable internal POR circuit to avoid unpredictable noise to cause chip reset by writing 0x5AA5 to this field.</p> <p>The POR function will be active again when this field is set to another value or chip is reset by other reset source, including:</p> <p>nRESET, Watchdog, LVR reset, ICE reset command and the software-chip reset function.</p>



GPIO PA Multiple Alternate Function and Input Type Control Register (SYS GPA MFP)

Register	Offset	R/W	Description	Reset Value
SYS_GPA_MFP	SYS_BA+0x30	R/W	GPIO PA Multiple Alternate Functions and Input Type Control Register	0x0000_0000

31	30	29	28	27	26	25	24
PA15TYPE	PA14TYPE	PA13TYPE	PA12TYPE	PA11TYPE	PA10TYPE	PA9TYPE	PA8TYPE
23	22	21	20	19	18	17	16
PA7TYPE	PA6TYPE	PA5TYPE	PA4TYPE	PA3TYPE	PA2TYPE	PA1TYPE	PA0TYPE
15	14	13	12	11	10	9	8
PA15MFP	PA14MFP	PA13MFP	PA12MFP	PA11MFP	PA10MFP	PA9MFP	PA8MFP
7	6	5	4	3	2	1	0
PA7MFP	PA6MFP	PA5MFP	PA4MFP	PA3MFP	PA2MFP	PA1MFP	PA0MFP

Bits	Description
[31:16]	PAnTYPEn 0 = PA.n I/O cell input Schmitt Trigger function is disabled. 1 = PA.n I/O cell input Schmitt Trigger function is enabled.
[15]	PA15MFP 0 = The GPIOA-15 is selected to the pin PA.15. 1 = ADC input channel 7.
[14]	PA14MFP 0 = The GPIOA-14 is selected to the pin PA.14. 1 = ADC input channel 6.
[13]	PA13MFP 0 = The GPIOA-13 is selected to the pin PA.13. 1 = ADC input channel 5.
[12]	PA12MFP 0 = The GPIOA-12 is selected to the pin PA.12. 1 = ADC input channel 4.
[11]	PA11MFP 0 = The GPIOA-11 is selected to the pin PA.11. 1 = ADC input channel 3.
[10]	PA10MFP 0 = The GPIOA-10 is selected to the pin PA.10. 1 = ADC input channel 2.
[9]	PA9MFP 0 = The GPIOA-9 is selected to the pin PA.9. 1 = ADC input channel 1.
[8]	PA8MFP 0 = The GPIOA-8 is selected to the pin PA.8. 1 = ADC input channel 0.
[7]	PA7MFP 0 = The GPIOA-7 is selected to the pin PA.7. 1 = ADC input external trigger input.
[6]	PA6MFP 0 = The GPIOA-6 is selected to the pin PA.6. 1 = External interrupt input.

NuVoice™ N572F065 Technical Reference Manual

[5]	PA5MFP	0 = The GPIOA-5 is selected to the pin PA.5. 1 = Timer0 counter external input.
[4]	PA4MFP	0 = The GPIOA-4 is selected to the pin PA.4. 1 = SPI0 data output.
[3]	PA3MFP	0 = The GPIOA-3 is selected to the pin PA.3. 1 = SPI0 data input.
[2]	PA2MFP	0 = The GPIOA-2 is selected to the pin PA.2. 1 = SPI0 clock output.
[1]	PA1MFP	0 = The GPIOA-1 is selected to the pin PA.1. 1 = SPI0 1st chip select output.
[0]	PA0MFP	0 = The GPIOA-0 is selected to the pin PA.0. 1 = SPI0 2 nd chip select output.


GPIO PB Multiple Alternate Function and Input Type Control Register (SYS_GPB_MFP)

Register	Offset	R/W	Description	Reset Value
SYS_GPB_MFP	SYS_BA+0x34	R/W	GPIO PB Multiple Alternate Functions and Input Type Control Register	0x0000_0000

31	30	29	28	27	26	25	24
PB15TYPE	PB14TYPE	PB13TYPE	PB12TYPE	PB11TYPE	PB10TYPE	PB9TYPE	PB8TYPE
23	22	21	20	19	18	17	16
PB7TYPE	PB6TYPE	PB5TYPE	PB4TYPE	PB3TYPE	PB2TYPE	PB1TYPE	PB0TYPE
15	14	13	12	11	10	9	8
PB15MFP	PB14MFP	PB13MFP	PB12MFP	PB11MFP	PB10MFP	PB9MFP	PB8MFP
7	6	5	4	3	2	1	0
Reserved			PB4MFP	PB3MFP	PB2MFP	PB1MFP	PB0MFP

Bits	Description	
[31:16]	PBnTYPE_n	0 = PB.n I/O cell input Schmitt Trigger function is disabled. 1 = PB.n I/O cell input Schmitt Trigger function is enabled.
[15]	PB15MFP	0 = The GPIOB-15 is selected to the pin PB.15. 1 = Timer2 counter external input.
[14]	PB14MFP	0 = The GPIOB-14 is selected to the pin PB.14. 1 = Timer1 counter external input.
[13]	PB13MFP	0 = The GPIOB-13 is selected to the pin PB.13. 1 = IR carrier output.
[12]	PB12MFP	0 = The GPIOB-12 is selected to the pin PB.12. 1 = PWM timer capture input.
[11]	PB11MFP	0 = The GPIOB-11 is selected to the pin PB.11. 1 = PWM output pin 3.
[10]	PB10MFP	0 = The GPIOB-10 is selected to the pin PB.10. 1 = PWM output pin 2.
[9]	PB9MFP	0 = The GPIOB-9 is selected to the pin PB.9. 1 = PWM output pin 1.
[8]	PB8MFP	0 = The GPIOB-8 is selected to the pin PB.8. 1 = PWM output pin 0.
[7:5]	Reserved	Reserved.
[4]	PB4MFP	0 = The GPIOB-4 is selected to the pin PB.4. 1 = SPI1 data output/input.
[3]	PB3MFP	0 = The GPIOB-3 is selected to the pin PB.3. 1 = SPI1 data input/output.

[2]	PB2MFP	0 = The GPIOB-2 is selected to the pin PB.2. 1 = SPI1 clock output/input.
[1]	PB1MFP	0 = The GPIOB-1 is selected to the pin PB.1. 1 = SPI1 1st chip select output or slave select input.
[0]	PB0MFP	0 = The GPIOB-0 is selected to the pin PB.0. 1 = SPI1 2 nd chip select output.

Register Lock Control Register (SYS_REGLCTL)

Certain critical system control registers are protected against inadvertent write operations which may disturb chip operation. These system control registers are locked after power on reset until the user specifically issues an unlock sequence to disable register protection. The unlock sequence is to write to SYS_REGLCTL the data 0x59, 0x16, 0x88 sequentially. Any different data value, different sequence or any other write to other address during these three data writing will abort the whole sequence.

User can check the lock status by reading SYS_REGLCTL bit0: "1" is unlocked, "0" is locked. Once unlocked, user can update the target protected register value. To lock registers again, write any data to the register SYS_REGLCTL to enable register protection.

This register is "write" accessible to disable/enable register protection and "read" accessible to know the lock/unlock status.

Register	Offset	R/W	Description	Reset Value
SYS_REGLCTL	SYS_BA+0x100	R/W	Register Lock Control Register	0x0000_0000

7	6	5	4	3	2	1	0
SYS_REGLCTL[7:1]							SYS_REGLCTL[0] REGLCTL

Bits	Description
[31:8]	Reserved Reserved.
[7:0]	SYS_REGLCTL[7:0] Register Lock Control Code (Write Only) Some registers have write-protection function. Writing these registers have to disable the protected function by writing the sequence value "59h", "16h", "88h" to this field. After this sequence is completed, the REGLCTL bit will be set to 1 and write-protection registers can be normal write.

[0]	REGLCTL	<p>Protected Register Lock/Unlock Index (Read Only)</p> <p>0 = Protected registers are locked. Any write to the target register is ignored. 1 = Protected registers are unlocked.</p> <p>The protected registers are:</p> <p>SYS_IPRST0 – address 0x5000_0008 SYS_BODCTL – address 0x5000_0018 SYS_PORCTL – address 0x5000_001C CLK_PWRCTL – address 0x5000_0200 (bit[6] is not protected for power wake-up interrupt clear) CLK_APBCLK bit[0] – address 0x5000_0208 (bit[0] is watch dog clock enable) CLK_CLKSEL0 – address 0x5000_0210 (for HCLK clock source select) CLK_CLKSEL1 – address 0x5000_0214 (bit[1:0] is watch dog clock selection) NMI_SEL[7]= IRQ_TM – address 0x5000_0380. SPI0_RCLK – address 0x4003_0030 FMC_ISPCTL – address 0x5000_C000 (Flash ISP Control register) WDT_CTL – address 0x4000_4000 FATCON – address 0x5000_C018</p>
-----	---------	--

5.2.4 Nested Vectored Interrupt Controller (NVIC)

Cortex-M0 provides an interrupt controller as an integral part of the exception mode, named as “Nested Vectored Interrupt Controller (NVIC)”. It is closely coupled to the processor kernel and provides following features:

- Nested and Vectored interrupt support
- Automatic processor state saving and restoration
- Dynamic priority changing
- Reduced and deterministic interrupt latency

The NVIC prioritizes and handles all supported exceptions. All exceptions are handled in “Handler Mode”. This NVIC architecture supports 16 (IRQ[15:0]) discrete interrupts with 4 levels of priority. All of the interrupts and most of the system exceptions can be configured to different priority levels. When an interrupt occurs, the NVIC will compare the priority of the new interrupt to the current running one’s priority. If the priority of the new interrupt is higher than the current one, the new interrupt handler will override the current handler.

When any interrupts is accepted, the starting address of the interrupt service routine (ISR) is fetched from a vector table in memory. There is no need to determine which interrupt is accepted and branch to the starting address of the correlated ISR by software. While the starting address is fetched, NVIC will also automatically save processor state including the registers “PC, PSR, LR, R0~R3, R12” to the stack. At the end of the ISR, the NVIC will restore the mentioned registers from stack and resume the normal execution. Thus it will take less and deterministic time to process the interrupt request.

The NVIC supports “Tail Chaining” which handles back-to-back interrupts efficiently without the overhead of states saving and restoration and therefore reduces delay time in switching to pending ISR at the end of current ISR. The NVIC also supports “Late Arrival” which improves the efficiency of concurrent ISRs. When a higher priority interrupt request occurs before the current ISR starts to execute (at the stage of state saving and starting address fetching), the NVIC will give priority to the higher one without delay penalty. Thus it advances the real-time capability.

For more detailed information, please refer to the documents [“ARM® Cortex™-M0 Technical Reference Manual”](#) and [“ARM® v6-M Architecture Reference Manual”](#).

5.2.4.1 Exception Model and System Interrupt Map

The following table lists the exception model supported by N572F072/P072. Software can set four levels of priority on certain exceptions as well as on all interrupts. The highest user-configurable priority is denoted as “0” and the lowest priority is denoted as “3”. The default priority of all the user-configurable interrupts is “0”. Note that priority “0” is treated as the fourth priority on the system, after three system exceptions “Reset”, “NMI” and “Hard Fault”.

Table 5-2 Exception Model

Exception Name	Vector Number	Priority
Reset	1	-3
NMI	2	-2
Hard Fault	3	-1
Reserved	4 ~ 10	N/A
SVCall	11	Configurable
Reserved	12 ~ 13	N/A
PendSV	14	Configurable
SysTick	15	Configurable
Interrupt (IRQ0 ~ IRQ15)	16 ~ 31	Configurable

Table 5-3 System Interrupt Map

Vector Number	Interrupt Number (Bit In Interrupt Registers)	Interrupt Name	Source IP	Interrupt Description
0 ~ 15	-	-	-	System exceptions
16	0	WDT_INT	WDT	Watch Dog Timer interrupt
17	1	APU_INT	APU	APU threshold interrupt
18	2	ADC_INT	ADC0~7	Interrupt from ADC
19	3	EXINT	GPIO	External interrupt
20	4	USB_INT	USBD	USB FS Device Interrupt
21	5	TMR0_INT	Timer0	Timer0
22	6	TMR1_INT	Timer1	Timer1
23	7	TMR2_INT	Timer2	Timer2
24	8	GPAB_INT	GPIO A/B	Port interrupt from GPIO port
25	9	SPI0_INT	SPI0	Interrupt from SPI0
26	10	PWM0_INT	PWM Timer	PWM Timer interrupt
27	11	SPI1_INT	SPI1	Interrupt from SPI1
28	12	TMRF_INT	TimerF	Fixed frequency TimerF
29	13	RTC_INT	RTC	Real time clock interrupt
30	14	PWRWU_INT	CLKC	Interrupt form Clock controller for chip wake up from power-down state
31	15	Reserved	-	-

5.2.4.2 Vector Table

When an interrupt is accepted, the processor will automatically fetch the starting address of the interrupt service routine (ISR) from the vector table in memory. For ARMv6-M, the vector table base address is fixed in flash at 0x00000000. The vector table contains the initialization value for the stack pointer on reset, and the entry point addresses for all exception handlers. The vector number on previous page defines the order of entries in the vector table associated with exception handler entry.

Table 5-4 Vector Table Format

Vector Table Word Offset	Description
0	SP_main - The Main stack pointer
Vector Number	Exception Entry Pointer using that Vector Number

5.2.4.3 Operation Description

NVIC interrupts can be enabled and disabled by writing to their corresponding Interrupt Set-Enable or Interrupt Clear-Enable register bit-field. The registers use a write-1-to-enable and write-1-to-clear policy, both registers reading back the current enabled state of the corresponding interrupts. When an interrupt is disabled, interrupt assertion will cause the interrupt to become Pending, however, the interrupt will not activate. If an interrupt is Active when it is disabled, it remains in its Active state until cleared by reset or an exception return. Clearing the enable bit prevents new activations of the associated interrupt.

NVIC interrupts can be pended/un-pended using a complementary pair of registers to those used to enable/disable the interrupts, named the Set-Pending Register and Clear-Pending Register respectively. The registers use a write-1-to-enable and write-1-to-clear policy, both registers reading back the current pended state of the corresponding interrupts. The Clear-Pending Register has no effect on the execution status of an Active interrupt.

NVIC interrupts are prioritized by updating an 8-bit field within a 32-bit register (each register supporting four interrupts).

The general registers associated with the NVIC are all accessible from a block of memory in the System Control Space and will be described in next section.



5.2.4.4 NVIC Control Registers

R: read only, W: write only, R/W: both read and write, W&C: Write 1 clear

Register	Offset	R/W	Description	Reset Value
SCS Base Address: SCS_BA = 0xE000_E000				
NVIC_ISER	SCS_BA+0x100	R/W	IRQ0 ~ IRQ15 Set-Enable Control Register	0x0000_0000
NVIC_ICER	SCS_BA+0x180	R/W	IRQ0 ~ IRQ15 Clear-Enable Control Register	0x0000_0000
NVIC_ISPR	SCS_BA+0x200	R/W	IRQ0 ~ IRQ15 Set-Pending Control Register	0x0000_0000
NVIC_ICPR	SCS_BA+0x280	R/W	IRQ0 ~ IRQ15 Clear-Pending Control Register	0x0000_0000
NVIC_IPR0	SCS_BA+0x400	R/W	IRQ0 ~ IRQ3 Priority Control Register	0x0000_0000
NVIC_IPR1	SCS_BA+0x404	R/W	IRQ4 ~ IRQ7 Priority Control Register	0x0000_0000
NVIC_IPR2	SCS_BA+0x408	R/W	IRQ8 ~ IRQ11 Priority Control Register	0x0000_0000
NVIC_IPR3	SCS_BA+0x40C	R/W	IRQ12 ~ IRQ15 Priority Control Register	0x0000_0000



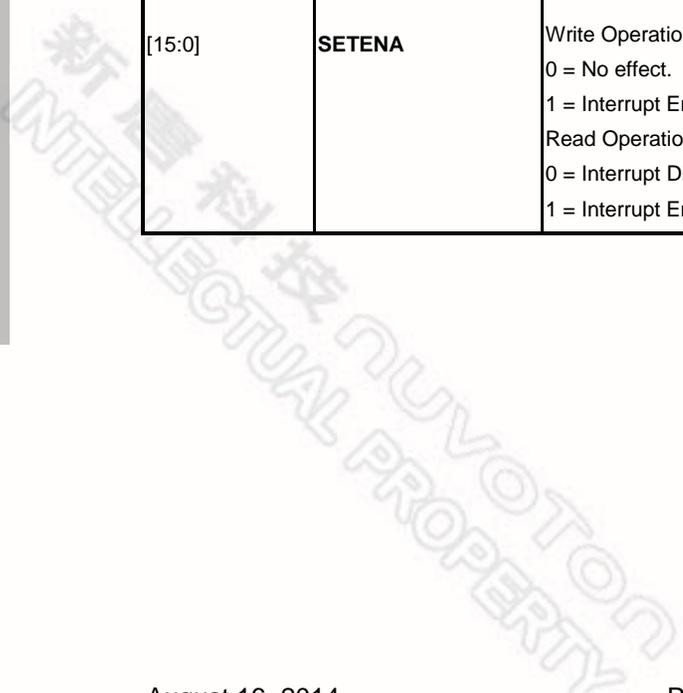
IRQ0 ~ IRQ15 Set-Enable Control Register (NVIC_ISER)

Register	Offset	R/W	Description	Reset Value
NVIC_ISER	SCS_BA+0x100	R/W	IRQ0 ~ IRQ15 Set-Enable Control Register	0x0000_0000

If a pending interrupt is enabled, the NVIC activates the interrupt based on its priority. If an interrupt is not enabled, asserting its interrupt signal changes the interrupt state to pending, but the NVIC never activates the interrupt, regardless of its priority.

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
SETENA[15:8]							
7	6	5	4	3	2	1	0
SETENA[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	SETENA	<p>Interrupt Set-Enable Bit</p> <p>The NVIC_ISER register enables interrupts, and shows what interrupts are enabled. Each bit represents an interrupt number from IRQ0 ~ IRQ15 (Vector number from 16 ~ 31).</p> <p>Write Operation: 0 = No effect. 1 = Interrupt Enabled.</p> <p>Read Operation: 0 = Interrupt Disabled. 1 = Interrupt Enabled.</p>





IRQ0 ~ IRQ15 Clear-Enable Control Register (NVIC_ICER)

Register	Offset	R/W	Description	Reset Value
NVIC_ICER	SCS_BA+0x180	R/W	IRQ0 ~ IRQ15 Clear-Enable Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
CLRENA[15:8]							
7	6	5	4	3	2	1	0
CLRENA[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	CLRENA	<p>Interrupt Clear-Enable Bit</p> <p>The NVIC_ICER register disables interrupts, and shows what interrupts are enabled. Each bit represents an interrupt number from IRQ0 ~ IRQ15 (Vector number from 16 ~ 31).</p> <p>Write Operation: 0 = No effect. 1 = Interrupt Disabled.</p> <p>Read Operation: 0 = Interrupt Disabled. 1 = Interrupt Enabled.</p>



IRQ0 ~ IRQ15 Set-Pending Control Register (NVIC_ISPR)

Register	Offset	R/W	Description	Reset Value
NVIC_ISPR	SCS_BA+0x200	R/W	IRQ0 ~ IRQ15 Set-Pending Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
SETPEND[15:8]							
7	6	5	4	3	2	1	0
SETPEND[7:0]							

Bits	Description
[31:16]	Reserved Reserved.
[15:0]	<p>SETPEND</p> <p>Interrupt Set-Pending Bit</p> <p>The NVIC_ISPR register forces interrupts into the pending state, and shows what interrupts are pending. Each bit represents an interrupt number from IRQ0 ~ IRQ15 (Vector number from 16 ~ 31).</p> <p>Write Operation: 0 = No effect. 1 = Changes interrupt state to pending.</p> <p>Read Operation: 0 = Interrupt is not pending. 1 = Interrupt is pending.</p>



IRQ0 ~ IRQ15 Clear-Pending Control Register (NVIC_ICPR)

Register	Offset	R/W	Description	Reset Value
NVIC_ICPR	SCS_BA+0x280	R/W	IRQ0 ~ IRQ15 Clear-Pending Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
CLRPEND[15:8]							
7	6	5	4	3	2	1	0
CLRPEND[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	CLRPEND	<p>Interrupt Clear-Pending Bit</p> <p>The NVIC_ICPR register removes the pending state of associated interrupts, and shows what interrupts are pending. Each bit represents an interrupt number from IRQ0 ~ IRQ15 (Vector number from 16 ~ 31).</p> <p>Write Operation:</p> <p>0 = No effect. 1 = Removes pending state of an interrupt.</p> <p>Read Operation:</p> <p>0 = Interrupt is not pending. 1 = Interrupt is pending.</p>



IRQ0 ~ IRQ3 Interrupt Priority Register (NVIC_IPR0)

Register	Offset	R/W	Description	Reset Value
NVIC_IPR0	SCS_BA+0x400	R/W	IRQ0 ~ IRQ3 Priority Control Register	0x0000_0000

31	30	29	28	27	26	25	24
PRI_3		Reserved					
23	22	21	20	19	18	17	16
PRI_2		Reserved					
15	14	13	12	11	10	9	8
PRI_1		Reserved					
7	6	5	4	3	2	1	0
PRI_0		Reserved					

Bits	Description	
[31:30]	PRI_3	Priority Of IRQ3 "0" denotes the highest priority and "3" denotes lowest priority
[23:22]	PRI_2	Priority Of IRQ2 "0" denotes the highest priority and "3" denotes lowest priority
[15:14]	PRI_1	Priority Of IRQ1 "0" denotes the highest priority and "3" denotes lowest priority
[7:6]	PRI_0	Priority Of IRQ0 "0" denotes the highest priority and "3" denotes lowest priority



IRQ4 ~ IRQ7 Interrupt Priority Register (NVIC IPR1)

Register	Offset	R/W	Description	Reset Value
NVIC_IPR1	SCS_BA+0x404	R/W	IRQ4 ~ IRQ7 Priority Control Register	0x0000_0000

31	30	29	28	27	26	25	24
PRI_7		Reserved					
23	22	21	20	19	18	17	16
PRI_6		Reserved					
15	14	13	12	11	10	9	8
PRI_5		Reserved					
7	6	5	4	3	2	1	0
PRI_4		Reserved					

Bits	Description	
[31:30]	PRI_7	Priority Of IRQ7 "0" denotes the highest priority and "3" denotes lowest priority
[23:22]	PRI_6	Priority Of IRQ6 "0" denotes the highest priority and "3" denotes lowest priority
[15:14]	PRI_5	Priority Of IRQ5 "0" denotes the highest priority and "3" denotes lowest priority
[7:6]	PRI_4	Priority Of IRQ4 "0" denotes the highest priority and "3" denotes lowest priority



IRQ8 ~ IRQ11 Interrupt Priority Register (NVIC_IPR2)

Register	Offset	R/W	Description	Reset Value
NVIC_IPR2	SCS_BA+0x408	R/W	IRQ8 ~ IRQ11 Priority Control Register	0x0000_0000

31	30	29	28	27	26	25	24
PRI_11		Reserved					
23	22	21	20	19	18	17	16
PRI_10		Reserved					
15	14	13	12	11	10	9	8
PRI_9		Reserved					
7	6	5	4	3	2	1	0
PRI_8		Reserved					

Bits	Description	
[31:30]	PRI_11	Priority Of IRQ11 "0" denotes the highest priority and "3" denotes lowest priority
[23:22]	PRI_10	Priority Of IRQ10 "0" denotes the highest priority and "3" denotes lowest priority
[15:14]	PRI_9	Priority Of IRQ9 "0" denotes the highest priority and "3" denotes lowest priority
[7:6]	PRI_8	Priority Of IRQ8 "0" denotes the highest priority and "3" denotes lowest priority





IRQ12 ~ IRQ15 Interrupt Priority Register (NVIC IPR3)

Register	Offset	R/W	Description	Reset Value
NVIC_IPR3	SCS_BA+0x40C	R/W	IRQ12 ~ IRQ15 Priority Control Register	0x0000_0000

31	30	29	28	27	26	25	24
PRI_15		RESERVED					
23	22	21	20	19	18	17	16
PRI_14		RESERVED					
15	14	13	12	11	10	9	8
PRI_13		RESERVED					
7	6	5	4	3	2	1	0
PRI_12		RESERVED					

Bits	Description	
[31:30]	PRI_15	Priority Of IRQ15 "0" denotes the highest priority and "3" denotes lowest priority
[23:22]	PRI_14	Priority Of IRQ14 "0" denotes the highest priority and "3" denotes lowest priority
[15:14]	PRI_13	Priority Of IRQ13 "0" denotes the highest priority and "3" denotes lowest priority
[7:6]	PRI_12	Priority Of IRQ12 "0" denotes the highest priority and "3" denotes lowest priority



5.2.4.5 Interrupt Source Control Registers

Along with the interrupt control registers associated with the NVIC, N572F065 also implements some specific control registers to facilitate the interrupt functions, including “interrupt source identify”, “NMI source selection” and “interrupt test mode”. They are described as below.

R: read only, **W**: write only, **R/W**: both read and write

Register	Offset	R/W	Description	Reset Value
INT Base Address:				
INT_BA = 0x5000_0300				
IRQ0_SRC	INT_BA+0x00	R	IRQ0 (WDT) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ1_SRC	INT_BA+0x04	R	IRQ1 (APU) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ2_SRC	INT_BA+0x08	R	IRQ2 (ADC) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ3_SRC	INT_BA+0x0C	R	IRQ3 (EXINT) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ4_SRC	INT_BA+0x10	R	IRQ4 (USBD) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ5_SRC	INT_BA+0x14	R	IRQ5 (Timer0) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ6_SRC	INT_BA+0x18	R	IRQ6 (Timer1) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ7_SRC	INT_BA+0x1C	R	IRQ7 (Timer2) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ8_SRC	INT_BA+0x20	R	IRQ8 (GPA/B) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ9_SRC	INT_BA+0x24	R	IRQ9 (SPI0) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ10_SRC	INT_BA+0x28	R	IRQ10 (PWM) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ11_SRC	INT_BA+0x2C	R	IRQ11 (SPI1) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ12_SRC	INT_BA+0x30	R	IRQ12 (TimerF) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ13_SRC	INT_BA+0x34	R	IRQ13 (RTC) Interrupt Source Identity Register	0XXXXX_XXXX
IRQ14_SRC	INT_BA+0x38	R	IRQ14 (PWRWU) Interrupt Source Identity Register	0XXXXX_XXXX
NMI_SEL	INT_BA+0x80	R/W	NMI Source Interrupt Select Control Register	0x0000_000F
MCU_IRQ	INT_BA+0x84	R/W	MCU IRQ Number Identify Register	0x0000_0000



IRQ0(BOD) Interrupt Source Identify Register (IRQ0_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ0_SRC	INT_BA+0x00	R	IRQ0 (WDT) Interrupt Source Identity Register	0xFFFF_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: WDT_INT



IRQ1(APU) Interrupt Source Identify Register (IRQ1_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ1_SRC	INT_BA+0x04	R	IRQ1 (APU) Interrupt Source Identity Register	0xXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: APU_INT



IRQ2(ADC) Interrupt Source Identify Register (IRQ2_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ2_SRC	INT_BA+0x08	R	IRQ2 (ADC) Interrupt Source Identity Register	0xFFFF_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: ADC_INT

IRQ3(EXINT) Interrupt Source Identify Register (IRQ3_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ3_SRC	INT_BA+0x0C	R	IRQ3 (EXINT) Interrupt Source Identity Register	0xFFFF_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description
[31:3]	Reserved Reserved.
[2:0]	INT_SRC Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: External interrupt



IRQ4(USB) Interrupt Source Identify Register (IRQ4_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ4_SRC	INT_BA+0x10	R	IRQ4 (USB) Interrupt Source Identity Register	0XXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: USBD_INT



IRQ5(Timer0) Interrupt Source Identify Register (IRQ5_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ5_SRC	INT_BA+0x14	R	IRQ5 (Timer0) Interrupt Source Identity Register	0xXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description
[31:3]	Reserved
[2:0]	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: Timer0_INT



IRQ6(Timer1) Interrupt Source Identify Register (IRQ6_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ6_SRC	INT_BA+0x18	R	IRQ6 (Timer1) Interrupt Source Identity Register	0XXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: Timer1_INT



IRQ7(Timer2) Interrupt Source Identify Register (IRQ7_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ7_SRC	INT_BA+0x1C	R	IRQ7 (Timer2) Interrupt Source Identity Register	0xXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: Timer2_INT



IRQ8(GPA/B) Interrupt Source Identify Register (IRQ8_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ8_SRC	INT_BA+0x20	R	IRQ8 (GPA/B) Interrupt Source Identity Register	0xFFFF_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: GPB_INT Bit0: GPA_INT



IRQ9(SPI0) Interrupt Source Identify Register (IRQ9_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ9_SRC	INT_BA+0x24	R	IRQ9 (SPI0) Interrupt Source Identity Register	0xXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: SPI0_INT



IRQ10(PWM) Interrupt Source Identify Register (IRQ10_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ10_SRC	INT_BA+0x28	R	IRQ10 (PWM) Interrupt Source Identity Register	0xFFFF_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: PWM_INT



IRQ11(SPI1) Interrupt Source Identify Register (IRQ11_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ11_SRC	INT_BA+0x2C	R	IRQ11 (SPI1) Interrupt Source Identity Register	0xXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: SPI1_INT



IRQ12(TimerF) Interrupt Source Identify Register (IRQ12_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ12_SRC	INT_BA+0x30	R	IRQ12 (TimerF) Interrupt Source Identity Register	0XXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: TimerF_INT

IRQ13(RTC) Interrupt Source Identify Register (IRQ13_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ13_SRC	INT_BA+0x34	R	IRQ13 (RTC) Interrupt Source Identity Register	0xXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description
[31:3]	Reserved Reserved.
[2:0]	INT_SRC Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: RTC_INT



IRQ14(PWRWU) Interrupt Source Identify Register (IRQ14_SRC)

Register	Offset	R/W	Description	Reset Value
IRQ14_SRC	INT_BA+0x38	R	IRQ14 (PWRWU) Interrupt Source Identity Register	0XXXXX_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					INT_SRC[2:0]		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	INT_SRC	Interrupt Source Identity Bit2: 0 Bit1: 0 Bit0: PWRWU_INT



NMI Interrupt Source Select Control Register (NMI_SEL)

Register	Offset	R/W	Description	Reset Value
NMI_SEL	INT_BA+0x80	R/W	NMI Source Interrupt Select Control Register	0x0000_000F

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
IRQ_TM	Reserved			NMI_SEL			

Bits	Description
[31:8]	Reserved Reserved.
[7]	<p>IRQ_TM</p> <p>IRQ Test Mode This bit is the protected bit. To program this bit needs an open lock sequence, write "59h", "16h", "88h" to register SYS_REGLCTL to un-lock this bit. Refer to the register SYS_REGLCTL at address SYS_BA+0x100.</p> <p>0 = The interrupt register MCU_IRQ operates in normal mode. The MCU_IRQ collects all the interrupts from the peripheral and generates interrupt to MCU. 1 = All the interrupts from peripheral to MCU are blocked. The peripheral IRQ signals (0-15) are replaced by the value in the MCU_IRQ register.</p>
[6:4]	Reserved Reserved.
[3:0]	<p>NMI_SEL</p> <p>NMI Source Interrupt Select The NMI interrupt to Cortex-M0 can be selected from one of the interrupt[15:0]. The NMI_SEL bit is used to select the NMI interrupt source. Note: IRQ15 is reserved in N572F065</p>

MCU Interrupt Request Source Test Mode Register (MCU_IRQ)

Register	Offset	R/W	Description	Reset Value
MCU_IRQ	INT_BA+0x84	R/W	MCU IRQ Number Identify Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
MCU_IRQ[15:8]							
7	6	5	4	3	2	1	0
MCU_IRQ[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	MCU_IRQ	<p>MCU IRQ Source Test Mode</p> <p>The MCU_IRQ collects all the interrupts from the peripherals and generates the synchronous interrupt to MCU Cortex-M0. There are two modes to generate interrupt to MCU Cortex-M0, the normal mode and test mode.</p> <p>In Normal mode (NMI_SEL register bit [7] = 0) The MCU_IRQ collects all interrupts from each peripheral and synchronizes them to interrupt the Cortex-M0.</p> <p>In Test mode (NMI_SEL register bit [7] = 1), the interrupts from peripherals are blocked, and the interrupts are replaced by MCU_IRQ[15:0].</p> <p>When MCU_IRQ[n] is "0" : Writing MCU_IRQ[n] "1" will generate an interrupt to Cortex_M0 IRQ[n].</p> <p>When MCU_IRQ[n] is "1" (meaning an interrupt is asserted): Writing MCU_IRQ[n] "1" will clear the interrupt; writing MCU_IRQ[n] "0": has no effect.</p> <p>Note: IRQ15 is reserved in N572F065.</p>

5.3 Clock Controller

The clock controller generates the clock sources for the whole chip. It includes all AMBA interface modules and all peripheral clocks, USB, ADC, and so on. The controller also implements the power control function, include the individually clock on or off control register, clock source select and the divided number from clock source. These functions minimize the extra power consumption and the chip run on the just clock condition. The chip will enter power-down mode after setting both the PD_WAIT_CPU and PWR_DOWN bits, asserting the signal SLEEPHOLDACKn and later the CPU Cortex-M0 executes the WFI or the WFE instruction. On the power down mode, the controller turns off the external crystal (except 32K oscillator) and internal oscillator to reduce the power consumption to minimum.

5.3.1 Clock Generator

The clock generator consists of 5 sources which are listed below:

- One external 32KHz crystal
- One external 12MHz/6MHz crystal
- One internal 24MHz RC oscillator.
- First PLL output, named as PLL1. PLL1 clock source is from 32KHz crystal, with fixed 6MHz clock output
- Second PLL output, named as PLL2. PLL2 clock source can be from 12MHz/6MHz crystal or PLL1 output, and its maximum frequency output is 48MHz

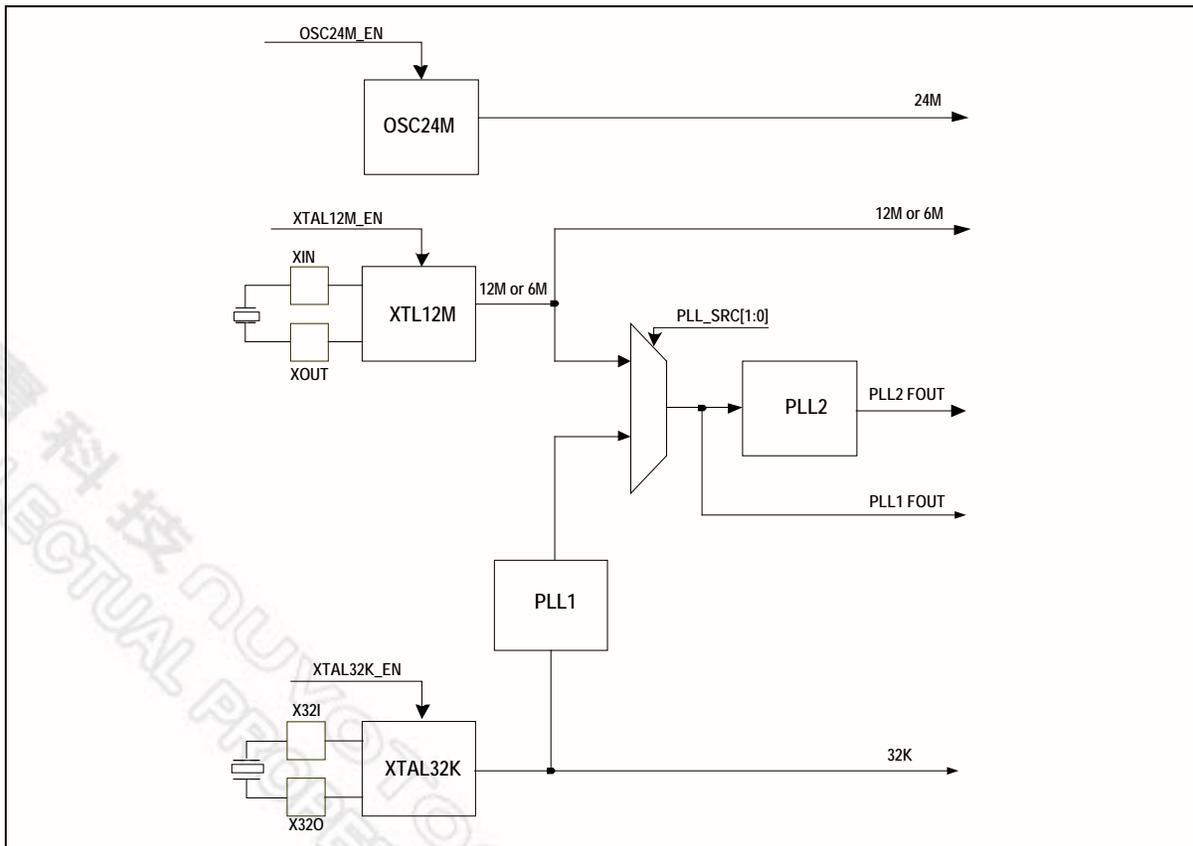


Figure 5-2 Clock generator block diagram

5.3.2 System Clock

The system clock has 5 clock sources which were generated from clock generator block. The clock source switch depends on the register HCLK_S(CLKSEL0[2:0]). The block diagram is listed below.

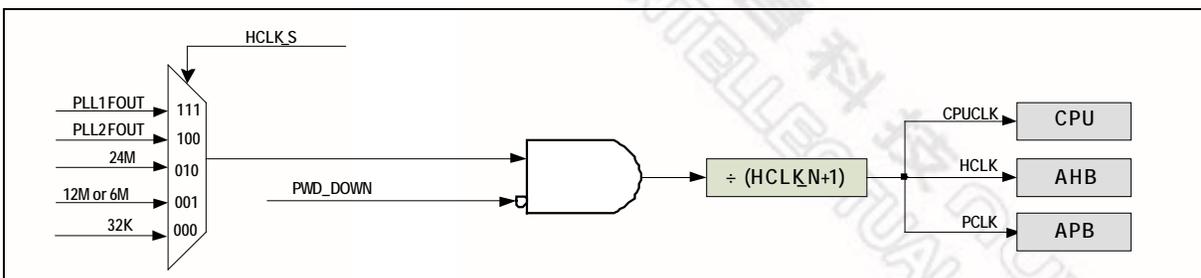


Figure 5-3 System Clock Block Diagram

5.3.3 Peripheral Clock

The peripheral clock has different clock source switch setting which depends on the different peripheral. Please refer to the descriptions in CLKSEL1 and APBCLK registers.

5.3.4 Power Down Mode Clock

When chip enters into power down mode, some clock sources and peripherals clock and system clock will be disabled. Some clock sources and peripherals clock can still be active in power down mode.

Those clocks which still can be kept active are listed below:

- Clock Generator
 - External 32K crystal clock
- Peripheral Clock (When these IP adopt 32KHz as clock source)
 - Watch Dog Clock
 - RTC Clock
 - Timer 0/1/2/F Clock
 - PWM Clock



5.3.5 Clock Control Register Map

R: read only, **W:** write only, **R/W:** both read and write

Register	Offset	R/W	Description	Reset Value
CLK Base Address: CLK_BA = 0x5000_0200				
CLK_PWRCTL	CLK_BA + 0x00	R/W	System Power Control Register	0x0000_001x
CLK_AHBCLK	CLK_BA + 0x04	R/W	AHB Device Clock Enable Control Register	0x0000_0004
CLK_APBCLK	CLK_BA + 0x08	R/W	APB Device Clock Enable Control Register	0x0000_000x
CLK_CLKSEL0	CLK_BA + 0x10	R/W	Clock Source Select Control Register 0	0x0000_000x
CLK_CLKSEL1	CLK_BA + 0x14	R/W	Clock Source Select Control Register 1	0x0000_0000
CLK_CLKDIV	CLK_BA + 0x18	R/W	Clock Divider Number Register	0x0000_0000
CLK_PLLCON	CLK_BA + 0x20	R/W	PLL Control Register	0x0000_0007



5.3.6 Clock Control Register Description

System Power Control Register (CLK_PWRCTL)

Except the BIT[6], all the other bits are protected. To program these bits needs an open lock sequence, write "59h", "16h", "88h" to register SYS_REGLCTL to un-lock these bits. Refer to the register SYS_REGLCTL at address SYS_BA + 0x100.

Register	Offset	R/W	Description	Reset Value
CLK_PWRCTL	CLK_BA + 0x00	R/W	System Power Control Register	0x0000_001x

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							PD_WAIT_CPU
7	6	5	4	3	2	1	0
PWR_DOWN	INTSTS	WINT_EN	WU_DLY	VOUTX_PD	OSC24M_EN	XTL32K_EN	XTL12M_EN

Bits	Description
[31:9]	Reserved Reserved.
[8]	<p>PD_WAIT_CPU</p> <p>This Bit Controls The Power Down Entry Condition Please refer to PWR_DOWN bit for the usage of PD_WAIT_CPU bit. The following is a brief description of PD_WAIT_CPU bit.</p> <p>0 = Chip is at normal mode. Note that PWR_DOWN cannot be set to 1 when PD_WAIT_CPU value remains at 0, otherwise the chip may not wake up normally.</p> <p>1 = Chip waits to enter power-down mode.</p>



<p>[7]</p>	<p>PWR_DOWN</p>	<p>System Power Down Active Or Enable Bit</p> <p>To make the programming simple, In order to allow the chip enter power-down mode, both the PWR_DOWN bit and PD_WAIT_CPU bit must be set to 1. To reset both PWR_DOWN=0 and PD_WAIT_CPU=0 can make sure the chip operates at normal mode.</p> <p>The below truth table summarizes the chip statuses caused by the setting of PWR_DOWN bit and PD_WAIT_CPU bit.</p> <table border="1" data-bbox="609 508 1391 982"> <thead> <tr> <th>PD_WAIT_CPU</th> <th>PWR_DOWN</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Chip is at normal operation.</td> </tr> <tr> <td>0</td> <td>1</td> <td>This setting is prohibited. The chip will shut down immediately but may not wake up normally.</td> </tr> <tr> <td>1</td> <td>0</td> <td>Chip is at normal operation. This condition exists after the chip wakes up</td> </tr> <tr> <td>1</td> <td>1</td> <td>This setting is a prerequisite for entering power-down mode. The chip still operates normally at this stage, but it is already standing by power-down entry condition. The complete procedure to enter power-down mode is described in below paragraph.</td> </tr> </tbody> </table> <p>When both the PD_WAIT_CPU and PWR_DOWN bit are set to “1”, the chip still operates normally. Until the Cortex_M0 enters sleep state and asserts the signal SLEEPHOLDACKn, then the chip enters the power-down mode.</p> <p>When the chip enters the power-down mode and all the chip clocks are disabled, except IP from the 32KHz clock.</p> <p>It is suggested to follow the below three steps in order to enter power-down mode. The following illustrates the pseudo code for these three steps,</p> <p>Step 1. CLK_PWRCTL[8]= PD_WAIT_CPU=1 and CLK_PWRCTL [7]= PWR_DOWN=1.</p> <p>Step 2. 0xE000ED10[2]=1.</p> <p>Step 3. __wfi();</p> <p>When chip wakes up from power down mode, this bit is auto cleared. User needs to set this bit again for next power down.</p> <p>In power down mode, the XIN/XOUT crystal oscillator, internal 24MHz OSC, PLL and all of the AMBA clocks are disabled. The 32KHz oscillator is still active if it has been working before entering power-down. So the RTC and WDT functions whose clocks are from 32KHz oscillator are active.</p> <p>If system wants to avoid interrupt by WDT waking up from power down mode, the WDT should be turned off before entering the power down mode.</p> <p>0 = Chip operates at normal mode. 1 = Chip is standing by power-down entry condition.</p>	PD_WAIT_CPU	PWR_DOWN	Function	0	0	Chip is at normal operation.	0	1	This setting is prohibited. The chip will shut down immediately but may not wake up normally.	1	0	Chip is at normal operation. This condition exists after the chip wakes up	1	1	This setting is a prerequisite for entering power-down mode. The chip still operates normally at this stage, but it is already standing by power-down entry condition. The complete procedure to enter power-down mode is described in below paragraph.
PD_WAIT_CPU	PWR_DOWN	Function															
0	0	Chip is at normal operation.															
0	1	This setting is prohibited. The chip will shut down immediately but may not wake up normally.															
1	0	Chip is at normal operation. This condition exists after the chip wakes up															
1	1	This setting is a prerequisite for entering power-down mode. The chip still operates normally at this stage, but it is already standing by power-down entry condition. The complete procedure to enter power-down mode is described in below paragraph.															
<p>[6]</p>	<p>WINT_STS</p>	<p>Chip Power Down Wake Up Status Flag</p> <p>Set by “power down wake up”, it indicates that resume from power down mode.</p> <p>The flag is set if the GPIO, USB, WDT or RTC wakeup.</p> <p>Note: Write 1 to clear the bit.</p>															



[5]	WINT_EN	Enable Interrupt When Wake Up From Power Down Mode 0 = Disable, 1 = Enable. The interrupt will occur when MCU wakes up from power down mode
[4]	WU_DLY	Enable The Wake Up Delay Counter When the chip wakes up from idle mode, the clock control will delay some clock cycles to wait the 12MHz (or 6MHz) crystal or the internal 24MHz oscillator clock stable. 0 = Disable the clock cycles delay 1 = Enable the clock cycles delay. The delay is 4096 clock cycles for 12MHz (or 6MHz) crystal, 256 clock cycles for 24MHz oscillator.
[3]	VOUTX_PD	Driving Out 3.0V (Through VOUTX Pad) LDO Control 0 = 3.0V LDO (VOUTX) is enabled. 1 = 3.0V LDO (VOUTX) is disabled.
[2]	OSC24M_EN	Internal 24MHz Oscillator Control After reset, this bit is "1". 0 = 24MHz oscillation is disabled. 1 = 24MHz Oscillation is enabled
[1]	XTL32K_EN	External 32.768KHz Crystal Control After reset, this bit is "0". 0 = 32.768KHz Crystal is disabled. 1 = 32.768KHz Crystal is enabled.
[0]	XTL12M_EN	External 12MHz (or 6MHz) Crystal Oscillator Control This bit is set by the combination logic of flash controller user configuration register CONFIG[26:24] after power on. 0 = 12MHz (or 6MHz) crystal oscillation is disabled 1 = 12MHz (or 6MHz) crystal oscillation is enabled



AHB Device Clock Enable Control Register (CLK_AHBCLK)

These register bits are used to enable/disable the clock source for AMBA, AHB (Advanced High-Performance Bus) blocks and peripherals.

Register	Offset	R/W	Description	Reset Value
CLK_AHBCLK	CLK_BA + 0x04	R/W	AHB Device Clock Enable Control Register	0x0000_0004

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					ISPCKEN	Reserved	

Bits	Description	
[31:3]	Reserved	Reserved.
[2]	ISPCKEN	Flash ISP Engine Clock Enable Control 0 = To disable the Flash ISP engine clock. 1 = To enable the Flash ISP engine clock.
[1:0]	Reserved	Reserved.



APB Device Clock Enable Control Register (CLK_APBCLK)

These register bits are used to enable/disable clocks for APB (Advanced Peripheral Bus) peripherals. To enable the clocks write '1' to the appropriate bit. To reduce power consumption and disable the peripheral, write '0' to the appropriate bit.

Register	Offset	R/W	Description	Reset Value
CLK_APBCLK	CLK_BA + 0x08	R/W	APB Device Clock Enable Control Register	0x0000_000x

31	30	29	28	27	26	25	24
Reserved	APU_EN	Reserved	ADC_EN	USBD_EN	Reserved		
23	22	21	20	19	18	17	16
Reserved			PWM_EN	Reserved			
15	14	13	12	11	10	9	8
Reserved		SPI1_EN	SPI0_EN	Reserved			
7	6	5	4	3	2	1	0
Reserved		TMRF_EN	TMR2_EN	TMR1_EN	TMR0_EN	RTC_EN	WDT_EN

Bits	Description
[31]	Reserved Reserved.
[30]	APU_EN APU Clock Enable Control 0 = To disable the APU engine clock. 1 = To enable the APU engine clock.
[29]	Reserved Reserved.
[28]	ADC_EN Audio Analog-Digital-Converter (ADC) Clock Enable Control 0 = Disable. 1 = Enable.
[27]	USBD_EN USB FS Device Controller Clock Enable Control 0 = Disable. 1 = Enable.
[26:21]	Reserved Reserved.
[20]	PWM_EN PWM Block Clock Enable Control 0=Disable. 1=Enable.
[19:14]	Reserved Reserved.
[13]	SPI1_EN SPI1 Clock Enable Control 0=Disable. 1=Enable.



[12]	SPI0_EN	SPI0 Clock Enable Control 0=Disable. 1=Enable.
[11:6]	Reserved	Reserved.
[5]	TMRF_EN	TimerF Clock Enable Control 0=Disable. 1=Enable.
[4]	TMR2_EN	Timer2 Clock Enable Control 0=Disable. 1=Enable.
[3]	TMR1_EN	Timer1 Clock Enable Control 0=Disable. 1=Enable.
[2]	TMR0_EN	Timer0 Clock Enable Control 0=Disable. 1=Enable.
[1]	RTC_EN	Real-Time-Clock APB Interface Clock Control This bit is used to control the RTC APB clock only. The RTC engine clock source is from the 32.768KHz crystal. 0=Disable. 1=Enable.
[0]	WDT_EN	Watchdog Clock Enable Control This bit is the protected bit. To program this bit needs an open lock sequence, write "59h", "16h", "88h" to register SYS_REGLCTL to un-lock this bit. Refer to the register SYS_REGLCTL at address SYS_BA+0x100. The default bit value is set according to the Flash Controller User Configuration Register CONFIG[31]. 0=Disable. 1=Enable.



Clock Source Select Control Register 0 (CLK_CLKSEL0)

These bits are protected bits. To program these bits needs an open lock sequence, write “59h”, “16h”, “88h” to register SYS_REGLCTL to un-lock these bits. Refer to the register SYS_REGLCTL at address SYS_BA+0x100.

Register	Offset	R/W	Description	Reset Value
CLK_CLKSEL0	CLK_BA + 0x10	R/W	Clock Source Select Control Register 0	0x0000_000x

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					HCLKSEL		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	HCLKSEL	<p>HCLK clock source select.</p> <p>000 = clock source from external 32KHz crystal clock 001 = clock source from external 12MHz (or 6MHz) crystal clock 010 = clock source from PLL1 output 100 = clock source from PLL2 output 111 = clock source from internal 24MHz oscillator clock Others = equivalent with “000”</p> <p>Note:</p> <ol style="list-style-type: none"> Before clock switch, the related clock sources (pre-select and new-select) must be turned on. The 3-bit default value is reloaded with the value of CFOSC (CONFIG[26:24]) in user configuration register in Flash controller by any reset. Therefore the default value is either 001b or 111b. Besides, before chip enters power down mode, HCLKSEL only can be either 001b or 111b.



Clock Source Select Control Register 1 (CLK_CLKSEL1)

Register	Offset	R/W	Description	Reset Value
CLK_CLKSEL1	CLK_BA + 0x14	R/W	Clock Source Select Control Register 1	0x0000_0000

31	30	29	28	27	26	25	24
Reserved		PWMSEL		Reserved			
23	22	21	20	19	18	17	16
Reserved	TMRFSEL			Reserved	TMR2SEL		
15	14	13	12	11	10	9	8
Reserved	TMR1SEL			Reserved	TMR0SEL		
7	6	5	4	3	2	1	0
Reserved				ADCSEL		WDTSEL	

Bits	Description	Description
[31:30]	Reserved	Reserved.
[29:28]	PWMSEL	PWM Timer Clock Source Select 00 = Clock source from HCLK 01 = Clock source from external 32KHz crystal clock 10 = Clock source from external 12MHz (or 6MHz) crystal clock 11 = Clock source from internal 24MHz oscillator clock
[27:23]	Reserved	Reserved.
[22:20]	TMRFSEL	TimerF Clock Source Select 000 = Clock source from external 32KHz crystal clock / 32. 001 = Clock source from external 32KHz crystal clock / (4x32). 010 = Clock source from external 12MHz (or 6MHz) crystal clock / 16384. 011 = Clock source from external 12MHz (or 6MHz) crystal clock / (4x16384). 110 = Clock source from 24MHz OSC clock / 32768. 111 = Clock source from 24MHz OSC clock / (4x32768). Others = Equivalent with "000".
[19]	Reserved	Reserved.
[18:16]	TMR2SEL	Timer2 Clock Source Select 000 = Clock source from HCLK 001 = Clock source from external 32KHz crystal clock 010 = Clock source from external 12MHz (or 6MHz) crystal clock 011 = Clock source from external trigger 1xx = Clock source from internal 24MHz oscillator clock
[15]	Reserved	Reserved.



[14:12]	TMR1SEL	Timer1 Clock Source Select 000 = Clock source from HCLK 001 = Clock source from external 32KHz crystal clock 010 = Clock source from external 12MHz (or 6MHz) crystal clock 011 = Clock source from external trigger 1xx = Clock source from internal 24MHz oscillator clock
[11]	Reserved	Reserved.
[10:8]	TMR0SEL	Timer0 Clock Source Select 000 = Clock source from HCLK 001 = Clock source from external 32KHz crystal clock 010 = Clock source from external 12MHz (or 6MHz) crystal clock 011 = Clock source from external trigger 1xx = Clock source from internal 24MHz oscillator clock
[7:4]	Reserved	Reserved.
[3:2]	ADCSEL	ADC Clock Source Select 00 = Clock source from PLL2 clock 01 = Clock source from external 12MHz (or 6MHz) crystal clock 1x = Clock source from internal 24MHz oscillator clock
[1:0]	WDTSEL	Watchdog Timer Clock Source Selection (Write Protect) These bits are protected bits. To program these bits needs an open lock sequence, write "59h", "16h", "88h" to SYS_REGLCTL to un-lock these bits. Refer to the register SYS_REGLCTL at address SYS_BA+0x100.. 00 = Clock source from HCLK/2048 clock 01 = Clock source from external 32KHz crystal clock 10 = Clock source from external 12MHz (or 6MHz) crystal clock 11 = Clock source from internal 24MHz oscillator clock



Clock Divider Register (CLK_CLKDIV)

Register	Offset	R/W	Description	Reset Value
CLK_CLKDIV	CLK_BA + 0x18	R/W	Clock Divider Number Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved				ADCDIV			
23	22	21	20	19	18	17	16
ADCDIV							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved				HCLKDIV			

Bits	Description
[31:28]	Reserved Reserved.
[23:16]	ADCDIV ADC Clock Divide Number From ADC Clock Source The ADC clock frequency $ADCLK = (\text{ADC clock source frequency}) / (\text{ADCDIV} + 1)$. The ADC clock must meet the constraint: $ADCLK \leq HCKL/2$
[15:4]	Reserved Reserved.
[3:0]	HCLKDIV HCLK Clock Divide Number From HCLK Clock Source The HCLK clock frequency = $(\text{HCLK clock source frequency}) / (\text{HCLKDIV} + 1)$.



PLL Control Register (CLK PLLCON)

Register	Offset	R/W	Description	Reset Value
CLK_PLLCON	CLK_BA + 0x20	R/W	PLL Control Register	0x0000_0007

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved				PLL_SRC		PLL1PD	PLL2PD

Bits	Descriptions	
[31:4]	Reserved	Reserved
[3:2]	PLL_SRC	PLL1 and PLL2 Input Source Clock Select. 00 = PLL1 output is direct from XIN/XOUT oscillator PLL2 output frequency is 8 times the output of PLL1. 01 = PLL1 output is 6MHz, its source clock is from 32KHz oscillator, PLL2 output frequency is 8 times the output of PLL1. 1x = PLL1 output is direct from XIN/XOUT oscillator, PLL2 output frequency is 4 times the output of PLL1.
[1]	PLL1PD	PLL1 Power Down Mode. 0 = PLL1 is in normal mode, 1 = PLL1 is in power-down mode.
[0]	PLL2PD	PLL2 Power Down Mode. 0 = PLL2 is in normal mode, 1 = PLL2 is in power-down mode.

5.4 SRAM

5.4.1 Overview

N572F065 equips with 8KB SRAM for program code and data storage. It's an AHB slave interface. It supports byte, half, and word read, and write access. An arbiter design enables Cortex-M0 CPU and DMA to access SRAM as the same without any data conflict.

5.4.2 Block Diagram

The block diagram of SRAM Controller with rotation engine is depicted as following:

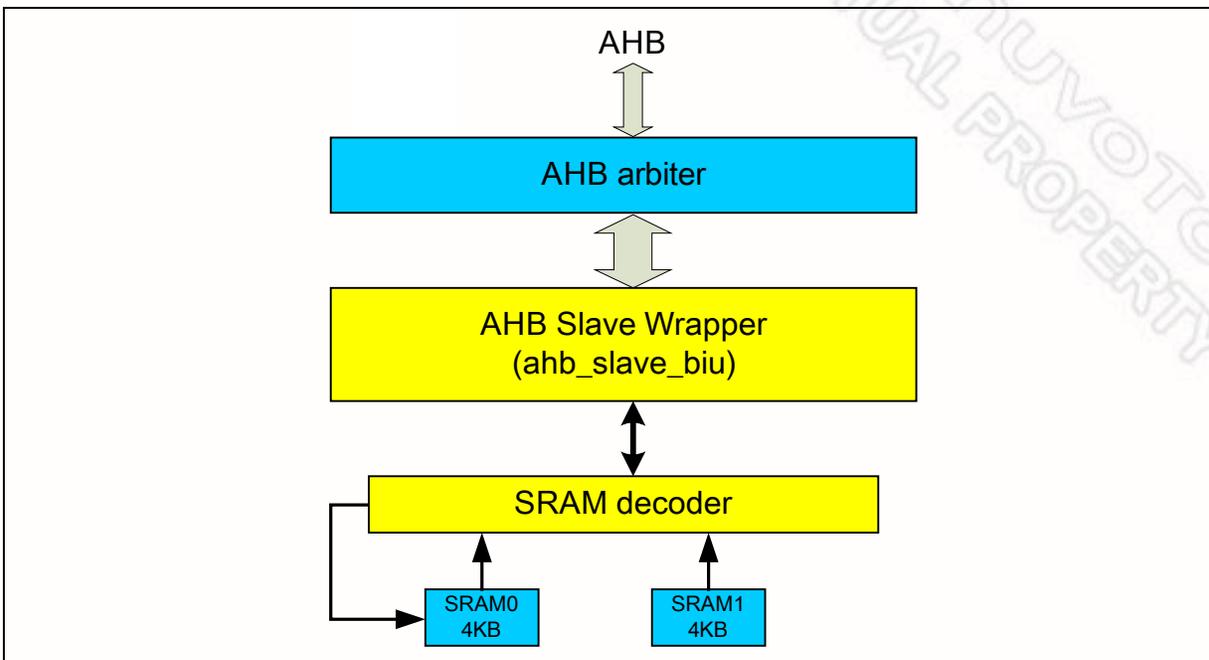


Figure 5-4 SRAM Controller Block Diagram

5.5 USB Device Controller (USB D)

5.5.1 Overview

There is one set of USB 2.0 full-speed device controller and transceiver in this device. It is compliant with USB 2.0 full-speed device specification and supports control/bulk/interrupt/ isochronous transfer types.

In this device controller, there are two main interfaces: the APB bus and USB bus which comes from the USB PHY transceiver. For the APB bus, the CPU can program control registers through it. There are 512 bytes internal SRAM as data buffer in this controller. For IN or OUT transfer, it is necessary to write data to SRAM or read data from SRAM through the APB interface or SIE. User needs to set the effective starting address of SRAM for each endpoint buffer through buffer segmentation register (USB D_BUFSEGx).

There are 8 endpoints in this controller. Each of the endpoint can be configured as IN or OUT endpoint. All the operations including Control, Bulk, Interrupt and Isochronous transfer are implemented in this block. The block of —Endpoint Controll is also used to manage the data sequential synchronization, endpoint states, current start address, transaction status, and data buffer status for each endpoint.

There are four different interrupt events in this controller. They are the no-event-wake-up, device plug-in or plug-out event, USB events, like IN ACK, OUT ACK etc, and BUS events, like suspend and resume, etc. Any event will cause an interrupt, and users just need to check the related event flags in interrupt event status register (USB D_INTSTS) to acknowledge what kind of interrupt occurring, and then check the related USB Endpoint Status Register (USB D_EPSTS) to acknowledge what kind of event occurring in this endpoint.

A software-disconnect function is also supported for this USB controller. It is used to simulate the disconnection of this device from the host. If user enables SE0 bit (USB D_SE0), the USB controller will force the output of USB_D+ and USB_D- to level low and its function is disabled. After disable the SE0 bit, host will enumerate the USB device again.

For more information on the Universal Serial Bus, please refer to *Universal Serial Bus Specification Revision 1.1*

5.5.2 Features

This Universal Serial Bus (USB) performs a single serial interface for attaching all USB peripherals to the host system. Following is the feature list of this USB.

- Conforming to USB2.0 Device
- Full Speed 12Mbps.
- Provide 1 interrupt vector with 4 interrupt events (NEVWK, VBUSDET, USB and BUS).
- Support Control, Bulk In/Out, Interrupt and Isochronous transfer type.
- Supports suspend function when no bus activity existing for 3 ms.
- Provide 6 sets configurable hardware for endpoints configuration for all Control/Bulk/ISO/Interrupt transfer type and max. 512 bytes endpoint size.
- Include 512 Bytes internal SRAM as USB buffer.
- Provide remote wakeup capability.

5.5.3 Block Diagram

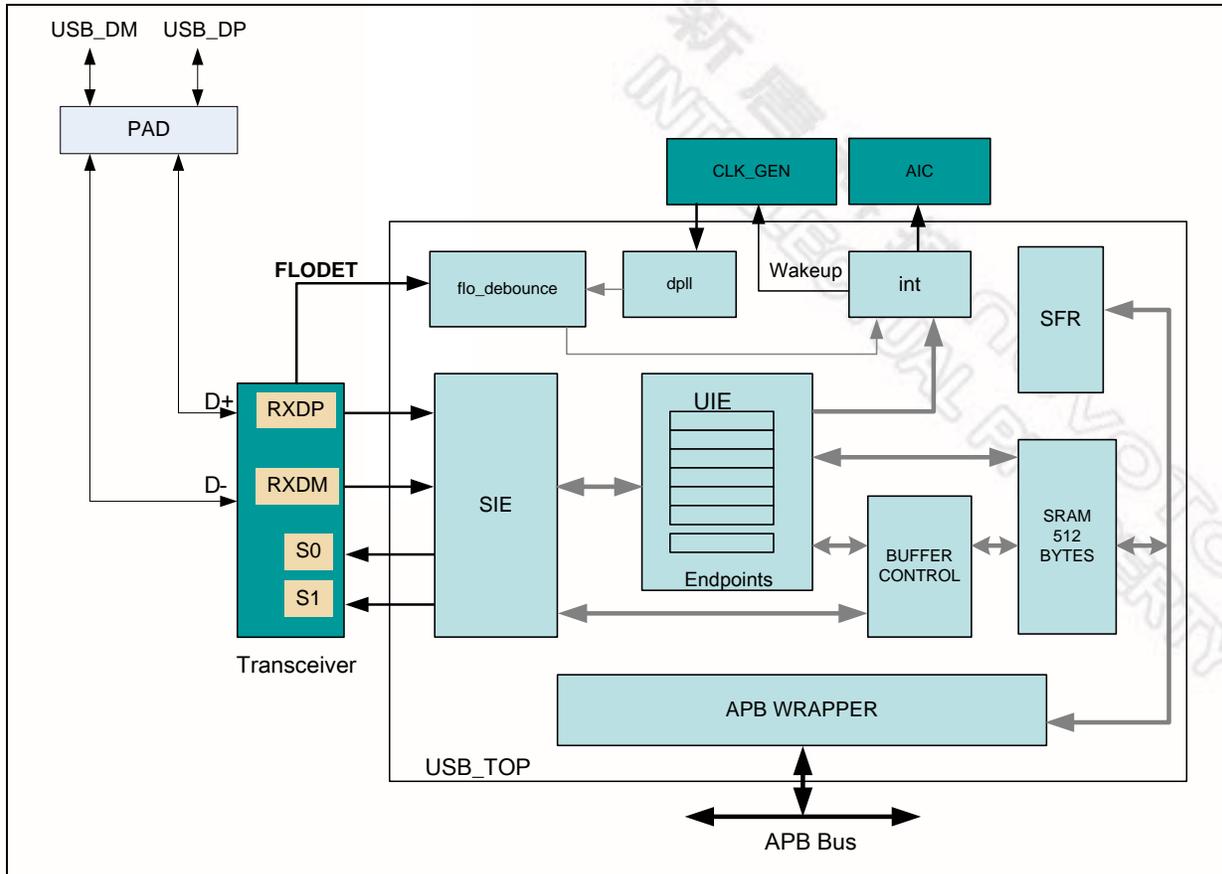


Figure 5-5 USB Block Diagram

5.5.4 Function Description

5.5.4.1 SIE (Serial Interface Engine)

The SIE is the front-end of the device controller and handles most of the USB packet protocol. The SIE typically comprehends signaling up to the transaction level. The functions that it handles could include:

- Packet recognition, transaction sequencing
- SOP, EOP, RESET, RESUME signal detection/generation
- Clock/Data separation
- NRZI Data encoding/decoding and bit-stuffing
- CRC generation and checking (for Token and Data)
- Packet ID (PID) generation and checking/ decoding
- Serial-Parallel/ Parallel-Serial conversion

5.5.4.2 UIE

The UIE is the endpoints management. All the operations include Control, Bulk, Interrupt and



Isochronous transfer are implemented in it.

5.5.4.3 Digital Phase Lock Loop

The bit rate of USB data is 12MHz. The DPLL use the 48MHz which comes from the clock controller to lock the input data RXDP and RXDM. The 12MHz bit rate clock is also converted from DPLL.

5.5.4.4 VBUS Detection De-bouncing

A USB device may be plugged-in or plugged-out from the USB host. To monitor the state of a USB device when it is detached from the USB host, the device controller provides hardware de-bouncing for USB VBUS detection interrupt to avoid bounce problems on USB plug-in or unplug. VBUS detection interrupt appears about 10 ms later than USB plug-in or plug-out. User can acknowledge USB plug-in/plug-out by reading USBD_VBUSDET register. The VBUSDET flag represents the current state on the bus without de-bouncing. If VBUSDET is 1, it means the USB cable is plugged-in. If user polls the flag to check USB state, software de-bouncing must be added if needed.

5.5.4.5 Interrupt control

This USB provides 1 interrupt vector with 4 interrupt events (NEVWK, VBUSDET, USB and BUS). The NEVWK event occurs after waking up the system from Power-down mode (The power mode function is defined in system power-down control register, CLK_PWRCTL). The VBUSDET event is used for USB plug-in or unplug. The USB event notifies users of some USB requests, such as IN ACK, OUT ACK., and the BUS event notifies users of some bus events, such as suspend and, resume. The related bits must be set in the interrupt enable register (USB_D_INTEN) of USB Device Controller to enable USB interrupts.

NEVWK interrupt is only presented when no the other USB interrupt events happened more than 20ms after the chip is waked up from Power-down mode. After the chip enters Power-down mode, any change on USB_VBUS, USB_D+ and USB_D- can wake up this chip if USB wake-up function is enabled. If this change is not intentionally, no interrupt but NEVWK interrupt will occur. After waking up by USB, this interrupt will occur when no the other USB interrupt events are presented for more than 20ms. The following figure is the control flow of wake-up interrupt.

The USB interrupt is used to notify users of any USB event on the bus, and user can read EPSTS (USB_D_EPSTS[31:8]) and EPEVT7~0 (USB_D_INTSTS[23:16]) to take necessary responses.

Same as USB interrupt, BUS interrupt notifies users of some bus events, like USB reset, suspend, time-out, and resume. A user can read USB_D_ATTR to acknowledge bus events.

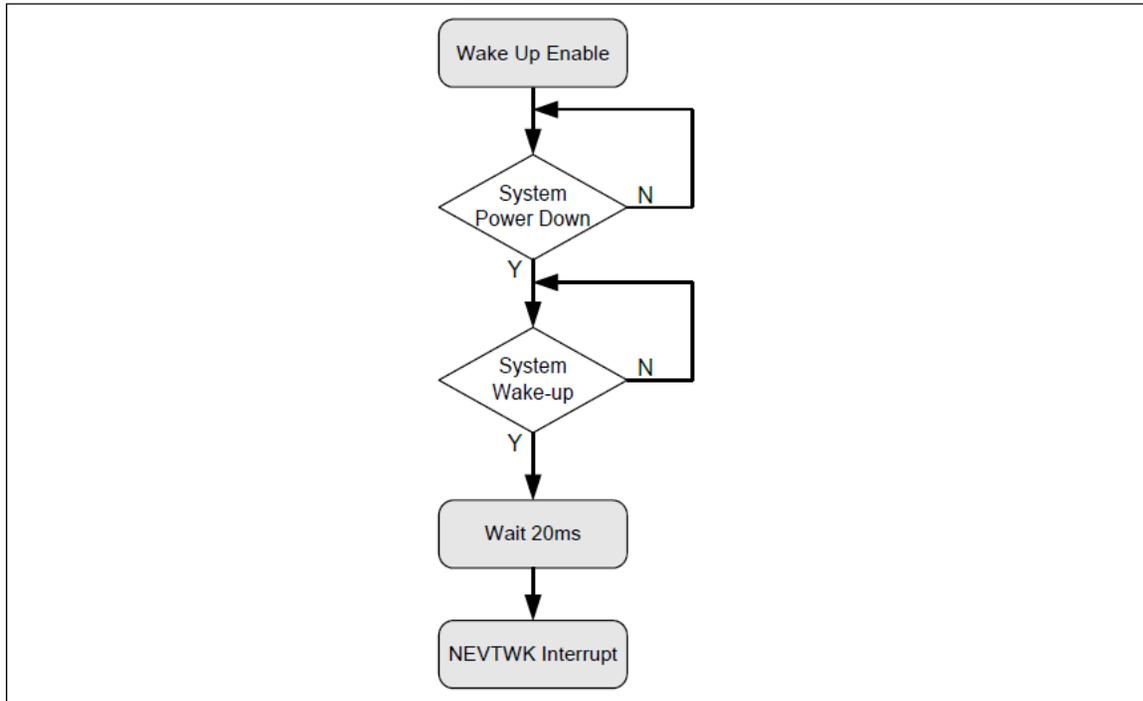


Figure 5-6 NEVWK Interrupt Operation Flow

5.5.4.6 Power Saving

User can write 0 to USBD_ATTR[4] to disable PHY under special circumstances, like suspend, to conserve power.

5.5.4.7 Buffer Control

There is 512 bytes SRAM in the controller and the 8 endpoints share this buffer. User shall configure each endpoint's effective starting address in the buffer segmentation register before the USB function active. The —Buffer Control block is used to control each endpoint's effective starting address and its SRAM size is defined in the USBD_MXPLDx register.

Figure 6.22-3 depicts the starting address for each endpoint according the content of USBD_BUFSEGx and USBD_MXPLDx registers. If the USBD_BUFSEG0 is programmed as 0x08h and USBD_MXPLD0 is set as 0x40h, the SRAM size of endpoint 0 is start from USBD_BA+0x108h and end in USBD_BA+0x148h. (**Note:** The USBD SRAM base is USBD_BA+0x100h).

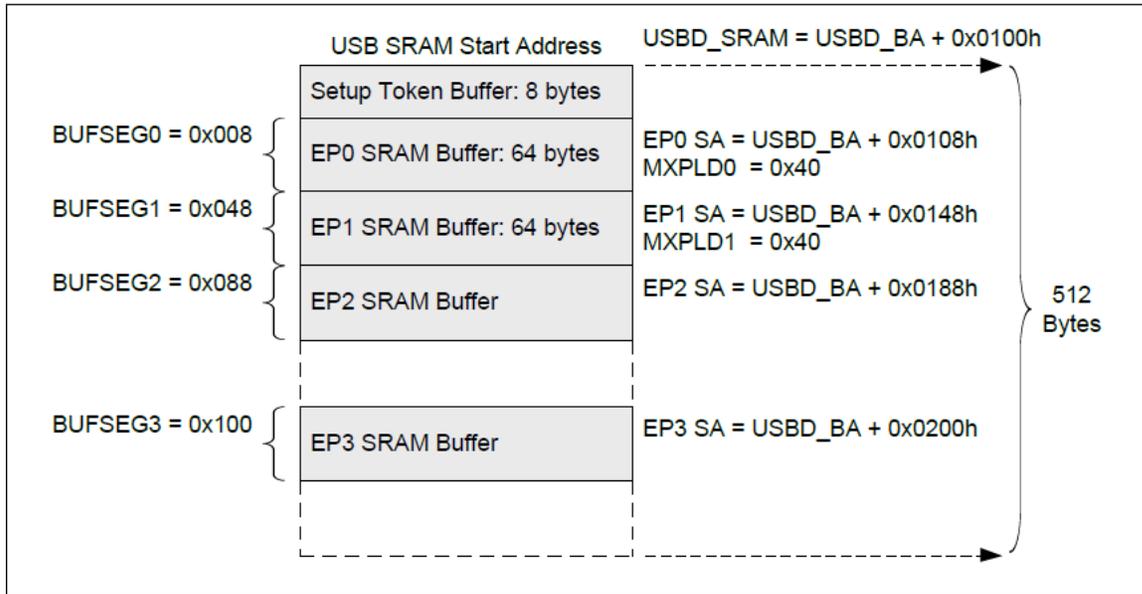


Figure 5-7 Endpoint SRAM Structure

5.5.5 Handling Transactions with USB Device Peripheral

User can use interrupt or polling USB_D_INTSTS to monitor the USB transactions. When transactions occur, USB_D_INTSTS will be set by hardware and send an interrupt request to CPU (if related interrupt enabled), or user can polling USB_D_INTSTS to get these events without interrupt. The following is the control flow with interrupt enabled.

When USB host has requested data from a device controller, user needs to prepare related data in the specified endpoint buffer in advance. After buffering the required data, user needs to write the actual data length in the specified USB_D_MXPLDx register. Once this register is written, the internal signal 'In_Rdy' will be asserted and the buffering data will be transmitted immediately after receiving associated IN token from Host. Note that after transferring the specified data, the signal 'In_Rdy' will de-assert automatically by hardware.

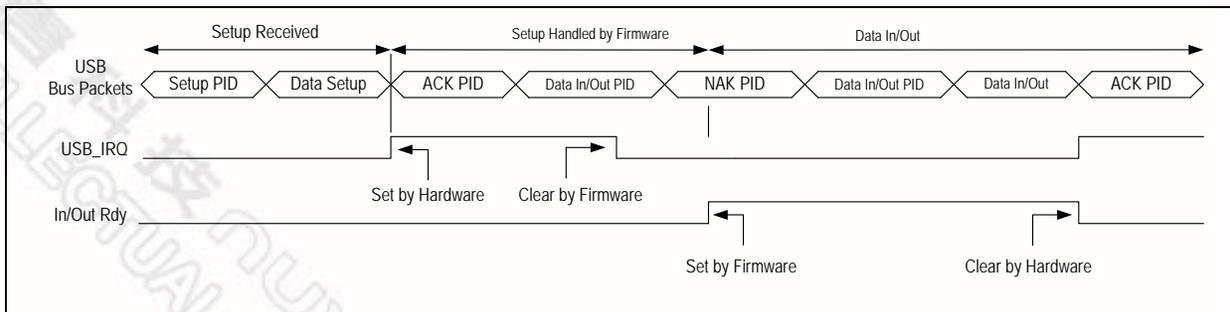


Figure 5-8 Setup Transaction followed by Data In/Out Transaction

Alternatively, when USB host wants to transmit data to the OUT endpoint in the device controller, hardware will buffer these data to the specified endpoint buffer. After this transaction is completed, hardware will record the data length in specified USB_D_MXPLDx register and de-assert the internal signal "Out_Rdy". This will avoid hardware accepting next transaction until user moves out the current

data in the related endpoint buffer. Once users have processed this transaction, the specified USBD_MXPLDx register needs to be written by firmware to assert the signal “Out_Rdy” again to accept the next transaction.

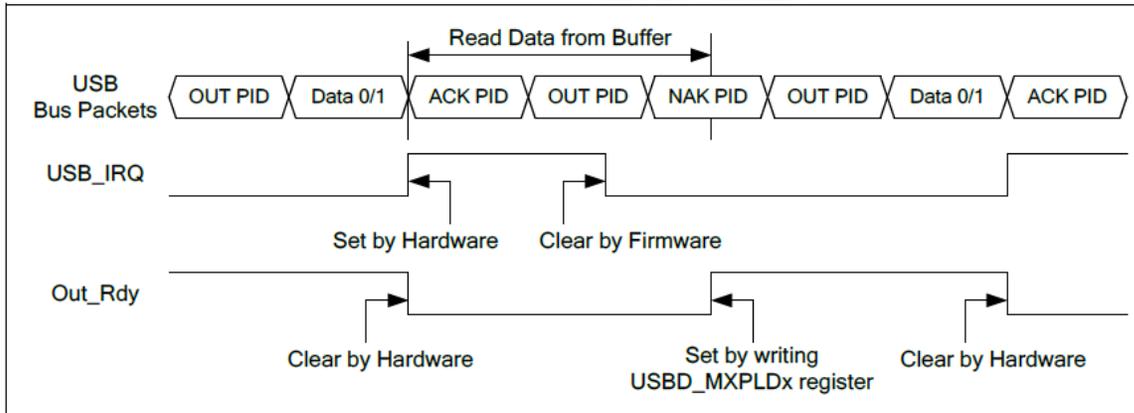


Figure 5-9 Data Out Transfer





5.5.6 Memory Mapping

Address	Size	Type	Description
4006_0000h ~ 4006_00FFh	256 Bytes	Register	Special function register
4006_0100h ~ 4006_02FFh	512 Bytes	SRAM	USB buffer

5.5.7 USB Control Registers Map

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
USB Base Address:				
USB_BA = 0x4006_0000				
USBD_INTEN	USBD_BA+0x000	R/W	USB Device Interrupt Enable Register	0x0000_0000
USBD_INTSTS	USBD_BA+0x004	R/W	USB Device Interrupt Event Status Register	0x0000_0000
USBD_FADDR	USBD_BA+0x008	R/W	USB Device Function Address Register	0x0000_0000
USBD_EPSTS	USBD_BA+0x00C	R	USB Device Endpoint Status Register	0x0000_0000
USBD_ATTR	USBD_BA+0x010	R/W	USB Device Bus Status and Attribution Register	0x0000_0040
USBD_VBUSDET	USBD_BA+0x014	R	USB Device VBUS Detection Register	0x0000_0000
USBD_STBUFSEG	USBD_BA+0x018	R/W	SETUP Token Buffer Segmentation Register	0x0000_0000
USBD_BUFSEG0	USBD_BA+0x020	R/W	Endpoint 0 Buffer Segmentation Register	0x0000_0000
USBD_MXPLD0	USBD_BA+0x024	R/W	Endpoint 0 Maximal Payload Register	0x0000_0000
USBD_CFG0	USBD_BA+0x028	R/W	Endpoint 0 Configuration Register	0x0000_0000
USBD_CFGP0	USBD_BA+0x02C	R/W	Endpoint 0 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_BUFSEG1	USBD_BA+0x030	R/W	Endpoint 1 Buffer Segmentation Register	0x0000_0000
USBD_MXPLD1	USBD_BA+0x034	R/W	Endpoint 1 Maximal Payload Register	0x0000_0000
USBD_CFG1	USBD_BA+0x038	R/W	Endpoint 1 Configuration Register	0x0000_0000
USBD_CFGP1	USBD_BA+0x03C	R/W	Endpoint 1 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_BUFSEG2	USBD_BA+0x040	R/W	Endpoint 2 Buffer Segmentation Register	0x0000_0000
USBD_MXPLD2	USBD_BA+0x044	R/W	Endpoint 2 Maximal Payload Register	0x0000_0000
USBD_CFG2	USBD_BA+0x048	R/W	Endpoint 2 Configuration Register	0x0000_0000
USBD_CFGP2	USBD_BA+0x04C	R/W	Endpoint 2 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_BUFSEG3	USBD_BA+0x050	R/W	Endpoint 3 Buffer Segmentation Register	0x0000_0000
USBD_MXPLD3	USBD_BA+0x054	R/W	Endpoint 3 Maximal Payload Register	0x0000_0000



USBD_CFG3	USBD_BA+0x058	R/W	Endpoint 3 Configuration Register	0x0000_0000
USBD_CFGP3	USBD_BA+0x05C	R/W	Endpoint 3 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_BUFSEG4	USBD_BA+0x060	R/W	Endpoint 4 Buffer Segmentation Register	0x0000_0000
USBD_MXPLD4	USBD_BA+0x064	R/W	Endpoint 4 Maximal Payload Register	0x0000_0000
USBD_CFG4	USBD_BA+0x068	R/W	Endpoint 4 Configuration Register	0x0000_0000
USBD_CFGP4	USBD_BA+0x06C	R/W	Endpoint 4 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_BUFSEG5	USBD_BA+0x070	R/W	Endpoint 5 Buffer Segmentation Register	0x0000_0000
USBD_MXPLD5	USBD_BA+0x074	R/W	Endpoint 5 Maximal Payload Register	0x0000_0000
USBD_CFG5	USBD_BA+0x078	R/W	Endpoint 5 Configuration Register	0x0000_0000
USBD_CFGP5	USBD_BA+0x07C	R/W	Endpoint 5 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_SE0	USBD_BA+0x090	R/W	USB Device Drive SE0 Control Register	0x0000_0001
USBD_BIST	USBD_BA+0x0A0	R/W	USB Buffer Self-test Register	0x0000_0000



5.5.8 USB Control Registers Description

USB Interrupt Enable Register (USB_D_INTEN)

Register	Offset	R/W	Description	Reset Value
USB_D_INTEN	USB_D_BA+0x000	R/W	USB Device Interrupt Enable Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
INNAKEN	Reserved						WKEN
7	6	5	4	3	2	1	0
Reserved				NEVWKIEN	VBDETIEN	USBIEN	BUSIEN

Bits	Description	
[31:16]	Reserved	Reserved
[15]	INNAKEN	<p>Active NAK Function And Its Status In IN Token</p> <p>0 = When device responds NAK after receiving IN token, IN NAK status will not be updated to USB_D_EPSTS register, so that the USB interrupt event will not be asserted.</p> <p>1 = IN NAK status will be updated to USB_D_EPSTS register and the USB interrupt event will be asserted, when the device responds NAK after receiving IN token.</p>
[14:9]	Reserved	Reserved
[8]	WKEN	<p>Wake-Up Function Enable Bit</p> <p>0 = USB wake-up function Disabled.</p> <p>1 = USB wake-up function Enabled.</p>
[7:4]	Reserved	Reserved
[3]	NEVWKIEN	<p>USB No-Event-Wake-Up Interrupt Enable Bit</p> <p>0 = No-event-wake-up Interrupt Disabled.</p> <p>1 = No-event-wake-up Interrupt Enabled.</p>
[2]	VBDETIEN	<p>VBUS Detection Interrupt Enable Bit</p> <p>0 = VBUS detection Interrupt Disabled.</p> <p>1 = VBUS detection Interrupt Enabled.</p>
[1]	USBIEN	<p>USB Event Interrupt Enable Bit</p> <p>0 = USB event interrupt Disabled.</p> <p>1 = USB event interrupt Enabled.</p>
[0]	BUSIEN	<p>Bus Event Interrupt Enable Bit</p> <p>0 = BUS event interrupt Disabled.</p> <p>1 = BUS event interrupt Enabled.</p>



USB Interrupt Event Status Register (USBD_INTSTS)

Register	Offset	R/W	Description	Reset Value
USBD_INTSTS	USBD_BA+0x004	R/W	USB Device Interrupt Event Status Register	0x0000_0000

31	30	29	28	27	26	25	24
SETUP	Reserved						
23	22	21	20	19	18	17	16
Reserved		EPEVT5	EPEVT4	EPEVT3	EPEVT2	EPEVT1	EPEVT0
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved				NEVWKIF	VBDTIF	USBIF	BUSIF

Bits	Description
[31]	<p>SETUP</p> <p>Setup Event Status 0 = No Setup event. 1 = Setup event occurred, cleared by write 1 to USBD_INTSTS[31].</p>
[30:22]	Reserved
[21]	<p>EPEVT5</p> <p>Endpoint 5's USB Event Status 0 = No event occurred in endpoint 5. 1 = USB event occurred on Endpoint 5, check USBD_EPSTS[25:23] to know which kind of USB event was occurred, cleared by write 1 to USBD_INTSTS[21] or USBD_INTSTS[1].</p>
[20]	<p>EPEVT4</p> <p>Endpoint 4's USB Event Status 0 = No event occurred in endpoint 4. 1 = USB event occurred on Endpoint 4, check USBD_EPSTS[22:20] to know which kind of USB event was occurred, cleared by write 1 to USBD_INTSTS[20] or USBD_INTSTS[1].</p>
[19]	<p>EPEVT3</p> <p>Endpoint 3's USB Event Status 0 = No event occurred in endpoint 3. 1 = USB event occurred on Endpoint 3, check USBD_EPSTS[19:17] to know which kind of USB event was occurred, cleared by write 1 to USBD_INTSTS[19] or USBD_INTSTS[1].</p>
[18]	<p>EPEVT2</p> <p>Endpoint 2's USB Event Status 0 = No event occurred in endpoint 2. 1 = USB event occurred on Endpoint 2, check USBD_EPSTS[16:14] to know which kind of USB event was occurred, cleared by write 1 to USBD_INTSTS[18] or USBD_INTSTS[1].</p>
[17]	<p>EPEVT1</p> <p>Endpoint 1's USB Event Status 0 = No event occurred in endpoint 1. 1 = USB event occurred on Endpoint 1, check USBD_EPSTS[13:11] to know which kind of USB event was occurred, cleared by write 1 to USBD_INTSTS[17] or USBD_INTSTS[1].</p>



[16]	EPEVTO	Endpoint 0's USB Event Status 0 = No event occurred in endpoint 0. 1 = USB event occurred on Endpoint 0, check USBD_EPSTS[10:8] to know which kind of USB event was occurred, cleared by write 1 to USBD_INTSTS[16] or USBD_INTSTS[1].
[15:4]	Reserved	Reserved
[3]	NEVWKIF	No-Event-Wake-Up Interrupt Status 0 = NEVWK event does not occur. 1 = No-event-wake-up event occurred, cleared by write 1 to USBD_INTSTS[3].
[2]	VBDETIF	VBUS Detection Interrupt Status 0 = There is not attached/detached event in the USB. 1 = There is attached/detached event in the USB bus and it is cleared by write 1 to USBD_INTSTS[2].
[1]	USBIF	USB Event Interrupt Status The USB event includes the SETUP Token, IN Token, OUT ACK, ISO IN, or ISO OUT events in the bus. 0 = No USB event occurred. 1 = USB event occurred, check EPSTS0-5[2:0] to know which kind of USB event was occurred, cleared by write 1 to USBD_INTSTS[1] or EPSTS0-7 and SETUP (USBD_INTSTS[31]).
[0]	BUSIF	BUS Interrupt Status The BUS event means that there is one of the suspense or the resume function in the bus. 0 = No BUS event occurred. 1 = Bus event occurred; check USBD_ATTR[3:0] to know which kind of bus event was occurred, cleared by write 1 to USBD_INTSTS[0].



USB Device Function Address Register (USBD_FADDR)

A seven-bit value uses as the address of a device on the USB BUS.

Register	Offset	R/W	Description	Reset Value
USBD_FADDR	USBD_BA+0x008	R/W	USB Device Function Address Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved	FADDR						

Bits	Description	
[31:7]	Reserved	Reserved
[6:0]	FADDR	USB Device Function Address



USB Endpoint Status Register (USB EPSTS)

Register	Offset	R/W	Description	Reset Value
USB_EPSTS	USB_BA+0x00C	R	USB Device Endpoint Status Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved						EPSTS5	
23	22	21	20	19	18	17	16
EPSTS5		EPSTS4			EPSTS3		EPSTS2
15	14	13	12	11	10	9	8
EPSTS2		EPSTS1			EPSTS0		
7	6	5	4	3	2	1	0
OV	STS			EPT			

Bits	Description	
[31:26]	Reserved	Reserved
[25:23]	EPSTS5	Endpoint 5 Status These bits are used to indicate the current status of this endpoint 000 = In ACK. 001 = In NAK. 010 = Out Packet Data0 ACK. 110 = Out Packet Data1 ACK. 011 = Setup ACK. 111 = Isochronous transfer end.
[22:20]	EPSTS4	Endpoint 4 Status These bits are used to indicate the current status of this endpoint 000 = In ACK. 001 = In NAK. 010 = Out Packet Data0 ACK. 110 = Out Packet Data1 ACK. 011 = Setup ACK. 111 = Isochronous transfer end.
[19:17]	EPSTS3	Endpoint 3 Status These bits are used to indicate the current status of this endpoint 000 = In ACK. 001 = In NAK. 010 = Out Packet Data0 ACK. 110 = Out Packet Data1 ACK. 011 = Setup ACK. 111 = Isochronous transfer end.

[16:14]	EPSTS2	Endpoint 2 Status These bits are used to indicate the current status of this endpoint 000 = In ACK. 001 = In NAK. 010 = Out Packet Data0 ACK. 110 = Out Packet Data1 ACK. 011 = Setup ACK. 111 = Isochronous transfer end.
[13:11]	EPSTS1	Endpoint 1 Status These bits are used to indicate the current status of this endpoint 000 = In ACK. 001 = In NAK. 010 = Out Packet Data0 ACK. 110 = Out Packet Data1 ACK. 011 = Setup ACK. 111 = Isochronous transfer end.
[10:8]	EPSTS0	Endpoint 0 Status These bits are used to indicate the current status of this endpoint 000 = In ACK. 001 = In NAK. 010 = Out Packet Data0 ACK. 110 = Out Packet Data1 ACK. 011 = Setup ACK. 111 = Isochronous transfer end.
[7]	OV	Overrun It indicates that the received data is over the maximum payload number or not. 0 = No overrun. 1 = Out Data is more than the Max Payload in MXPLD register or the Setup Data is more than 8 Bytes.
[6:4]	STS	000 = In ACK 001 = In NAK 010 = Out 0 ACK 110 = Out 1 ACK 011 = Setup ACK 111 = Isochronous translation end
[3:0]	EPT	Endpoint number



USB Bus Status and Attribution Register (USB_D_ATTR)

Register	Offset	R/W	Description	Reset Value
USB_D_ATTR	USB_D_BA+0x010	R/W	USB Device Bus Status and Attribution Register	0x0000_0040

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved					BYTEM	PDB	DPPUEN
7	6	5	4	3	2	1	0
USBEN	Reserved	RWAKEUP	PHYEN	TOUT	RESUME	SUSPEND	USB_RST

Bits	Description
[31:11]	Reserved
[10]	BYTEM CPU Access USB SRAM Size Mode Selection 0 = Word mode: The size of the transfer from CPU to USB SRAM can be Word only. 1 = Byte mode: The size of the transfer from CPU to USB SRAM can be Byte only.
[9]	PDB Power Down USB-IP Related Power and Control 0 = Enable power down. 1 = Disable power down.
[8]	DPPUEN Pull-Up Resistor On USB_DP Enable Bit 0 = Pull-up resistor in USB_D+ bus Disabled. 1 = Pull-up resistor in USB_D+ bus Active.
[7]	USBEN USB Controller Enable Bit 0 = USB Controller Disabled. 1 = USB Controller Enabled.
[6]	Reserved
[5]	RWAKEUP Remote Wake-Up 0 = Release the USB bus from K state. 1 = Force USB bus to K (USB_D+ low, USB_D-: high) state, used for remote wake-up.
[4]	PHYEN PHY Transceiver Function Enable Bit 0 = PHY transceiver function Disabled. The PHY means USB transceiver output. 1 = PHY transceiver function Enabled.
[3]	TOUT Time-Out Status 0 = No time-out. 1 = No Bus response more than 18 bits time. Note: This bit is read only.

[2]	RESUME	<p>Resume Status</p> <p>0 = No bus resume. 1 = Resume from suspend.</p> <p>Note: This bit is read only.</p>
[1]	SUSPEND	<p>Suspend Status</p> <p>0 = Bus no suspend. 1 = Bus idle more than 3ms, either cable is plugged off or host is sleeping.</p> <p>Note: This bit is read only.</p>
[0]	USBRST	<p>USB Reset Status</p> <p>0 = Bus no reset. 1 = Bus reset when SE0 (single-ended 0) more than 2.5us.</p> <p>Note: This bit is read only.</p>


USB Device VBUS Detection Register (USB_D_VBUSDET)

Register	Offset	R/W	Description	Reset Value
USB_D_VBUSDET	USB_D_BA+0x014	R	USB Device VBUS Detection Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							VBUSDET

Bits	Description	
[31:1]	Reserved	Reserved
[0]	VBUSDET	Device VBUS Detection 0 = Controller is not attached to the USB host. 1 = Controller is attached to the USB host. Note: This bit is read only.


USB SETUP Token Buffer Segmentation Register (USB_D_STBUFSEG)

Register	Offset	R/W	Description	Reset Value
USB_D_STBUFSEG	USB_D_BA+0x018	R/W	SETUP Token Buffer Segmentation Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							STBUFSEG[8]
7	6	5	4	3	2	1	0
STBUFSEG[7:3]					Reserved		

Bits	Description	
[31:9]	Reserved	Reserved
[8:3]	STBUFSEG	<p>SETUP Token Buffer Segmentation</p> <p>It is used to indicate the offset address for the SETUP token with the USB Device SRAM starting address. The effective starting address is USB_D_SRAM address + {STBUFSEG[8:3], 3'b000}</p> <p>Where the USB_D_SRAM address = USB_D_BA+0x100h.</p> <p>Note: It is used for SETUP token only.</p>
[2:0]	Reserved	Reserved



USB Buffer Segmentation Register (USB BUFSEGx)

Register	Offset	R/W	Description	Reset Value
USBD_BUFSEG0	USBD_BA+0x020	R/W	Endpoint 0 Buffer Segmentation Register	0x0000_0000
USBD_BUFSEG1	USBD_BA+0x030	R/W	Endpoint 1 Buffer Segmentation Register	0x0000_0000
USBD_BUFSEG2	USBD_BA+0x040	R/W	Endpoint 2 Buffer Segmentation Register	0x0000_0000
USBD_BUFSEG3	USBD_BA+0x050	R/W	Endpoint 3 Buffer Segmentation Register	0x0000_0000
USBD_BUFSEG4	USBD_BA+0x060	R/W	Endpoint 4 Buffer Segmentation Register	0x0000_0000
USBD_BUFSEG5	USBD_BA+0x070	R/W	Endpoint 5 Buffer Segmentation Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							BUFSEG[8]
7	6	5	4	3	2	1	0
BUFSEG[7:3]					Reserved		

Bits	Description	
[31:9]	Reserved	Reserved
[8:3]	BUFSEG	<p>Endpoint Buffer Segmentation</p> <p>It is used to indicate the offset address for each endpoint with the USB SRAM starting address. The effective starting address of the endpoint is</p> <p>USBD_SRAM address + { BUFSEG[8:3], 3'b000}</p> <p>Where the USBD_SRAM address = USBD_BA+0x100h.</p> <p>Refer to the section 5.5.4.7 for the endpoint SRAM structure and its description.</p>
[2:0]	Reserved	Reserved



USB Maximal Payload Register (USB_MXPLDx)

Register	Offset	R/W	Description	Reset Value
USBD_MXPLD0	USBD_BA+0x024	R/W	Endpoint 0 Maximal Payload Register	0x0000_0000
USBD_MXPLD1	USBD_BA+0x034	R/W	Endpoint 1 Maximal Payload Register	0x0000_0000
USBD_MXPLD2	USBD_BA+0x044	R/W	Endpoint 2 Maximal Payload Register	0x0000_0000
USBD_MXPLD3	USBD_BA+0x054	R/W	Endpoint 3 Maximal Payload Register	0x0000_0000
USBD_MXPLD4	USBD_BA+0x064	R/W	Endpoint 4 Maximal Payload Register	0x0000_0000
USBD_MXPLD5	USBD_BA+0x074	R/W	Endpoint 5 Maximal Payload Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							MXPLD[8]
7	6	5	4	3	2	1	0
MXPLD[7:0]							

Bits	Description	
[31:9]	Reserved	Reserved
[8:0]	MXPLD	<p>Maximal Payload</p> <p>Define the data length which is transmitted to host (IN token) or the actual data length which is received from the host (OUT token). It also used to indicate that the endpoint is ready to be transmitted in IN token or received in OUT token.</p> <p>(1) When the register is written by CPU, For IN token, the value of MXPLD is used to define the data length to be transmitted and indicate the data buffer is ready.</p> <p>For OUT token, it means that the controller is ready to receive data from the host and the value of MXPLD is the maximal data length comes from host.</p> <p>(2) When the register is read by CPU, For IN token, the value of MXPLD is indicated by the data length be transmitted to host</p> <p>For OUT token, the value of MXPLD is indicated the actual data length receiving from host.</p> <p>Note: Once MXPLD is written, the data packets will be transmitted/received immediately after IN/OUT token arrived.</p>



USB Configuration Register (USB CFGx)

Register	Offset	R/W	Description	Reset Value
USBD_CFG0	USBD_BA+0x028	R/W	Endpoint 0 Configuration Register	0x0000_0000
USBD_CFG1	USBD_BA+0x038	R/W	Endpoint 1 Configuration Register	0x0000_0000
USBD_CFG2	USBD_BA+0x048	R/W	Endpoint 2 Configuration Register	0x0000_0000
USBD_CFG3	USBD_BA+0x058	R/W	Endpoint 3 Configuration Register	0x0000_0000
USBD_CFG4	USBD_BA+0x068	R/W	Endpoint 4 Configuration Register	0x0000_0000
USBD_CFG5	USBD_BA+0x078	R/W	Endpoint 5 Configuration Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved						stall_ctl	Reserved
7	6	5	4	3	2	1	0
DSQ	state		ISOCH	EPT			

Bits	Description
[31:10]	Reserved Reserved
[9]	CSTALL Clear STALL Response 0 = Disable the device to clear the STALL handshake in setup stage. 1 = Clear the device to response STALL handshake in setup stage.
[8]	Reserved Reserved
[7]	DSQSYNC Data Sequence Synchronization 0 = DATA0 PID. 1 = DATA1 PID. Note: It is used to specify the DATA0 or DATA1 PID in the following IN token transaction. hardware will toggle automatically in IN token base on the bit.
[6:5]	STATE Endpoint STATE 00 = Endpoint is Disabled. 01 = Out endpoint. 10 = IN endpoint. 11 = Undefined.
[4]	ISOCH Isochronous Endpoint This bit is used to set the endpoint as Isochronous endpoint, no handshake. 0 = No Isochronous endpoint. 1 = Isochronous endpoint.



[3:0]	EPNUM	Endpoint Number These bits are used to define the endpoint number of the current endpoint
-------	-------	---

新層科技 NUVOTON
INTELLECTUAL PROPERTY

新層科技 NUVOTON
INTELLECTUAL PROPERTY



USB Extra Configuration Register (USB CFGPx)

Register	Offset	R/W	Description	Reset Value
USBD_CFGP0	USBD_BA+0x02C	R/W	Endpoint 0 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_CFGP1	USBD_BA+0x03C	R/W	Endpoint 1 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_CFGP2	USBD_BA+0x04C	R/W	Endpoint 2 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_CFGP3	USBD_BA+0x05C	R/W	Endpoint 3 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_CFGP4	USBD_BA+0x06C	R/W	Endpoint 4 Set Stall and Clear In/Out Ready Control Register	0x0000_0000
USBD_CFGP5	USBD_BA+0x07C	R/W	Endpoint 5 Set Stall and Clear In/Out Ready Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved						SSTALL	CLRRDY

Bits	Description	
[31:2]	Reserved	Reserved
[1]	SSTALL	Set STALL 0 = Disable the device to response STALL. 1 = Set the device to respond STALL automatically.
[0]	CLRRDY	Clear Ready When the USBD_MXPLDx register is set by user, it means that the endpoint is ready to transmit or receive data. If the user wants to disable this transaction before the transaction start, users can set this bit to 1 to disable it and it is auto clear to 0. For IN token, write '1' to clear the IN token had ready to transmit the data to USB. For OUT token, write '1' to clear the OUT token had ready to receive the data from USB. This bit is written only and is always 0 when it is read back.



USB Drive SE0 Register (USB_D_SE0)

Register	Offset	R/W	Description	Reset Value
USB_D_SE0	USB_D_BA+0x090	R/W	USB Device Drive SE0 Control Register	0x0000_0001

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							SE0

Bits	Description	
[31:1]	Reserved	Reserved
[0]	SE0	<p>Drive Single Ended Zero In USB Bus</p> <p>The Single Ended Zero (SE0) is when both lines (USB_D+ and USB_D-) are being pulled low.</p> <p>0 = Normal operation.</p> <p>1 = Force USB PHY transceiver to drive SE0.</p>



USB Buffer Self-test Register (USB BIST)

Register	Offset	R/W	Description	Reset Value
USB_BIST	USB_BA+0x0A0	R/W	USB Buffer Self-test Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					BISTFAIL	FINISH	BISTEN

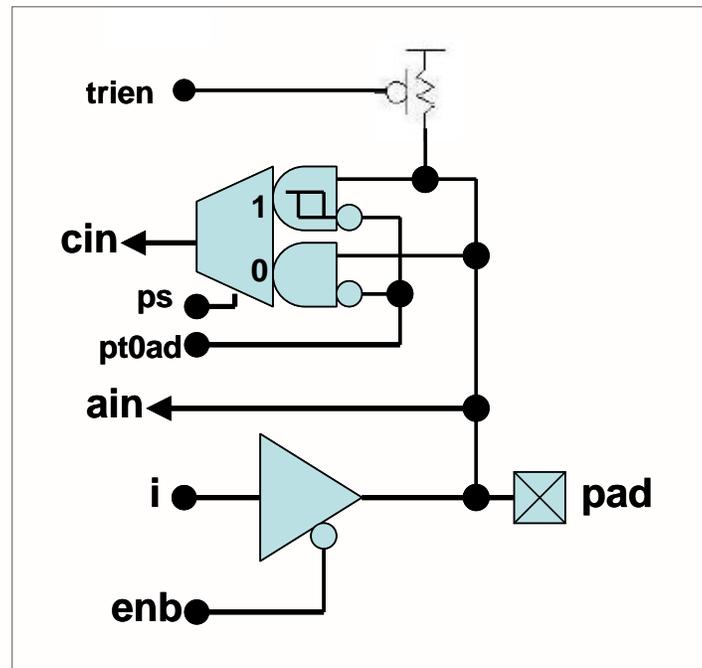
Bits	Description	
[31:3]	Reserved	Reserved
[2]	BISTFAIL	<p>BIST Fail</p> <p>The BISTFAIL indicates if the BIST test fails or succeeds. If the BISTFAIL is low at the end, the embedded SRAM pass the BIST test, otherwise, it is faulty. The BISTFAIL will be high once the BIST detects the error and remains high during the BIST operation.</p> <p>The BISTFAIL is a write clear field. Write 1 to this field clears the content and write 0 has no effect.</p> <p>Note: This bit is read only.</p>
[1]	FINISH	<p>BIST Operation Finish</p> <p>It indicates the end of the BIST operation. When BIST controller finishes all operations, this bit will be set high.</p> <p>This bit is a write clear field. Write 1 to this field clears the content and write 0 has no effect.</p> <p>Note: This bit is read only.</p>
[0]	BISTEN	<p>BIST mode enable</p> <p>0 = BIST is disabled or completed (automatically cleared by BIST controller).</p> <p>1 = BIST is enabled; begin to perform BIST on selected memory group.</p>

5.6 General Purpose I/O

5.6.1 Overview and Features

There are 32 pins of General Purpose I/O shared with special feature functions. These pins are arranged in 2 groups, PA and PB. Each pin of the 32 pins is independent and has the corresponding register bits to control the pin mode function and data.

The I/O type of each of I/O pins can be independently software configured as input, output, open-drain or quasi-bidirectional mode. In input mode, the I/O pin is in tri-state (high impedance) without output drive capability. In output mode, the I/O pin supports digital output function with source/sink current capability. In open-drain mode, the I/O pin supports digital output function but only with sink current capability, an additional pull-up resistor is needed for driving high state. In quasi-mode, the I/O pin supports digital output and input function at the same time but the source current is only up to hundreds μA .



5.6.1.1 Open-Drain mode explanation

For $Px_MODEn = 10b$ the GPIO $Px[n]$ pin is in Open-Drain mode.

If the bit value in the corresponding bit $Px_DOUT[n]$ is "0", the pin drive a "low" output on the pin.

If the bit value in the corresponding bit $Px_DOUT[n]$ is "1", the pin output drive is disabled and the pin status is controlled by external.

5.6.1.2 Quasi-bidirectional Mode Explanation

For $Px_MODEn = 11b$ the GPIO $Px[n]$ pin is in Quasi-bidirectional mode.

If the bit value in the corresponding bit $Px_DOUT[n]$ is "0", the pin drive a "low" output on the pin.

If the bit value in the corresponding bit Px_DOUT[n] is "1", the pin will check the pin value. If pin value is high, no action takes. If pin state is low, it will drive 2 clock cycles "high" and then disable the output drive and then the pin status is controlled by internal pull-high MOS.

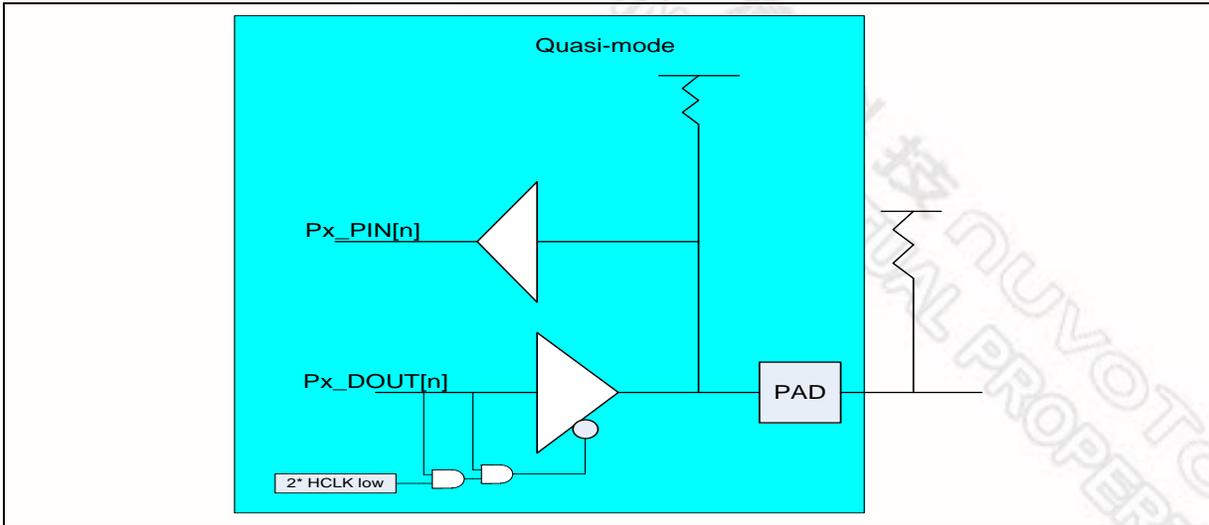


Figure 5-10 Quasi-bidirectional GPIO Mode



5.6.2 GPIO Control Register Map

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
GPIO Base Address: GPIO_BA = 0x5000_4000				
PA_MODE	GPIO_BA+0x000	R/W	GPIO PA Pin I/O Mode Control	0xFFFF_FFFF
PA_DINOFF	GPIO_BA+0x004	R/W	GPIO PA Digital Input Path Disable Control	0x0000_0000
PA_DOUT	GPIO_BA+0x008	R/W	GPIO PA Data Output Value	0x0000_FFFF
PA_DATMSK	GPIO_BA+0x00C	R/W	GPIO PA Data Output Write Mask	0x0000_0000
PA_PIN	GPIO_BA+0x010	R	GPIO PA Pin Value	0x0000_XXXX
PA_INTTYPE	GPIO_BA+0x018	R/W	GPIO PA Interrupt Trigger Type	0x0000_0000
PA_INTEN	GPIO_BA+0x01C	R/W	GPIO PA Interrupt Enable	0x0000_0000
PA_INTSRC	GPIO_BA+0x020	R/W	GPIO PA Interrupt Source Flag	0x0000_0000
PB_MODE	GPIO_BA+0x040	R/W	GPIO PB Pin I/O Mode Control	0xFFFF_FFFF
PB_DINOFF	GPIO_BA+0x044	R/W	GPIO PB Digital Input Path Disable Control	0x0000_0000
PB_DOUT	GPIO_BA+0x048	R/W	GPIO PB Data Output Value	0x0000_FFFF
PB_DATMSK	GPIO_BA+0x04C	R/W	GPIO PB Data Output Write Mask	0x0000_0000
PB_PIN	GPIO_BA+0x050	R	GPIO PB Pin Value	0x0000_XXXX
PB_INTTYPE	GPIO_BA+0x058	R/W	GPIO PB Interrupt Trigger Type	0x0000_0000
PB_INTEN	GPIO_BA+0x05C	R/W	GPIO PB Interrupt Enable	0x0000_0000
PB_INTSRC	GPIO_BA+0x060	R/W	GPIO PB Interrupt Source Flag	0x0000_0000



5.6.3 GPIO Control Register Description

GPIO Port [A/B] I/O Mode Control (Px_MODE)

Register	Offset	R/W	Description	Reset Value
PA_MODE	GPIO_BA+0x000	R/W	GPIO PA Pin I/O Mode Control	0xFFFF_FFFF
PB_MODE	GPIO_BA+0x040	R/W	GPIO PB Pin I/O Mode Control	0xFFFF_FFFF

31	30	29	28	27	26	25	24
MODE15		MODE14		MODE13		MODE12	
23	22	21	20	19	18	17	16
MODE11		MODE10		MODE9		MODE8	
15	14	13	12	11	10	9	8
MODE7		MODE6		MODE5		MODE4	
7	6	5	4	3	2	1	0
MODE3		MODE2		MODE1		MODE0	

Bits	Description
[2n+1 :2n] n=0,1..15	<p>MODEn</p> <p>Port [A/B] Pin[N] I/O Mode Control Determine each I/O type of GPIO Px pins 00 = GPIO Px[n] pin is in INPUT mode. 01 = GPIO Px[n] pin is in OUTPUT mode. 10 = GPIO Px[n] pin is in Open-Drain mode. 11 = GPIO Px[n] pin is in Quasi-bidirectional mode.</p>

GPIO Port [A/B] Digital Input Path Disable (Px_DINOFF)

Register	Offset	R/W	Description	Reset Value
PA_DINOFF	GPIO_BA+0x004	R/W	GPIO PA Digital Input Path Disable Control	0x0000_0000
PB_DINOFF	GPIO_BA+0x044	R/W	GPIO PB Digital Input Path Disable Control	0x0000_0000

31	30	29	28	27	26	25	24
DINOFF [15:8]							
23	22	21	20	19	18	17	16
DINOFF [7:0]							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							

Bits	Description	
[n+16] n=0,1..15	DINOFF[n]	Port [A/B] Pin[N] Digital Input Path Disable Control 0 = Px.n Digital input path Enable (Default). 1 = Px.n Digital input path Disable (digital input tied to low).
[15:0]	Reserved	Reserved.



GPIO Port [A/B] Data Output Value (Px DOUT)

Register	Offset	R/W	Description	Reset Value
PA_DOUT	GPIO_BA+0x008	R/W	GPIO PA Data Output Value	0x0000_FFFF
PB_DOUT	GPIO_BA+0x048	R/W	GPIO PB Data Output Value	0x0000_FFFF

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
DOUT[15:8]							
7	6	5	4	3	2	1	0
DOUT[7:0]							

Bits	Description
[15:0]	<p>DOUT</p> <p>Port [A/B] Pin[N] Output Value</p> <p>Each of these bits controls the status of a GPIO pin when the GPIO pin is configured as output, open-drain or quasi-bidirectional mode.</p> <p>0 = GPIO port [A/B] Pin[n] will drive Low if the corresponding output mode bit is set.</p> <p>1 = GPIO port [A/B] Pin[n] will drive High if the corresponding output mode bit is set.</p>

GPIO Port [A/B] Data Output Write Mask (Px_DATMSK)

Register	Offset	R/W	Description	Reset Value
PA_DATMSK	GPIO_BA+0x00C	R/W	GPIO PA Data Output Write Mask	0x0000_0000
PB_DATMSK	GPIO_BA+0x04C	R/W	GPIO PB Data Output Write Mask	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
DATMSK[15:8]							
7	6	5	4	3	2	1	0
DATMSK[7:0]							

Bits	Description
[15:0]	<p>Port [A/B] Pin[N] Data Output Write Mask</p> <p>These bits are used to protect the corresponding register of Px_DOUT[n]. When set the DATMSK[n] to "1", the corresponding Px_DOUT[n] bit is writing protected.</p> <p>0 = The corresponding Px_DOUT[n] bit can be updated.</p> <p>1 = The corresponding Px_DOUT[n] bit is read only.</p>



GPIO Port [A/B] Pin Value (Px PIN)

Register	Offset	R/W	Description	Reset Value
PA_PIN	GPIO_BA+0x010	R	GPIO PA Pin Value	0x0000_XXXX
PB_PIN	GPIO_BA+0x050	R	GPIO PB Pin Value	0x0000_XXXX

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
PIN[15:8]							
7	6	5	4	3	2	1	0
PIN[7:0]							

Bits	Description
[15:0]	<p>PIN[n]</p> <p>Port [A/B] Pin[N] Pin Values Each bit of the register reflects the actual status of the respective Px.n pin. If the bit is 1, it indicates the corresponding pin status is high; else the pin status is low.</p>

GPIO Port [A/B] Interrupt Mode Control (Px_INTTYPE)

Register	Offset	R/W	Description	Reset Value
PA_INTTYPE	GPIO_BA+0x018	R/W	GPIO PA Interrupt Trigger Type	0x0000_0000
PB_INTTYPE	GPIO_BA+0x058	R/W	GPIO PB Interrupt Trigger Type	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
TYPE[15:8]							
7	6	5	4	3	2	1	0
TYPE[7:0]							

Bits	Description
[15:0]	<p>Port [A/B] Pin[N] Edge Or Level Detection Interrupt Trigger Type Control</p> <p>TYPE[n] is used to control whether the interrupt mode is level triggered or edge triggered. If the interrupt mode is level triggered, the input source is sampled by one HCLK clock to generate the interrupt</p> <p>0 = Edge triggered interrupt. 1 = Level triggered interrupt.</p> <p>Note: If level triggered interrupt is selected, then only one level can be selected in the Px_INTEN register. If both levels are set, the setting is ignored and no interrupt will occur</p>



GPIO Port [A/B] Interrupt Enable Control (Px_INTEN)

Register	Offset	R/W	Description	Reset Value
PA_INTEN	GPIO_BA+0x01C	R/W	GPIO PA Interrupt Enable	0x0000_0000
PB_INTEN	GPIO_BA+0x05C	R/W	GPIO PB Interrupt Enable	0x0000_0000

31	30	29	28	27	26	25	24
RHIE[15:8]							
23	22	21	20	19	18	17	16
RHIE[7:0]							
15	14	13	12	11	10	9	8
FLIE[15:8]							
7	6	5	4	3	2	1	0
FLIE[7:0]							

Bits	Description
[n+16] n=0,1..15	<p>Port [A/B] Interrupt Enable By Input Rising Edge Or Input Level High</p> <p>RHIE[n] is used to enable the rising/high-level interrupt for each of the corresponding input Px.n pin. To set "1" also enables the pin wake-up function.</p> <p>When setting the RHIE[n] (Px_INTEN[n+16]) bit to 1 :</p> <p>If the interrupt is configured as level trigger mode (TYPE[n] is set to 1), one interrupt will occur while the input Px.n state is at high level.</p> <p>If the interrupt is configured as edge trigger mode (TYPE[n] is set to 0), one interrupt will occur while the input Px.n state changes from low to high.</p> <p>0 = Disable Px.n for low-to-high or level-high interrupt. 1 = Enable Px.n for low-to-high or level-high interrupt.</p>
[n] n=0,1..15	<p>Port [A/B] Interrupt Enable By Input Falling Edge Or Input Level Low</p> <p>FLIE[n] is used to enable the falling/low-level interrupt for each of the corresponding input Px.n pin. To set "1" also enables the pin wake-up function.</p> <p>When setting the FLIE[n] (Px_INTEN[n]) bit to 1 :</p> <p>If the interrupt is configured as level trigger mode (TYPE[n] is set to 1), one interrupt will occur while the input Px.n state is at low level.</p> <p>If the interrupt is configured as edge trigger mode (TYPE[n] is set to 0), one interrupt will occur while the input Px.n state changes from high to low.</p> <p>0 = Disable Px.n for low-level or high-to-low interrupt. 1 = Enable Px.n for low-level or high-to-low interrupt.</p>



GPIO Port [A/B] Interrupt Source Flag(Px_INTSRC)

Register	Offset	R/W	Description	Reset Value
PA_INTSRC	GPIO_BA+0x020	R/W	GPIO PA Interrupt Source Flag	0x0000_0000
PB_INTSRC	GPIO_BA+0x060	R/W	GPIO PB Interrupt Source Flag	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
INTSRC[15:8]							
7	6	5	4	3	2	1	0
INTSRC[7:0]							

Bits	Description
[15:0]	<p>Port [A/B] Interrupt Source Flag</p> <p>Read operation: 0 = No interrupt from Px.n. 1 = Px.n generated an interrupt.</p> <p>Write operation: 0 = No action. 1 = Clear the corresponding pending interrupt.</p>

5.7 PWM Generator and Capture Timer

5.7.1 Introduction

The N572F065 has 1 PWM timer. The PWM timer has 1 prescaler, 1 clock divider, 5 clock selectors, one 16-bit counter, four 16-bit comparators, and two Dead-Zone generators.

Clock divider provides PWM timer with 5 clock sources (1, 1/2, 1/4, 1/8, 1/16). PWM timer receives its own clock signal from clock divider which receives clock from 8-bit prescaler. The 16-bit counter receives clock signal from clock selector and can be used to handle one PWM period. The 16-bit comparator compares number in counter with threshold number in register loaded previously to generate PWM duty cycle.

The clock signal from clock divider is called PWM clock. Dead-Zone generator utilize PWM clock as clock source. Once Dead-Zone generator is enabled, output of two PWM comparators is blocked. Two output pin are used as Dead-Zone generator output signal to control off-chip power device. Dead-Zone generator 0 is used to control outputs of PWM0 & PWM1, and Dead-Zone generator 1 is used to control outputs of PWM2 & PWM3.

When 16-bit down counter reaches zero, the interrupt request is generated to inform CPU that time is up. When counter reaches zero, if counter is set as toggle mode, it is reloaded automatically and starts to generate next cycle. User can set counter as one-shot mode instead of toggle mode. If counter is set as one-shot mode, counter will stop and generate one interrupt request when it reaches zero.

The value of comparator is used for pulse width modulation. The comparator control logic changes the output level when down-counter value matches the value of compare register.

The PWM timer includes a capture channel. The Capture input and PWM output share the PWM timer. Therefore user must setup the PWM timer before turn on Capture feature. After enabling Capture feature, the Capture always latches PWM counter to CRLR register when capture input has a rising transition and latches PWM counter to CFLR register when capture input has a falling transition. Capture input interrupt is programmable by setting PWM_CAPCTL[1] (Rising latch Interrupt enable) and PWM_CAPCTL[2] (Falling latch Interrupt enable) to decide the condition of interrupt event. Whenever Capture issues an interrupt, the PWM counter will be reloaded at this moment.

There is only an interrupt from PWM to interrupt controller. PWM Timer and Capture input share the same interrupt. Therefore, PWM function and Capture function in the same channel cannot be used at the same time.

5.7.2 Features

- One 8-bit prescaler and clock divider
- Five clock input selector
- One 16-bit counters and four 16-bit comparators
- Two Dead-Zone generator
- Four PWM output, one capture function

5.7.3 PWM Generator Architecture

The following figures illustrate the architecture of the PWM generator.

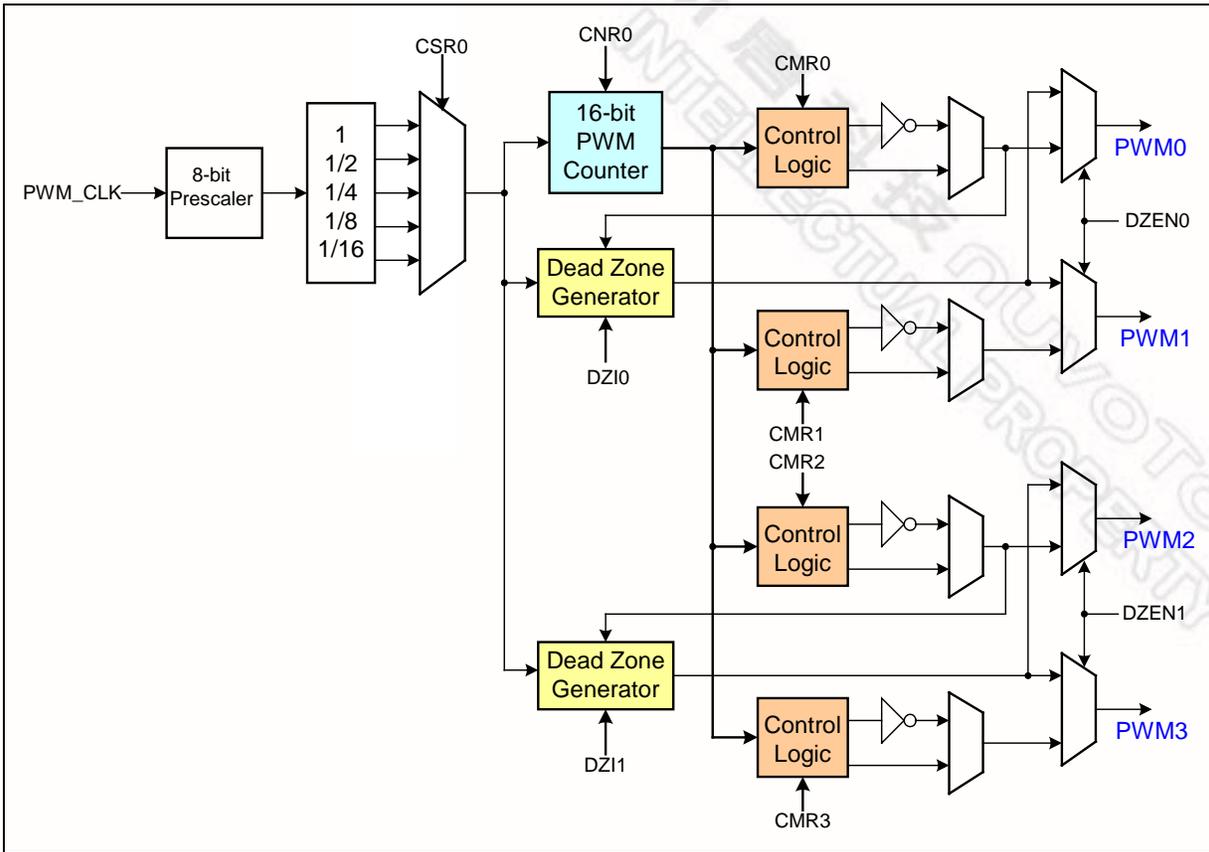


Figure 5-11 PWM Generator Architecture Diagram

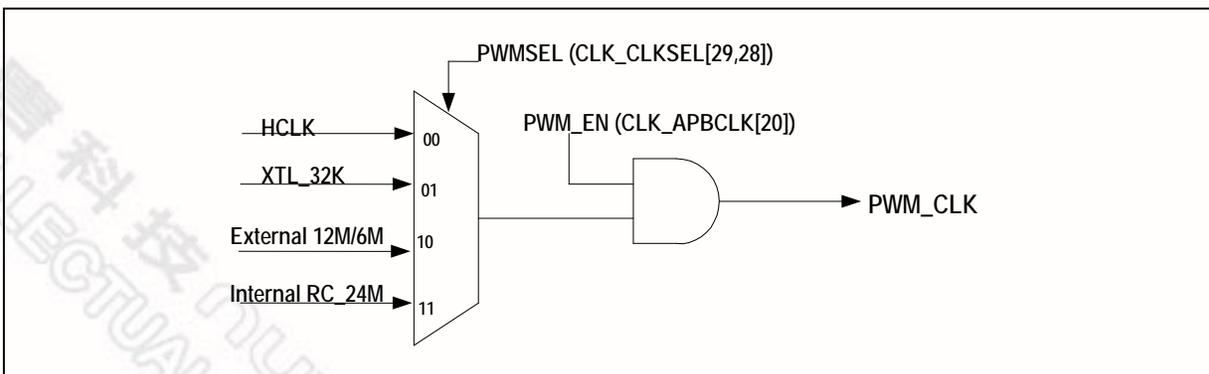


Figure 5-12 PWM Generator Clock Source Control

5.7.4 PWM-Timer Operation

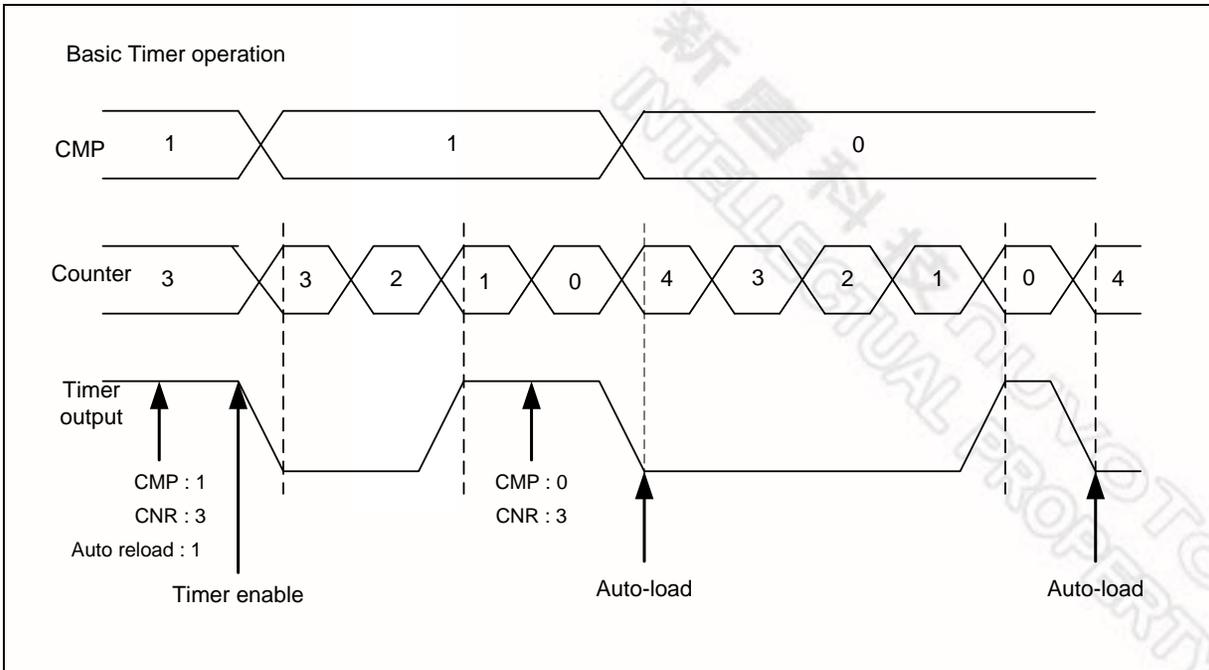


Figure 5-13 PWM Timer Operation Timing

5.7.5 PWM Auto-reload

The counter value can be written into CNR0 and current counter value can be read from PDR0.

The auto-reload operation copies from CNR0 to down-counter when down-counter reaches zero. If CNR0 is set as zero, counter will be halt when counter counts to zero. If auto-reload bit is set as zero, counter will be stopped immediately.

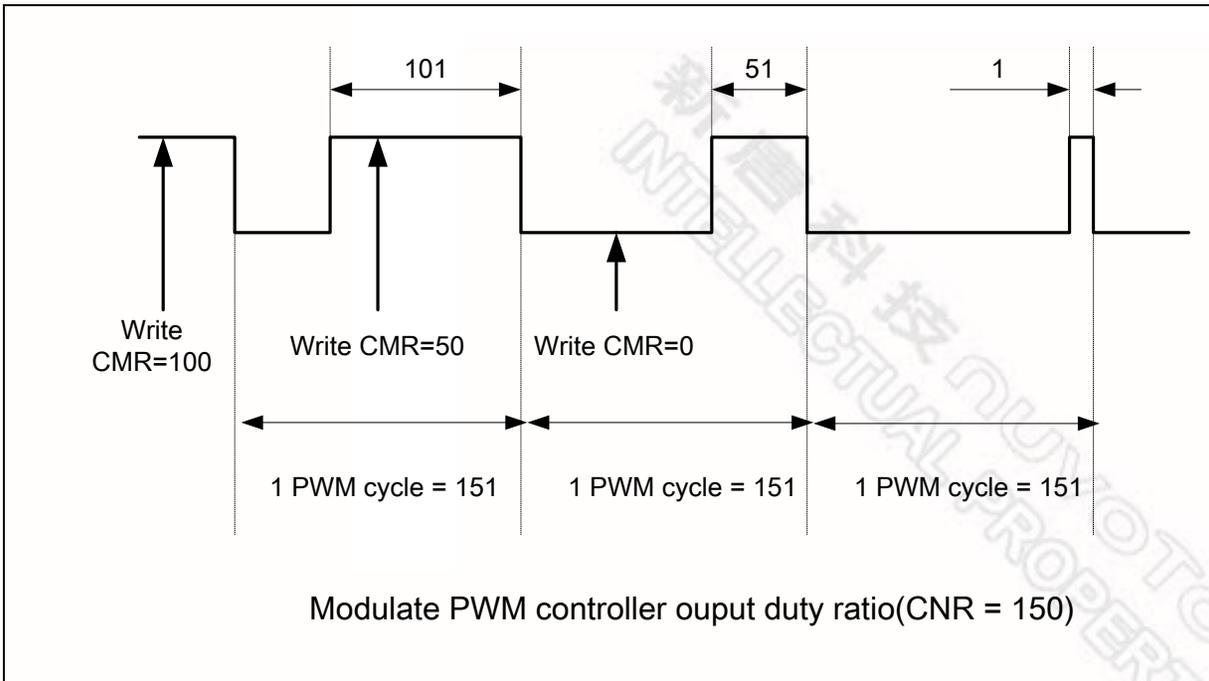


Figure 5-14 PWM Controller Output Duty Ratio.

5.7.6 Dead-Zone Generator

N572F065 PWM is implemented with Dead Zone generator. They are built for power device protection. This function enables generation of a programmable time gap at the rising of PWM output waveform. User can program PPR[31:24] and PPR[23:16] to determine the two Dead Zone interval respectively.

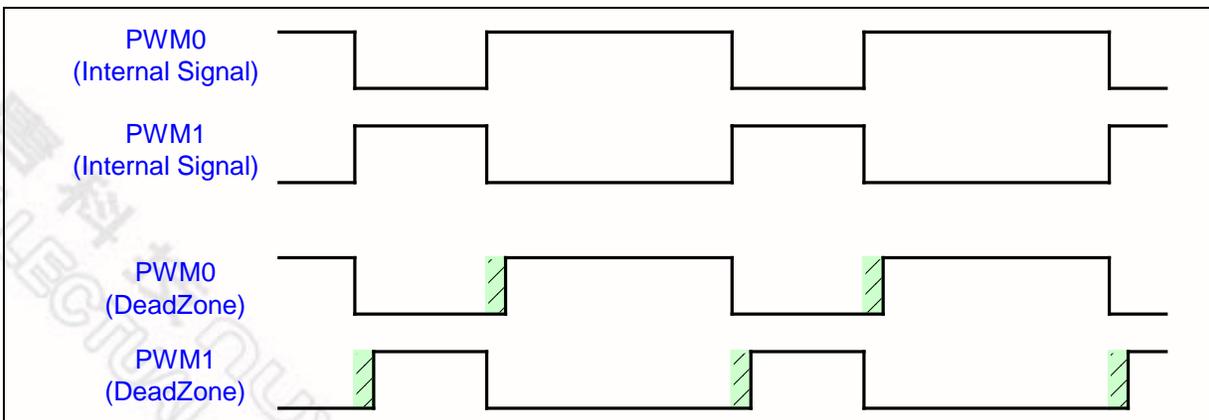


Figure 5-15 Dead Zone Generation Operation

5.7.7 PWM-Timer Start Procedure

The following procedure is recommended for starting a PWM generator.

1. Setup clock selector (PWM_CLKDIV)
2. Setup prescaler and dead zone interval (PWM_CLKPSC)
3. Setup inverter on/off, dead zone generator on/off, auto-reload/one-shot mode and PWM timer off (PWM_CTL)
4. Setup comparator register (PWM_CMPDATn) to set PWM duty cycle.
5. Setup PWM down-counter register (PWM_PERIOD) to set PWM period.
6. Setup interrupt enable register (PWM_INTEN)
7. Setup PWM output enable (PWM_PCEN)
8. Setup the corresponding GPIO pins to PWM function (SYS_GPB_MFP)
9. Enable PWM timer start running (PWM_CTL)

5.7.8 PWM-Timer Stop Procedure

Method 1:

Set 16-bit down counter (PWM_PERIOD) as 0, and monitor PWM_CNT (current value of 16-bit down-counter). When PWM_CNT reaches to 0, disable PWM-Timer (PWM_CTL). (**Recommended**)

Method 2:

Set 16-bit down counter (PWM_PERIOD) as 0. When interrupt request occurs, disable PWM-Timer (PWM_CTL). (**Recommended**)

Method 3:

Disable PWM-Timer directly (PWM_CTL). (**Not recommended**)

5.7.9 Capture Start Procedure

1. Setup clock selector (PWM_CLKDIV)
2. Setup prescaler and dead zone interval (PWM_CLKPSC)
3. Setup inverter on/off, dead zone generator on/off, auto-reload/one-shot mode and PWM timer off (PWM_CTL)
4. Setup PWM down-counter register (PWM_PERIOD)
5. Setup capture register (PWM_CAPCTL)
6. Enable PWM timer start running (PWM_CTL)

5.7.10 Capture Timer Operation

The Capture and PWM output function share the same PWM timer. The capture timer latches PWM-counter to PWM_RCAPDAT when input channel has a rising transition and latches PWM-counter to PWM_FCAPDAT when input channel has a falling transition. Capture interrupt is programmable by setting PWM_CAPCTL[1] (Rising latch Interrupt enable) and PWM_CAPCTL[2] (Falling latch Interrupt enable) to decide the condition of interrupt occurrence. Whenever the Capture module issues a capture interrupt, the corresponding PWM counter will be reloaded with PERIOD at this moment. Note that the corresponding GPIO pin must be configured as their alternate function before Capture function is enabled.

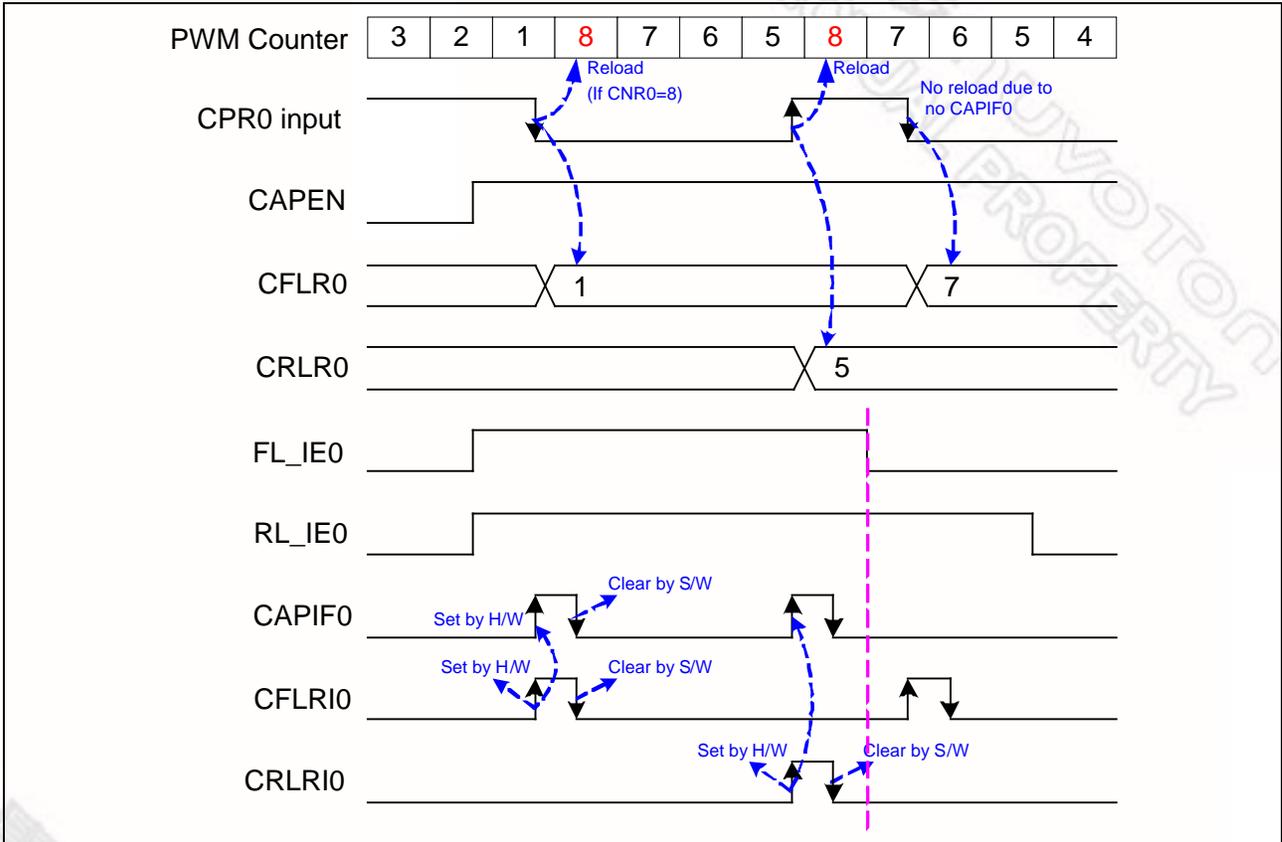


Figure 5-16 Capture Operation Timing

Figure 5-14 demonstrates the case where PERIOD = 8:

1. The PWM counter will be reloaded with PERIOD=8 at the time of interrupt occurrence when both falling and rising interrupt enabled.
2. The channel low pulse width is given by (PERIOD - RCAPDAT).
3. The channel high pulse width is given by (PERIOD - FCAPDAT).



5.7.11 Register Map

R: read only, W: write only, R/W: both read and write, C: Only value 0 can be written

Register	Offset	R/W	Description	Reset Value
PWM Base Address:				
PWM_BA = 0x4004_0000				
PWM_CLKPSC	PWM_BA+0x000	R/W	PWM Prescaler Register	0x0000_0000
PWM_CLKDIV	PWM_BA+0x004	R/W	PWM Clock Select Register	0x0000_0000
PWM_CTL	PWM_BA+0x008	R/W	PWM Control Register	0x0000_0000
PWM_PERIOD	PWM_BA+0x00C	R/W	PWM Period Register	0x0000_0000
PWM_CMPDAT0	PWM_BA+0x010	R/W	PWM Comparator Register 0	0x0000_0000
PWM_CNT	PWM_BA+0x014	R	PWM Counter Register	0x0000_0000
PWM_CMPDAT1	PWM_BA+0x01C	R/W	PWM Comparator Register 1	0x0000_0000
PWM_CMPDAT2	PWM_BA+0x028	R/W	PWM Comparator Register 2	0x0000_0000
PWM_CMPDAT3	PWM_BA+0x034	R/W	PWM Comparator Register 3	0x0000_0000
PWM_INTEN	PWM_BA+0x040	R/W	PWM Interrupt Enable Register	0x0000_0000
PWM_INTSTS	PWM_BA+0x044	R/W	PWM Interrupt Flag Register	0x0000_0000
PWM_CAPCTL	PWM_BA+0x050	R/W	Capture Control Register	0x0000_0000
PWM_RCAPDAT	PWM_BA+0x058	R	Capture Rising Latch Register	0x0000_0000
PWM_FCAPDAT	PWM_BA+0x05C	R	Capture Falling Latch Register	0x0000_0000
PWM_PCEN	PWM_BA+0x07C	R/W	PWM Output and Capture Input Enable Register	0x0000_0000

5.7.12 Register Description

PWM Pre-Scale Register (PWM_CLKPSC)

Register	Offset	R/W	Description	Reset Value
PWM_CLKPSC	PWM_BA+0x000	R/W	PWM Prescaler Register	0x0000_0000

31	30	29	28	27	26	25	24
DZ11							
23	22	21	20	19	18	17	16
DZ10							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
CLKPSC							

Bits	Description	
[31:24]	DZ11	Dead Zone Interval Register 1 These 8 bits determine dead zone length. The unit time of dead zone length is that from clock selector.
[23:16]	DZ10	Dead Zone Interval Register 0 These 8 bits determine dead zone length. The unit time of dead zone length is that from clock selector.
[15:8]	Reserved	Reserved.
[7:0]	CLKPSC	Clock Prescaler For PWM Timer Clock input is divided by (CLKPSC + 1) If CLKPSC = 0, then the prescaler output clock will be stopped. This implies PWM counter will also be stopped.



PWM Clock Select Register (PWM_CLKDIV)

Register	Offset	R/W	Description	Reset Value
PWM_CLKDIV	PWM_BA+0x004	R/W	PWM Clock Select Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					CLKDIV		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	CLKDIV	PWM Timer Clock Source Selection Value : Input clock divided by 000 : 2 001 : 4 010 : 8 011 : 16 1xx : 1



PWM Control Register (PWM_CTL)

Register	Offset	R/W	Description	Reset Value
PWM_CTL	PWM_BA+0x008	R/W	PWM Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved		DTEN1	DTEN0	CNTMODE	PINV	Reserved	CNTEN

Bits	Description	
[31:6]	Reserved	Reserved.
[5]	DTEN1	Dead-Zone 1 Generator Enable/Disable 0 = Disable. 1 = Enable.
[4]	DTEN0	Dead-Zone 0 Generator Enable/Disable 0 = Disable. 1 = Enable.
[3]	CNTMODE	PWM-Timer Auto-Reload/One-Shot Mode 0 = One-Shot Mode. 1 = Auto-reload Mode.
[2]	PINV	PWM-Timer Output Inverter ON/OFF 0 = Inverter OFF. 1 = Inverter ON.
[0]	CNTEN	PWM-Timer Enable 0 = Stop PWM-Timer Running. 1 = Enable PWM-Timer.



PWM Period Register (PWM_PERIOD)

Register	Offset	R/W	Description	Reset Value
PWM_PERIOD	PWM_BA+0x00C	R/W	PWM Period Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
PERIOD [15:8]							
7	6	5	4	3	2	1	0
PERIOD [7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	PERIOD	PWM Counter/Timer Reload Value PERIOD determines the PWM period. Note: One PWM cycle width = (PERIOD + 1).

PWM Comparator Register 3-0 (PWM_CMPDAT3-0)

Register	Offset	R/W	Description	Reset Value
PWM_CMPDAT0	PWM_BA+0x010	R/W	PWM Comparator Register 0	0x0000_0000
PWM_CMPDAT1	PWM_BA+0x01C	R/W	PWM Comparator Register 1	0x0000_0000
PWM_CMPDAT2	PWM_BA+0x028	R/W	PWM Comparator Register 2	0x0000_0000
PWM_CMPDAT3	PWM_BA+0x034	R/W	PWM Comparator Register 3	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
CMP[15:8]							
7	6	5	4	3	2	1	0
CMP[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	CMP	<p>PWM Comparator Register CMP determines the PWM duty ratio. Assumption: PWM output initial is high</p> <ul style="list-style-type: none"> • $CMP \geq PERIOD$: PWM output is always high. • $CMP < PERIOD$: PWM low width = (PERIOD - CMP) unit; PWM high width = (CMP+1) unit. • $CMP = 0$: PWM low width = (PERIOD) unit; PWM high width = 1 unit. <p>Note1: Unit = one PWM clock cycle. Note2: Any write to CMP will take effect in next PWM cycle.</p>

PWM Counter Register (PWM_CNT)

Register	Offset	R/W	Description	Reset Value
PWM_CNT	PWM_BA+0x014	R	PWM Counter Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
CNT[15:8]							
7	6	5	4	3	2	1	0
CNT[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	CNT	PWM Counter Register Reports the current value of the 16-bit down counter.



PWM Interrupt Enable Register (PWM_INTEN)

Register	Offset	R/W	Description	Reset Value
PWM_INTEN	PWM_BA+0x040	R/W	PWM Interrupt Enable Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							PIEN

Bits	Description	
[31:1]	Reserved	Reserved.
[0]	PIEN	PWM Timer Interrupt Enable 0 = Disable. 1 = Enable.

PWM Interrupt Flag Register (PWM_INTSTS)

Register	Offset	R/W	Description	Reset Value
PWM_INTSTS	PWM_BA+0x044	R/W	PWM Interrupt Flag Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							PIF

Bits	Description	
[31:1]	Reserved	Reserved.
[0]	PIF	PWM Timer Interrupt Flag Flag is set by hardware when PWM down counter reaches zero, software can clear this bit by writing '1' to it.



Capture Control Register (PWM_CAPCTL)

Register	Offset	R/W	Description	Reset Value
PWM_CAPCTL	PWM_BA+0x050	R/W	Capture Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
CFLIF	CRLIF	Reserved	CAPIF	CAPEN	CFLIEN	CRLIEN	CAPIV

Bits	Description
[31:8]	Reserved Reserved.
[7]	CFLIF PWM_FCAPDAT Latched Indicator Bit When input channel has a falling transition, PWM_FCAPDAT was latched with the value of PWM down-counter and this bit is set by hardware, software can clear this bit by writing a zero to it.
[6]	CRLIF PWM_RCAPDAT Latched Indicator Bit When input channel has a rising transition, PWM_RCAPDAT was latched with the value of PWM down-counter and this bit is set by hardware, software can clear this bit by writing a zero to it.
[5]	Reserved Reserved.
[4]	CAPIF Capture Interrupt Indication Flag When capture input has a falling/rising transition and falling/rising latch interrupt is enabled (CFLIEN = 1/CRLIEN = 1), CAPIF0 is set 1 by hardware. Software can clear this bit by writing a one to it. Note: If this bit is "1", PWM counter will not be reloaded when next capture interrupt occurs.
[3]	CAPEN Capture Channel Input Transition Enable/Disable 0 = Disable capture function. 1 = Enable capture function. When enabled, Capture function latches the PMW-counter to RCAPDAT (Rising latch) and FCAPDAT (Falling latch) registers on input edge transition. When disabled, Capture function is inactive as is interrupt.
[2]	CFLIEN Falling Latch Interrupt Enable ON/OFF 0 = Disable falling latch interrupt. 1 = Enable falling latch interrupt. When enabled, capture block generates an interrupt on falling edge of input.

[1]	CRLIEN	Rising Latch Interrupt Enable ON/OFF 0 = Disable rising latch interrupt. 1 = Enable rising latch interrupt. When enabled, capture block generates an interrupt on rising edge of input.
[0]	CAPINV	Inverter ON/OFF 0 = Inverter OFF. 1 = Inverter ON. Reverse the input signal from GPIO before fed to Capture timer

Capture Rising Latch Register (PWM_RCAPDAT)

Register	Offset	R/W	Description	Reset Value
PWM_RCAPDAT	PWM_BA+0x058	R	Capture Rising Latch Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
RCAPDAT[15:8]							
7	6	5	4	3	2	1	0
RCAPDAT[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	RCAPDAT	Capture Rising Latch Register In Capture mode, this register is latched with the value of the PWM counter on a rising edge of the input signal.



Capture Falling Latch Register (PWM FCAPDAT)

Register	Offset	R/W	Description	Reset Value
PWM_FCAPDAT	PWM_BA+0x05C	R	Capture Falling Latch Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
FCAPDAT[15:8]							
7	6	5	4	3	2	1	0
FCAPDAT[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	FCAPDAT	Capture Falling Latch Register In Capture mode, this register is latched with the value of the PWM counter on a falling edge of the input signal.

PWM Output and Capture Input Enable Register (PWM_PCEN)

Register	Offset	R/W	Description	Reset Value
PWM_PCEN	PWM_BA+0x07C	R/W	PWM Output and Capture Input Enable Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							CAPINEN
7	6	5	4	3	2	1	0
Reserved				POEN3	POEN2	POEN1	POEN0

Bits	Description	
[31:9]	Reserved	Reserved.
[8]	CAPINEN	Capture Input Enable Register 0 = OFF (PB.12 pin input disconnected from Capture block). 1 = ON (PB.12 pin, if in PWM alternative function, will be configured as an input and fed to capture function).
[7:4]	Reserved	Reserved.
[3]	POEN3	PWM3 Output Enable Register 0 = Disable PWM3 output to pin. 1 = Enable PWM3 output to pin. Note: The corresponding GPIO pin also must be switched to PWM function (refer to SYS_GPB_MFP Error! Reference source not found.)
[2]	POEN2	PWM2 Output Enable Register 0 = Disable PWM2 output to pin. 1 = Enable PWM2 output to pin. Note: The corresponding GPIO pin also must be switched to PWM function (refer to SYS_GPA_MFP Error! Reference source not found.)
[1]	POEN1	PWM1 Output Enable Register 0 = Disable PWM1 output to pin. 1 = Enable PWM1 output to pin. Note: The corresponding GPIO pin also must be switched to PWM function (refer to SYS_GPA_MFP Error! Reference source not found.)
[0]	POEN0	PWM0 Output Enable Register 0 = Disable PWM0 output to pin. 1 = Enable PWM0 output to pin. Note: The corresponding GPIO pin also must be switched to PWM function (refer to SYS_GPA_MFP Error! Reference source not found.)



5.8 Real Time Clock (RTC)

5.8.1 Overview

The 32 KHz crystal oscillator is clock source of RTC. The RTC time-out frequency is programmable. During normal operation, the RTC time-out can generate an interrupt at a pre-programmed frequency. During power down mode, the RTC time-out could be one of wake-up source.

5.8.2 Register Map

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
RTC Base Address: RTC_BA = 0x4000_8000				
RTC_CTL	RTC_BA+0x000	R/W	RTC Control Register	0x0000_0000

5.8.3 Register Description

RTC Control Register (RTC_CTL)

Register	Offset	R/W	Description	Reset Value
RTC_CTL	RTC_BA+0x000	R/W	RTC Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved			RTIS		RTCE	RTIE	RTIF

Bits	Description
[31:5]	Reserved
[4:3]	RTIS RTC Timer Interval Select These two bits select the timeout interval for the RTC. 00 = Time-out frequency is 0.25Hz., 01 = Time-out frequency is 2Hz., 10 = Time-out frequency is 8Hz., 11 = Time-out frequency is 32Hz.



[2]	RTCE	RTC Enable 0 = Disable RTC function. 1 = Enable RTC function.
[1]	RTIE	RTC Interrupt Enable 0 = Disable the RTC interrupt. 1 = Enable the RTC interrupt.
[0]	RTIF	RTC Interrupt Flag If the RTC interrupt is enabled, then the hardware will set this bit to indicate that the RTC interrupt has occurred. If the RTC interrupt is not enabled, then this bit indicates that a timeout period has elapsed. 0 = RTC interrupt does not occur. 1 = RTC interrupt occurs. Note: This bit is cleared by writing 1 to this bit.

5.9 Serial Peripheral Interface (SPI) Controller

5.9.1 Overview

The SPI Synchronous Serial Interface controller performs a serial-to-parallel conversion on data received from the peripheral, and a parallel-to-serial conversion on data received from CPU. This interface can drive up to 2 external peripherals, but is time-shared and cannot operate simultaneously. Only master mode SPI is supported.

It can generate an interrupt signal when data transfer is finished and can be cleared by writing 1 to the interrupt flag. The active level of device/slave select signal can be chosen to low active or high active on SS_LVL bit, which depends on the peripheral it's connected. Writing a divisor into DIVIDER register can program the frequency of serial clock output when it is as the master.

This SPI core contains two 32-bit transmit/receive buffers, and can provide burst mode operation. It supports variable length transfer and the maximum transmitted/received length can be up to 32 bits.

5.9.2 Features

- AMBA APB interface compatible
- Support SPI master mode
- Variable length of transfer word up to 32 bits
- Provide burst mode operation, transmit/receive can be executed up to two times in one transfer
- MSB or LSB first data transfer
- 2 slave/device select pins
- Fully static synchronous design with one clock domain

5.9.3 SPI Block Diagram

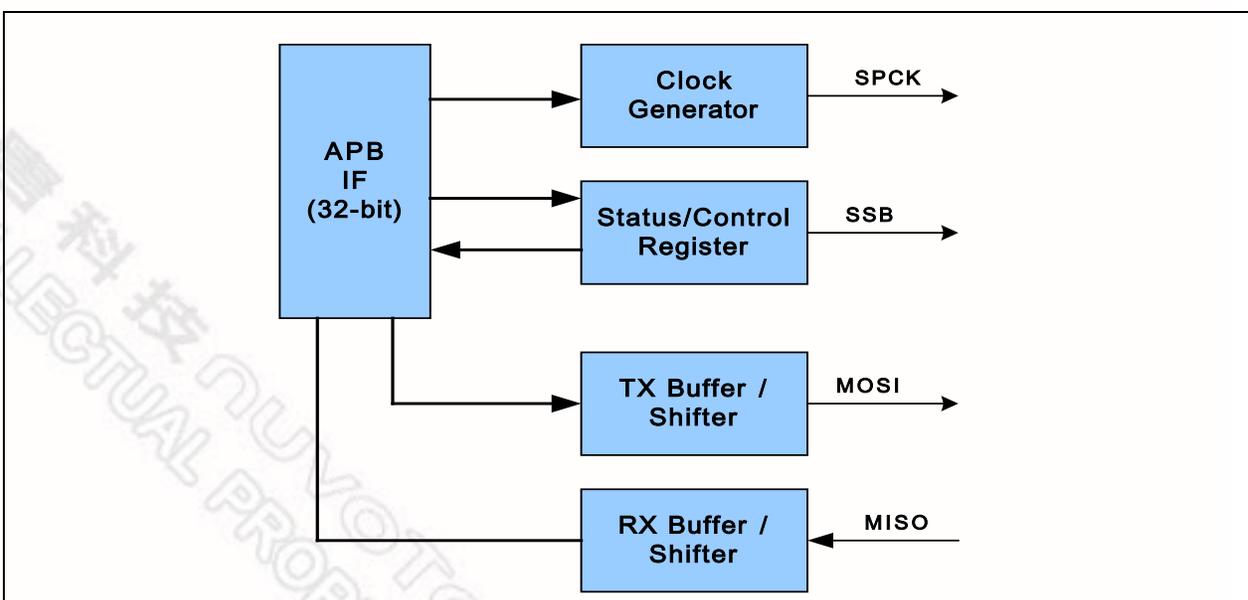


Figure 5-17 SPI Block Diagram

SPI Timing Diagram

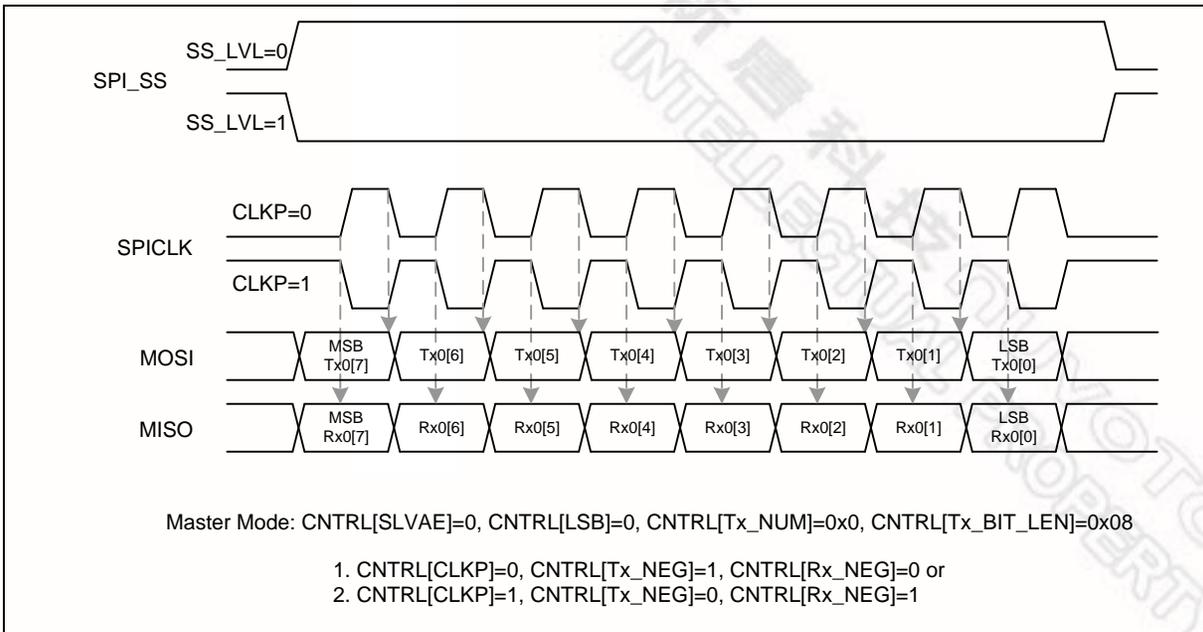


Figure 5-18 SPI Timing

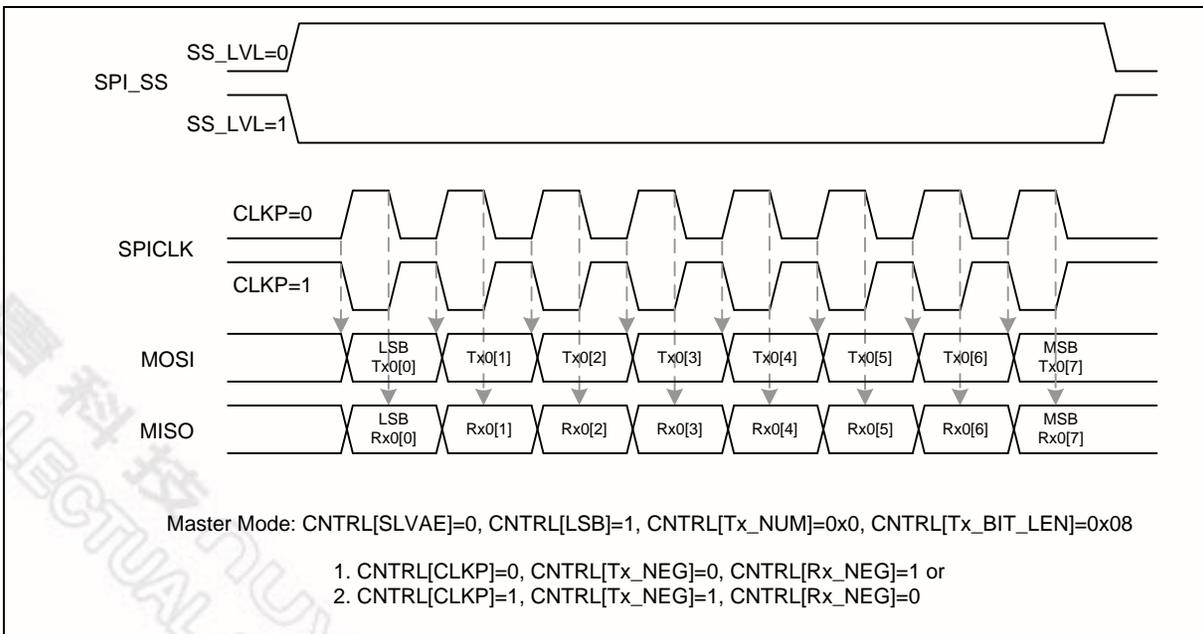


Figure 5-19 Alternate Phase SCLK Clock Timing

5.9.4 SPI Configuration Examples

When using this SPI controller as a master to access a slave device with following specifications:

- Data bit latches on positive edge of serial clock
- Data bit drives on negative edge of serial clock
- Data is transferred with the MSB first
- SPCK idle low.
- Only one byte transmit/receive in a transfer
- Device/Slave select signal is active low

Basically, the following actions should be done (also, the specification of the connected slave device should be referred to when consider the following steps in detail):

- 1) Write a divisor into DIVIDER to determine the frequency of serial clock.
- 2) Setup SPI_SSR. Set AUTOSS=0, SS_LVL=0 and SS[0] or SS[1] to 1 to activate the device to be accessed.

When transmit (write) data to device:

- 3) Write the data to be transmitted into SPI_TX0[7:0].

When receive (read) data from device:

- 4) Write 0xFFFFFFFF into SPI_TX0.
- 5) Setup SPI_CTL. Set CLKP=0, RX_NEG=0, TX_NEG=1, DWIDTH=0x08, TX_NUM=0x0, LSB=0, SUSPITV=0x0, and GO_BUSY=1 to start the transfer.

--- Wait for interrupt (if UNIT_INTEN=1) or polling the GO_BUSY bit until it turns to 0 ---

- 6) Read out the received data from SPI_RX0.
- 7) Go to 3) to continue another data transfer, or set SS[0] or SS[1] to 0 to inactivate the device.



5.9.5 SPI Control Register Map

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
SPI Base Address: $SPI_n_BA = 0x4003_0000 + (0x4000 * n)$ $n = 0,1$				
SPI_CTL	SPI _n _BA+0x00	R/W	Control and Status Register	0x0000_0004
SPI_CLKDIV	SPI _n _BA+0x04	R/W	Clock Divider Register	0x0000_0000
SPI_SSCTL	SPI _n _BA+0x08	R/W	Slave Select Register	0x0000_0000
SPI_RX0	SPI _n _BA+0x10	R	Data Receive Register 0	0x0000_0000
SPI_RX1	SPI _n _BA+0x14	R	Data Receive Register 1	0x0000_0000
SPI_TX0	SPI _n _BA+0x20	W	Data Transmit Register 0	0x0000_0000
SPI_TX1	SPI _n _BA+0x24	W	Data Transmit Register 1	0x0000_0000



5.9.6 SPI Control Register Description

SPI Control and Status Register (SPI_CTL)

Register	Offset	R/W	Description	Reset Value
SPI_CTL	SPIIn_BA+0x00	R/W	Control and Status Register	0x0000_0004

31	30	29	28	27	26	25	24	
Reserved								
23	22	21	20	19	18	17	16	
Reserved			REORDER	Reserved			UNIT_INTEN	UNIT_INTSTS
15	14	13	12	11	10	9	8	
SUSPITV				CLKP	LSB	TX_NUM		
7	6	5	4	3	2	1	0	
DWIDTH					TX_NEG	RX_NEG	GO_BUSY	

Bits	Description	Description
[31:21]	Reserved	Reserved.
[20]	REORDER	<p>BYTE ENDIAN</p> <p>0 = Disable the BYTE ENDIAN. 1 = Enable the BYTE ENDIAN. Only the 16, 24, and 32 bits which are defined in DWIDTH are supported.</p> <p>When the transition is set as MSB first and REORDER=1, the data stored in the SPI_TXn buffer will be arranged in the order as [BYTE0, BYTE1, BYTE2, BYTE3] in DWIDTH=32-bit mode, and the sequence of transmitted data will be BYTE0, BYTE1, BYTE2, and BYTE3. If the DWIDTH is set as 24-bit mode, the data in SPI_TXn buffer will be arranged as [unknown_byte, BYTE0, BYTE1, BYTE2] and the BYTE0, BYTE1, and BYTE2 will be transmitted step by step in MSB first. The rule of 16-bit mode is the same as above.</p>
[19:18]	Reserved	Reserved.
[17]	UNIT_INTEN	<p>Unit Transfer Interrupt Enable</p> <p>0 = Disable SPI Unit Transfer Interrupt. 1 = Enable SPI Unit Transfer Interrupt to CPU.</p>
[16]	UNIT_INTSTS	<p>Unit Transfer Interrupt Status</p> <p>0 = No transaction has been finished since this bit was cleared to 0. 1 = SPI controller has finished one unit transfer.</p> <p>Note: This bit is read only, but can be cleared by writing 1 to this bit.</p>



[15:12]	SUSPITV	<p>Suspend Interval (Master Mode Only)</p> <p>The four bits provide configurable suspend interval between two successive transmit/receive transaction in a transfer. The definition of the suspend interval is the interval between the last falling edge of SPI_SCLKn of the preceding transaction word and the first rising edge of SPI_SCLKn of the following transaction word. The default value is 0x0. When TX_NUM=00, setting this field has no effect on transfer. The period of the suspend interval is obtained according to the following equation. $(SUSPITV+2) * SPI_SCLKn \text{ clock cycles}$ Note: SUSPITV cannot be "0" for SPI0.</p>
[11]	CLKP	<p>Clock Polarity</p> <p>0 = SPI_SCLKn idle low. 1 = SPI_SCLKn idle high.</p>
[10]	LSB	<p>Send LSB First</p> <p>0 = The MSB is transmitted/received first (which bit in SPI_TXn/SPI_RXn is MSB is dependent on DWIDTH). 1 = The LSB (SPI_TXn[0]) is sent first to SPI_MOSIn, and the first bit received from SPI_MISOIn will be put in the LSB (SPI_RXn[0]).</p>
[9:8]	TX_NUM	<p>Transmit/Receive Numbers</p> <p>This field specifies how many transmit/receive numbers should be executed in one transfer. 00 = Only one transmit/receive will be executed in one transfer. 01 = Two successive transmit/receive will be executed in one transfer. 1x = Reserved.</p>
[7:3]	DWIDTH	<p>Transmit Bit Length</p> <p>This field specifies how many bits are transmitted in one transmit/receive. Up to 32 bits can be transmitted. DWIDTH = 0x01 ... 1 bit DWIDTH = 0x02 ... 2 bits DWIDTH = 0x1F ... 31 bits DWIDTH = 0x00 ... 32 bits.</p>
[2]	TX_NEG	<p>Transmit On Negative Edge</p> <p>0 = The output on SPI_MOSIn is changed on the rising edge of SPI_SCLKn. 1 = The output on SPI_MOSIn is changed on the falling edge of SPI_SCLKn.</p>
[1]	RX_NEG	<p>Receive On Negative Edge</p> <p>0 = The input on SPI_MISOIn is latched on the rising edge of SPI_SCLKn. 1 = The input on SPI_MISOIn is latched on the falling edge of SPI_SCLKn.</p>
[0]	GO_BUSY	<p>Go And Busy Status</p> <p>0 = Writing 0 to this bit has no effect. 1 = Writing 1 to this bit starts the transfer. This bit remains "1" during the transfer and is automatically cleared after the transfer is finished. NOTE: All registers should be set readily before writing 1 to the GO_BUSY bit. When a transfer is in progress, writing to any register of the SPI core has no effect.</p>

NuVoice™ N572F065 Technical Reference Manual

SPI Divider Register (SPI_CLKDIV)

Register	Offset	R/W	Description	Reset Value
SPI_CLKDIV	SPIn_BA+0x04	R/W	Clock Divider Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
DIVIDER[15:8]							
7	6	5	4	3	2	1	0
DIVIDER[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	DIVIDER	<p>Clock Divider Register</p> <p>The value in this field is the frequency divider of the system clock, PCLK, to generate the serial clock on the output (SPICLK). The desired frequency is obtained according to the following equation:</p> $f_{sclk} = \frac{f_{pclk}}{(DIVIDER + 1) * 2}$ <p>NOTE: Suggest DIVIDER should be at least 1 in master mode.</p>



SPI Slave Select Register (SPI_SSCTL)

Register	Offset	R/W	Description	Reset Value
SPI_SSCTL	SPIn_BA+0x08	R/W	Slave Select Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved				AUTOSS	SS_LVL	SS	

Bits	Description
[31:4]	Reserved Reserved.
[3]	AUTOSS Automatic Slave Select 0 = Slave select signal (SPI_SSBx0/SPI_SSBx1) is asserted and de-asserted by setting and clearing related bit SS. 1 = Slave select signal (SPI_SSBx0/SPI_SSBx1) is generated automatically. It means that slave select signal, which is set in bits SS, is asserted by the SPI controller when transmit/receive is started by setting GO_BUSY, and is de-asserted after the transfer is finished.
[2]	SS_LVL Slave Select Active Level It defines the active level of device/slave select signal. 0 = The SPI_SSBx0/SPI_SSBx1 slave select signal is active Low. 1 = The SPI_SSBx0/SPI_SSBx1 slave select signal is active High.
[1:0]	SS Slave Select Pin Control (Master Mode Only) If AUTOSS bit is 0, SPIn_SSBx0 and SPIn_SSBx1 output are determined by SS[0] and SS[1] respectively. 0 = Any bit location of this field forces the pin to inactive state. 1 = Any bit location of this field forces the proper SPIn_SSBx0/SPI_SSBx1 pin to an active state If AUTOSS bit is 1, 0 = Any bit location of this field will select appropriate SPIn_SSBx0/SPI_SSBx1 pin to be driven to inactive state. 1 = Any bit location of this field will select appropriate SPIn_SSBx0/SPI_SSBx1 pin to be automatically driven to active state for the duration of the transmit/receive, and to be driven to inactive state for the rest of the time. The active state of SPIn_SSBx0/SPI_SSBx1 is specified in SS_LVL bit (SPI_SSCTL[2]).



SPI Data Receive Register (SPI_RXn)

Register	Offset	R/W	Description	Reset Value
SPI_RX0	SPIn_BA+0x10	R	Data Receive Register 0	0x0000_0000
SPI_RX1	SPIn_BA+0x14	R	Data Receive Register 1	0x0000_0000

31	30	29	28	27	26	25	24
RX[31:24]							
23	22	21	20	19	18	17	16
RX[23:16]							
15	14	13	12	11	10	9	8
RX[15:8]							
7	6	5	4	3	2	1	0
RX[7:0]							

Bits	Description
[31:0]	<p>RX</p> <p>Data Receive Register</p> <p>The Data Receive Registers hold the value of received data of the last executed transfer. Valid bits depend on the transmit bit length field in the SPI_CTL register. For example, if DWIDTH=0x08 and TX_NUM=0x0, bits SPI_RX0[7:0] hold the received data.</p> <p>Note: The Data Receive Registers are read only registers. A Write to these registers will actually modify the Data Transmit Registers because those registers share the same flip-flops.</p>

SPI Data Transmit Register (SPI_TXn)

Register	Offset	R/W	Description	Reset Value
SPI_TX0	SPIn_BA+0x20	W	Data Transmit Register 0	0x0000_0000
SPI_TX1	SPIn_BA+0x24	W	Data Transmit Register 1	0x0000_0000

31	30	29	28	27	26	25	24
TX[31:24]							
23	22	21	20	19	18	17	16
TX[23:16]							
15	14	13	12	11	10	9	8
TX[15:8]							
7	6	5	4	3	2	1	0
TX[7:0]							

Bits	Description
[31:0]	<p>TX</p> <p>Data Transmit Register</p> <p>The Data Transmit Registers hold the data to be transmitted in the next transfer. Valid bit depends on the transmit bit length field in the SPI_CTL register. For example, if DWIDTH=0x08 and the TX_NUM=0x0, the bits SPI_TX0[7:0] will be transmitted in next transfer. If DWIDTH=0x00 and TX_NUM=0x1, H/W will perform two 32-bit transmit/receive successively with the data order {SPI_TX0[31:0], SPI_TX1[31:0]}.</p> <p>Note: The SPI_RXn and SPI_TXn registers share the same flip-flops, which mean that what is received in one transfer will be transmitted in the next transfer if there is not any write to the SPI_TXn register between the two transfers.</p>

5.10 Timer Controller

5.10.1 General Timer Controller

The timer module includes three channels, Timer0, Timer1 and Timer2 which allow user to easily implement a counting scheme for use. The timer can perform functions like frequency measurement, event counting, interval measurement, clock generation, delay timing, and so on. The timer possesses features such as adjustable resolution, programmable counting period, and detailed information. The timer can generate an interrupt signal upon timeout, or provide the current value of count during operation.

Besides the general timer function, Timer0 and Timer1 provide another function. Timer0 time out rate is used as APU sample output rate when APU is enabled. Timer1 can be used to generate IR carrier output.

A simple fixed frequency timer TimerF is provided. Its frequency can be one of the followings.

12MHz (or 6MHz) crystal / 16384,
 12MHz (or 6MHz) crystal / (4x16384),
 32KHz crystal / 32,
 32KHz crystal / (4x32),
 24MHz RC oscillator / 32768,
 24MHz RC oscillator / (4x32768).

5.10.2 Features

The general timer (Timer0/1/2) controller includes the following features

- AMBA APB interface compatible
- Each channel with an 8-bit pre-scale counter, a 16-bit up-counter and an interrupt request signal.
- Independent clock source for each channel.
- Time out period = (Period of timer clock input) * (Prescale + 1) * (TCMP)
- Maximum counting cycle time = (1/25 MHz) * (2⁸) * (2¹⁶), if TMRx_CLK = 25 MHz.
- Internal 16-bit up counter is readable on the fly.

5.10.3 Timer Controller Block Diagram

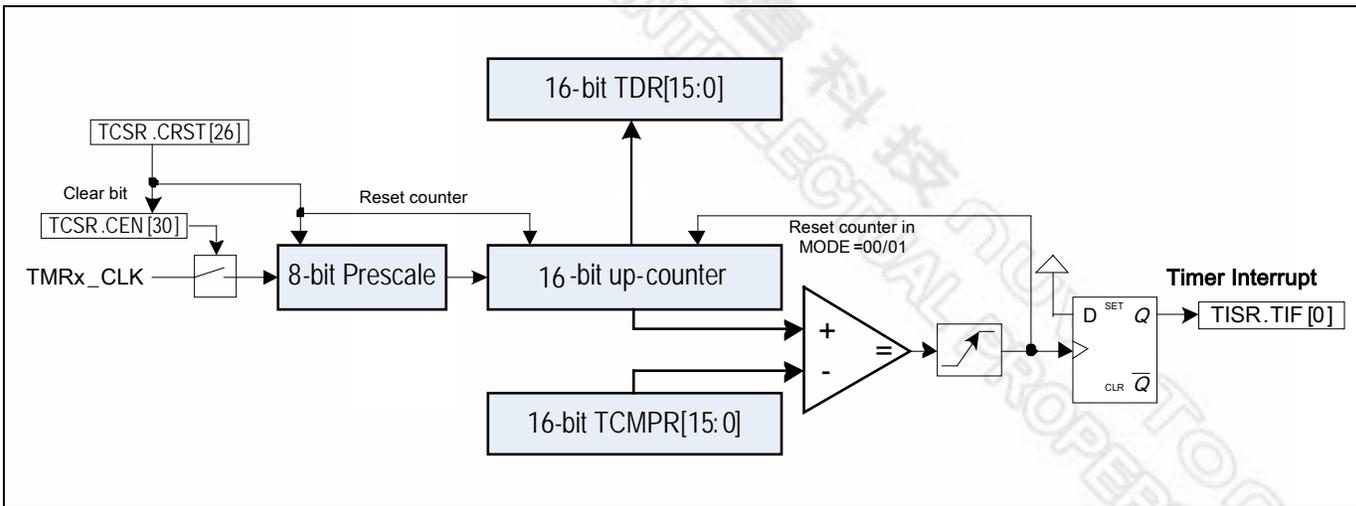


Figure 5-20 Timer0/1/2 Block Diagram

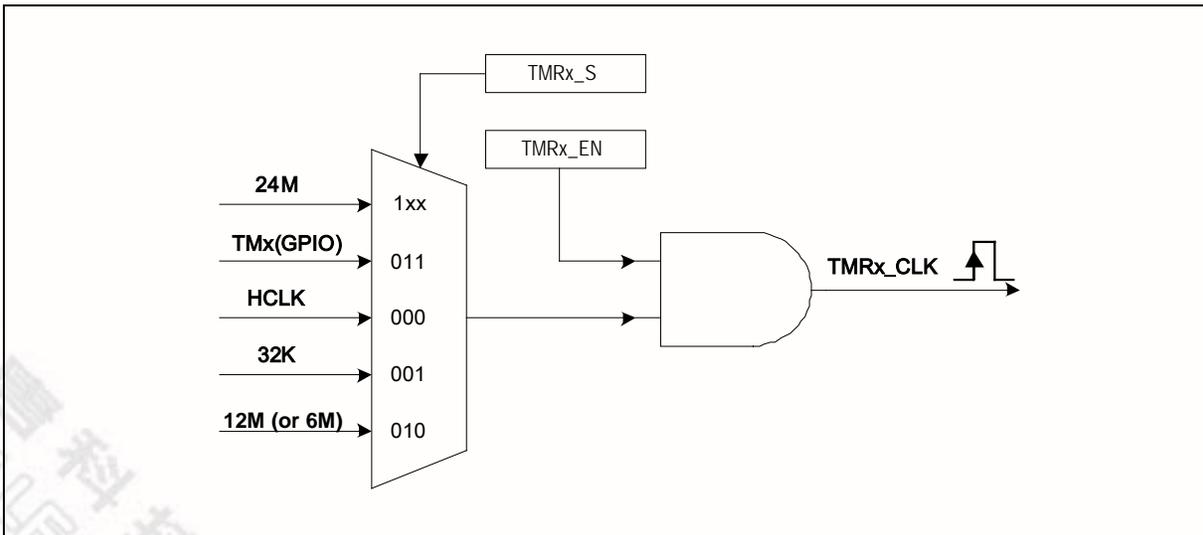


Figure 5-21 Clock Source of Timer0/1/2



5.10.4 Timer Controller Register Map

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
TMR Base Address: TMR_BA=0x4001_0000				
TIMER0_CTL	TMR_BA+0x00	R/W	Timer0 Control and Status Register	0x0000_0005
TIMER0_CMP	TMR_BA+0x04	R/W	Timer0 Compare Register	0x0000_0000
TIMER0_INTSTS	TMR_BA+0x08	R/W	Timer0 Interrupt Status Register	0x0000_0000
TIMER0_CNT	TMR_BA+0x0C	R	Timer0 Data Register	0x0000_0000
TIMER1_CTL	TMR_BA+0x20	R/W	Timer1 Control and Status Register	0x0000_0005
TIMER1_CMP	TMR_BA+0x24	R/W	Timer1 Compare Register	0x0000_0000
TIMER1_INTSTS	TMR_BA+0x28	R/W	Timer1 Interrupt Status Register	0x0000_0000
TIMER1_CNT	TMR_BA+0x2C	R	Timer1 Data Register	0x0000_0000
TIMERF_INTSTS	TMR_BA+0x30	R/W	TimerF Interrupt Status Register	0x0000_0000
IR_CTL	TMR_BA+0x34	R/W	IR Carrier Output Control Register	0x0000_0000
TIMER2_CTL	TMR_BA+0x40	R/W	Timer2 Control and Status Register	0x0000_0005
TIMER2_CMP	TMR_BA+0x44	R/W	Timer2 Compare Register	0x0000_0000
TIMER2_INTSTS	TMR_BA+0x48	R/W	Timer2 Interrupt Status Register	0x0000_0000
TIMER2_CNT	TMR_BA+0x4C	R	Timer2 Data Register	0x0000_0000



5.10.5 Timer Controller Register Description

Timer Control Register (TIMERx_CTL)

Register	Offset	R/W	Description	Reset Value
TIMER0_CTL	TMR_BA+0x00	R/W	Timer0 Control and Status Register	0x0000_0005
TIMER1_CTL	TMR_BA+0x20	R/W	Timer1 Control and Status Register	0x0000_0005
TIMER2_CTL	TMR_BA+0x40	R/W	Timer2 Control and Status Register	0x0000_0005

31	30	29	28	27	26	25	24
Reserved	CNTEN	INTEN	OPMODE[1:0]		RSTCNT	ACTSTS	Reserved
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
PSC[7:0]							

Bits	Description	
[31]	Reserved.	Reserved.
[30]	CNTEN	<p>Counter Enable Bit</p> <p>0 = Stop/Suspend counting. 1 = Start counting.</p> <p>Note: This bit is auto-cleared by hardware in one-shot mode (OPMODE = 00b) when the associated timer interrupt is generated (INTEN = 1).</p>
[29]	INTEN	<p>Interrupt Enable Bit</p> <p>0 = Disable TIMER Interrupt. 1 = Enable TIMER Interrupt.</p> <p>If timer interrupt is enabled, the timer asserts its interrupt signal when the associated count is equal to TIMERx_CMP.</p>
[28:27]	OPMODE	<p>Timer Operating Mode</p> <p>00 = The Timer is operating in the one-shot mode. The associated interrupt signal is generated once (if INTEN is 1) and CNTEN is automatically cleared by hardware. 01 = The Timer is operating in the periodic mode. The associated interrupt signal is generated periodically (if INTEN is 1). 10 = Reserved. 11 = The Timer is operating in continuous counting mode. The associated interrupt signal is generated when TIMERx_CNT = TIMERx_CMP (if INTEN is 1); however, the 16-bit up-counter counts continuously without reset.</p>

[26]	RSTCNT	Counter Reset Bit Set this bit will reset the Timer counter, pre-scale and also force CNTEN to 0. 0 = No effect. 1 = Reset Timer's pre-scale counter, internal 16-bit up-counter and CNTEN bit.
[25]	ACTSTS	Timer Active Status Bit (Read Only) This bit indicates the counter status of Timer. 0 = Timer is not active. 1 = Timer is active.
[24:8]	Reserved	Reserved.
[7:0]	PSC	Timer Clock Prescaler Clock input is divided by (PSC+1) before it is fed to the counter. If PSC = 0, then there is no scaling.

Timer Compare Register (TIMERx_CMP)

Register	Offset	R/W	Description	Reset Value
TIMER0_CMP	TMR_BA+0x04	R/W	Timer0 Compare Register	0x0000_0000
TIMER1_CMP	TMR_BA+0x24	R/W	Timer1 Compare Register	0x0000_0000
TIMER2_CMP	TMR_BA+0x44	R/W	Timer2 Compare Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
CMPDAT [15:8]							
7	6	5	4	3	2	1	0
CMPDAT[7:0]							

Bits	Description
[31:16]	Reserved Reserved.
[15:0]	<p>CMPDAT</p> <p>Timer Comparison Value CMPDAT is a 16-bit comparison register. When the 16-bit up-counter is enabled and its value is equal to CMPDAT value, a Timer Interrupt is requested if the timer interrupt is enabled with <code>TIMERx_CTL.INTEN = 1</code>.</p> <p>Note 1: Never set CMPDAT to 0x000 or 0x001. Timer will not function correctly.</p> <p>Note 2: No matter CNTEN is 0 or 1, whenever software writes a new value into this register, TIMER will restart counting by using this new value and abort previous count.</p>



Timer Interrupt Status Register (TIMERx INTSTS)

Register	Offset	R/W	Description	Reset Value
TIMER0_INTSTS	TMR_BA+0x08	R/W	Timer0 Interrupt Status Register	0x0000_0000
TIMER1_INTSTS	TMR_BA+0x28	R/W	Timer1 Interrupt Status Register	0x0000_0000
TIMER2_INTSTS	TMR_BA+0x48	R/W	Timer2 Interrupt Status Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							TIF

Bits	Description	
[31:1]	Reserved	Reserved.
[0]	TIF	<p>Timer Interrupt Flag (Read Only)</p> <p>This bit indicates the interrupt status of Timer.</p> <p>TIF bit is set by hardware when the 16-bit counter matches the timer comparison value (CMPDAT). It is cleared by writing 1 to itself</p> <p>0 = No effect.</p> <p>1 = CNT (TIMERx_CNT[15:0]) value matches the CMPDAT (TIMERx_CMP[15:0]) value.</p>

Timer Data Register (TIMERx_CNT)

Register	Offset	R/W	Description	Reset Value
TIMER0_CNT	TMR_BA+0x0C	R	Timer0 Data Register	0x0000_0000
TIMER1_CNT	TMR_BA+0x2C	R	Timer1 Data Register	0x0000_0000
TIMER2_CNT	TMR_BA+0x4C	R	Timer2 Data Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
CNT[15:8]							
7	6	5	4	3	2	1	0
CNT[7:0]							

Bits	Description	
[31:16]	Reserved	Reserved.
[15:0]	CNT	Timer Data Register User can read this register for the current up-counter value while TIMERx_CTL.CNTEN is set to 1,



TimerF Interrupt Status Register (TIMERF_INTSTS)

Register	Offset	R/W	Description	Reset Value
TIMERF_INTSTS	TMR_BA+0x30	R/W	TimerF Interrupt Status Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved						TFIE	TFIF

Bits	Description	
[31:2]	Reserved	Reserved.
[1]	TFIE	TimerF Interrupt Enable 0 = Disable TimerF Interrupt. 1 = Enable TimerF Interrupt.
[0]	TFIF	TimerF Interrupt Flag This bit indicates the interrupt status of TimerF. TFIF bit is set by hardware when TimerF time out. It is cleared by writing 1 to this bit. 0 = It indicates that TimerF does not time out yet. 1 = It indicates that TimerF time out. The interrupt flag is set if TimerF interrupt was enabled.

IR Carrier Output Control Register (IR_CTL)

Register	Offset	R/W	Description	Reset Value
IR_CTL	TMR_BA+0x34	R/W	IR Carrier Output Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved						IRCEM	NONCS

Timer1 time-out signal is used to toggle IROUT output. Before IR carrier output is enabled, user needs to setup Timer1 according to output frequency of IR carrier.

Bits	Descriptions	
[31:2]	Reserved	Reserved
[1]	IRCEM	IR carrier output enable 0 = Disable IR carrier output, 1 = Enable IR carrier output. Timer1 time out will toggle the output state on IROUT pin.
[0]	NONCS	Non-carrier state 0 = IROUT keeps low when IRCEM is 0, 1 = IROUT keeps high when IRCEM is 0.

5.11 Watchdog Timer

The purpose of Watchdog Timer is to perform a system reset if system runs into an unknown state. This prevents system from hanging for an indefinite period of time. The watchdog timer includes a 19-bit free running counter with programmable time-out intervals.

Setting WDT_CTL[7](WDTEN) enables the watchdog timer and the WDT counter starts counting up.

When the counter reaches the selected time-out interval, Watchdog timer interrupt flag WDT_CTL[3](IF) will be set immediately to request a WDT interrupt if the watchdog timer interrupt enable bit WDT_CTL[6](INTEN) is set, in the meantime, a specified delay time follows the time-out event. User must set WDT_CTL[0](RSTCNT) (Watchdog timer reset) high to reset the 19-bit WDT counter to prevent Watchdog timer reset before the delay time expires. WDT_CTL[0](RSTCNT) bit is auto cleared by hardware after WDT counter is reset. There are eight time-out intervals with specific delay time which are selected by Watchdog timer interval select bits WDT_CTL[10:8](TOUTSEL). If the WDT counter has not been cleared after the specific delay time expires, the watchdog timer will set Watchdog Timer Reset Flag WDT_CTL[2](RSTF) high and reset CPU. This reset will last 64 WDT clocks then CPU restarts executing program from reset vector (0x0000 0000). WDT_CTL[2](RSTF) will not be cleared by Watchdog reset. User may poll WDT_CTL[2](RSTF) by software to recognize the reset source.

Table 5-5 Watchdog Timeout Interval Selection

TOUTSEL	WDT Interrupt Timeout	WDT Reset Timeout	WDT Reset Timeout Interval (WDT_CLK=12 MHz)	WDT Reset Timeout Interval (WDT_CLK=32 KHz)
000	2^4 WDT_CLK	$(2^4 + 1024)$ WDT_CLK	69.33 μ s	32.5 ms
001	2^6 WDT_CLK	$(2^6 + 1024)$ WDT_CLK	72.53 μ s	34 ms
010	2^8 WDT_CLK	$(2^8 + 1024)$ WDT_CLK	85.33 μ s	40 ms
011	2^{10} WDT_CLK	$(2^{10} + 1024)$ WDT_CLK	170.67 μ s	64 ms
100	2^{12} WDT_CLK	$(2^{12} + 1024)$ WDT_CLK	426.67 μ s	160 ms
101	2^{14} WDT_CLK	$(2^{14} + 1024)$ WDT_CLK	1.45 ms	544 ms
110	2^{16} WDT_CLK	$(2^{16} + 1024)$ WDT_CLK	5.55 ms	2080 ms
111	2^{18} WDT_CLK	$(2^{18} + 1024)$ WDT_CLK	21.93 ms	8224 ms

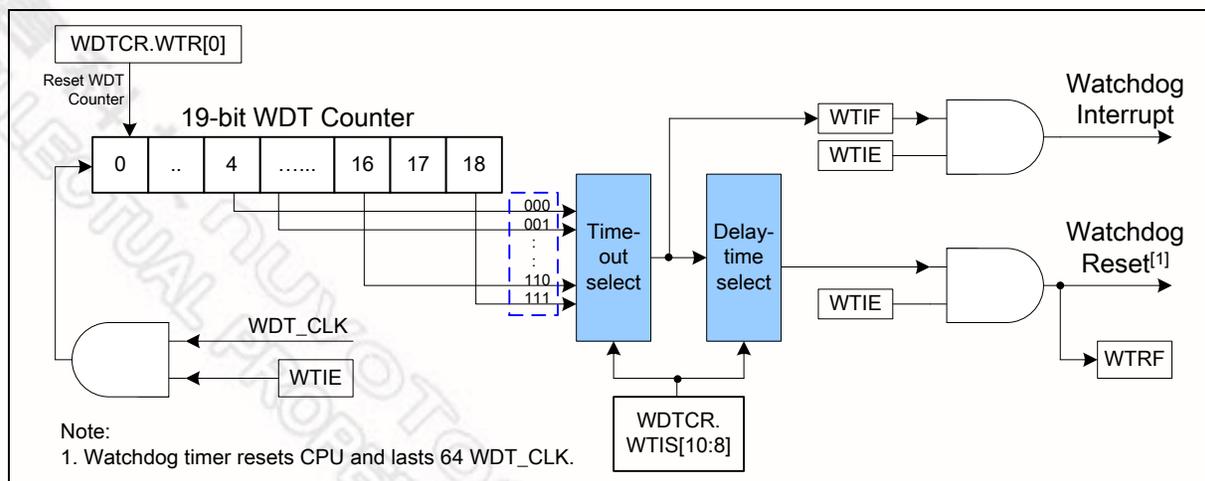


Figure 5-22 Watchdog Timer Block Diagram



5.11.1 Watchdog Timer Control Register Map

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
WDT Base Address: WDT_BA = 0x4000_4000				
WDT_CTL	WDT_BA+0x00	R/W	Watchdog Timer Control Register	0x0000_0700

5.11.2 Watchdog Timer Control Register Description

Watchdog Timer Control Register (WDT_CTL)

This is a protected register, to write to register, first issue the unlock sequence ([see Register Lock Control Register \(SYS_REGLCTL\)](#)). Only flag bits IF and RSTF are unprotected and can be write-cleared at any time.

Register	Offset	R/W	Description	Reset Value
WDT_CTL	WDT_BA+0x00	R/W	Watchdog Timer Control Register	0x0000_0700

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved					TOUTSEL		
7	6	5	4	3	2	1	0
WDTEN	INTEN	WKF	WKEN	IF	RSTF	RSTEN	RSTCNT

Bits	Description
[31:11]	Reserved
	Reserved.

[10:8]	TOUTSEL	<p>Watchdog Timer Interval Select</p> <p>These three bits select the timeout interval for the Watchdog timer, a watchdog reset will occur 1024 clock cycles later if Watchdog timer is not reset.</p> <p>The WDT interrupt timeout is given by:</p> <p>000 = $2^4 * \text{WDT_CLK}$. 001 = $2^6 * \text{WDT_CLK}$. 010 = $2^8 * \text{WDT_CLK}$. 011 = $2^{10} * \text{WDT_CLK}$. 100 = $2^{12} * \text{WDT_CLK}$. 101 = $2^{14} * \text{WDT_CLK}$. 110 = $2^{16} * \text{WDT_CLK}$. 111 = $2^{18} * \text{WDT_CLK}$.</p> <p>WDT reset timeout = (WDT interrupt timeout + 1024) * WDT_CLK. Where WDT_CLK is the period of the Watchdog Timer clock source.</p>
[7]	WDTEN	<p>Watchdog Timer Enable</p> <p>0 = Disable the WDT(Watchdog timer) (This action will reset the internal counter). 1 = Enable the WDT(Watchdog timer).</p>
[6]	INTEN	<p>Watchdog Time-Out Interrupt Enable</p> <p>0 = Disable the WDT time-out interrupt. 1 = Enable the WDT time-out interrupt.</p>
[5]	WKF	<p>WDT Time-Out Wake-Up Flag</p> <p>If WDT causes CPU wake up from sleep or power-down mode, this bit will be set to high.</p> <p>0 = WDT does not cause CPU wake-up. 1 = CPU wakes up from sleep or power-down mode by WDT time-out interrupt.</p> <p>Note: This bit is cleared by writing 1 to it.</p>
[4]	WKEN	<p>WDT Time-Out Wake-Up Function Control</p> <p>If this bit is set to 1, while WDT time-out interrupt flag IF (WDT_CTL[3]) is generated to 1 and interrupt enable bit INTEN (WDT_CTL[6]) is enabled, the WDT time-out interrupt signal will generate a wake-up trigger event to chip.</p> <p>0 = Enable the Wakeup function that WDT timeout can wake up CPU from power-down mode. 1 = Disable WDT Wakeup CPU function.</p>
[3]	IF	<p>Watchdog Timer Interrupt Flag</p> <p>If the Watchdog timer interrupt is enabled, then the hardware will set this bit to indicate that the Watchdog timer interrupt has occurred. If the Watchdog timer interrupt is not enabled, then this bit indicates that a timeout period has elapsed.</p> <p>0 = Watchdog timer interrupt has not occurred. 1 = Watchdog timer interrupt has occurred.</p> <p>Note: This bit is cleared by writing 1 to this bit.</p>
[2]	RSTF	<p>Watchdog Timer Reset Flag</p> <p>When the Watchdog timer initiates a reset, the hardware will set this bit. This flag can be read by software to determine the source of reset. Software is responsible to clear it manually by writing 1 to it. If RSTEN is disabled, then the Watchdog timer has no effect on this bit.</p> <p>0 = Watchdog timer reset has not occurred. 1 = Watchdog timer reset has occurred.</p> <p>Note: This bit is cleared by writing 1 to this bit.</p>

[1]	RSTEN	<p>Watchdog Timer Reset Enable</p> <p>Setting this bit will enable the Watchdog timer reset function.</p> <p>0 = Disable Watchdog timer reset function.</p> <p>1 = Enable Watchdog timer reset function.</p> <p>Note: This function cannot work with XTL32-based clock source.</p>
[0]	RSTCNT	<p>Clear Watchdog Timer</p> <p>Set this bit will clear the Watchdog timer.</p> <p>0 = Writing 0 to this bit has no effect.</p> <p>1 = Reset the contents of the Watchdog timer.</p> <p>Note: This bit will auto clear after few clock cycles</p>

5.12 Audio Processing Unit (APU)

The main purpose of Audio Processing Unit (APU) is used to playback the audio data (PCM format) which are decoded and stored in local buffer by CPU. The APU builds in a H/W mixer and a digital to analog convertor (DAC) with 13-bit resolution, and a Power Amplifier (PA) with 7-level volume adjustment.

There are two channels input for the H/W mixer. H/W Mixer mixes inputs and clips the mixed output to 13 bits.

If DAC is enabled, Timer0 is used as the channel 0 sample rate control timer automatically. User needs to program the time-out rate of Timer0 as the audio data sample rate.

There is an 8-level buffer for channel 0 audio data buffer. Whenever Timer0 time-out, one sample is read out from buffer sequentially to DAC. Once the data in the defined threshold of buffer is read out, a threshold interrupt is generated if its corresponding interrupt is enabled.

There is only one level buffer for channel 1 audio data buffer. No Timer H/W is associated with channel 1.

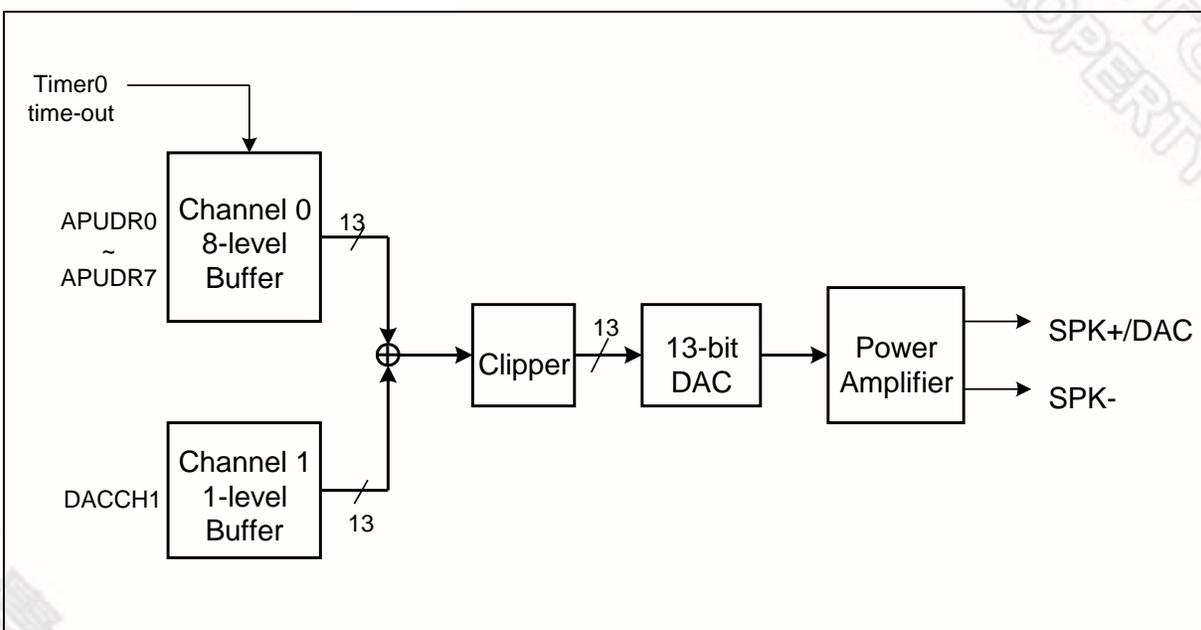


Figure 5-23 APU Block Diagram



5.12.1 APU Control Register Map

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
APU Base Address: APU_BA = 0x400C_0000				
APU_CTL	APU_BA+0x00	R/W	APU Control Register	0x0000_0000
APU_VM	APU_BA+0x04	R/W	APU Volume Control Register	0x0000_0000
APU_CH1DAT0	APU_BA+0x0C	R/W	APU Channel 1 Data Buffer Register	0x0000_0000
APU_CH0DAT0	APU_BA+0x10	R/W	APU Channel 0 Data Buffer Register 0	0x0000_0000
APU_CH0DAT1	APU_BA+0x14	R/W	APU Channel 0 Data Buffer Register 1	0x0000_0000
APU_CH0DAT2	APU_BA+0x18	R/W	APU Channel 0 Data Buffer Register 2	0x0000_0000
APU_CH0DAT3	APU_BA+0x1C	R/W	APU Channel 0 Data Buffer Register 3	0x0000_0000
APU_CH0DAT4	APU_BA+0x20	R/W	APU Channel 0 Data Buffer Register 4	0x0000_0000
APU_CH0DAT5	APU_BA+0x24	R/W	APU Channel 0 Data Buffer Register 5	0x0000_0000
APU_CH0DAT6	APU_BA+0x28	R/W	APU Channel 0 Data Buffer Register 6	0x0000_0000
APU_CH0DAT7	APU_BA+0x2C	R/W	APU Channel 0 Data Buffer Register 7	0x0000_0000



5.12.2 APU Control Register Description

APU Control Register (APU_CTL)

Register	Offset	R/W	Description	Reset Value
APU_CTL	APU_BA+0x00	R/W	APU Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved		DACGN	Reserved			BPPAM	DACE
7	6	5	4	3	2	1	0
PAMPE	APUIE	APUIS	Reserved		TSHD		

Bits	Description	
[31:14]	Reserved	Reserved.
[13]	DACGN	DAC Output Current Control 0 = 3mA. 1 = 5mA. This bit is effective only when BPPAM is "1".
[12:10]	Reserved	Reserved.
[9]	BPPAM	Bypass Power Amplifier, DAC Output To Pin 0 = SPK+ and SPK- are power amplifier outputs. 1 = No output at SPK-. This setting is for testing only and prohibited for normal operation. Note: User must set BPPAM to "0" to use SPK+ and SPK- as the power amplifier outputs.
[8]	DACE	DAC Enable 0 = Disable DAC function. 1 = Enable DAC function.
[7]	PAMPE	Power Amplifier Enable 0 = Disable PA function. 1 = Enable PA function.
[6]	APUIE	APU Interrupt Enable 0 = Disable the APU threshold interrupt. 1 = Enable the APU threshold interrupt.

[5]	APUIS	APU Interrupt Status 0 = APU threshold interrupt does not occur. 1 = APU threshold interrupt occur. This flag is set by hardware when APU threshold is met. Software can clear this bit by writing a zero to it.
[4:3]	Reserved	Reserved.
[2:0]	TSHD	APU Interrupt Threshold 000 = Buffer 0 is read out by APU. 001 = Buffer 1 is read out by APU. 010 = Buffer 2 is read out by APU. 011 = Buffer 3 is read out by APU. 100 = Buffer 4 is read out by APU. 101 = Buffer 5 is read out by APU. 110 = Buffer 6 is read out by APU. 111 = Buffer 7 is read out by APU.



APU Volume Control Register (APU_VM)

Register	Offset	R/W	Description	Reset Value
APU_VM	APU_BA+0x04	R/W	APU Volume Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved					VOLUM		

Bits	Description	
[31:3]	Reserved	Reserved.
[2:0]	VOLUM	APU Volume Adjustment 000 = 0 dB. 001 = -3 dB. 010 = -6 dB. 011 = -9 dB. 100 = -12 dB. 101 = -15 dB. 110 = -18 dB. 111 = Reserved.



APU Channel 1 Data Register (APU_CH1DAT0)

Register	Offset	R/W	Description	Reset Value
APU_CH1DAT0	APU_BA+0x0C	R/W	APU Channel 1 Data Buffer Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved				PCM1[12:8]			
7	6	5	4	3	2	1	0
PCM1[7:0]							

Bits	Description	
[31:13]	Reserved	Reserved.
[12:0]	PCM1	PCM Data Of Channel 1 This field contains 13 bits PCM data that is one of the mixer input. The data format of PCM is 2's complement.



APU Channel 0 Data Register (APU_CH0DATn)

Register	Offset	R/W	Description	Reset Value
APU_CH0DAT0	APU_BA+0x10	R/W	APU Channel 0 Data Buffer Register 0	0x0000_0000
APU_CH0DAT1	APU_BA+0x14	R/W	APU Channel 0 Data Buffer Register 1	0x0000_0000
APU_CH0DAT2	APU_BA+0x18	R/W	APU Channel 0 Data Buffer Register 2	0x0000_0000
APU_CH0DAT3	APU_BA+0x1C	R/W	APU Channel 0 Data Buffer Register 3	0x0000_0000
APU_CH0DAT4	APU_BA+0x20	R/W	APU Channel 0 Data Buffer Register 4	0x0000_0000
APU_CH0DAT5	APU_BA+0x24	R/W	APU Channel 0 Data Buffer Register 5	0x0000_0000
APU_CH0DAT6	APU_BA+0x28	R/W	APU Channel 0 Data Buffer Register 6	0x0000_0000
APU_CH0DAT7	APU_BA+0x2C	R/W	APU Channel 0 Data Buffer Register 7	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved				PCM[12:8]			
7	6	5	4	3	2	1	0
PCM[7:0]							

Bits	Description	
[31:13]	Reserved	Reserved.
[12:0]	PCM	PCM Data Of Channel 0 This field contains 13 bits PCM data that is one of the mixer input. The data format of PCM is 2' complement.

5.13 12-bit Analog-to-Digital Converter (ADC) and Pre-amplifier

N572F065 contains one 12-bit successive approximation analog-to-digital converters (ADC). ADC can be programmed operation independently. ADC has 8-channel inputs. The A/D converter supports three operation modes: single, single-cycle scan and continuous scan mode. There are two kinds of scan mode: continuous mode and single cycle mode. Each of A/D converters can be started by software and external STADC pin.

Note: The PCLK is equivalent to HCLK thereafter.

5.13.1 Features

- Analog input voltage range: 0~5.0V
- 12-bits resolution and 10-bits accuracy is guaranteed
- Total 8 analog inputs: ADC has 8-channel inputs
- Maximum ADC clock frequency is 5MHz
- Up to 200K SPS conversion rate
- Three operating modes
 - Single mode: Single channel A/D conversion
 - Single-cycle scan mode: Continuous conversion on enabled channels
 - Continuous scan mode: Repetitive conversion on enabled channels
- An A/D conversion can be started by
 - Software write SWTRG bit
 - External pin STADC
- Conversion results are held in data registers for each channel with valid and overrun indicators
- Conversion result can be compared with specified value and user can select whether to generate an interrupt when conversion result is equal to the compare register setting.
- Support single-end
- Self-calibration function to minimum conversion error

5.13.2 ADC Block Diagram

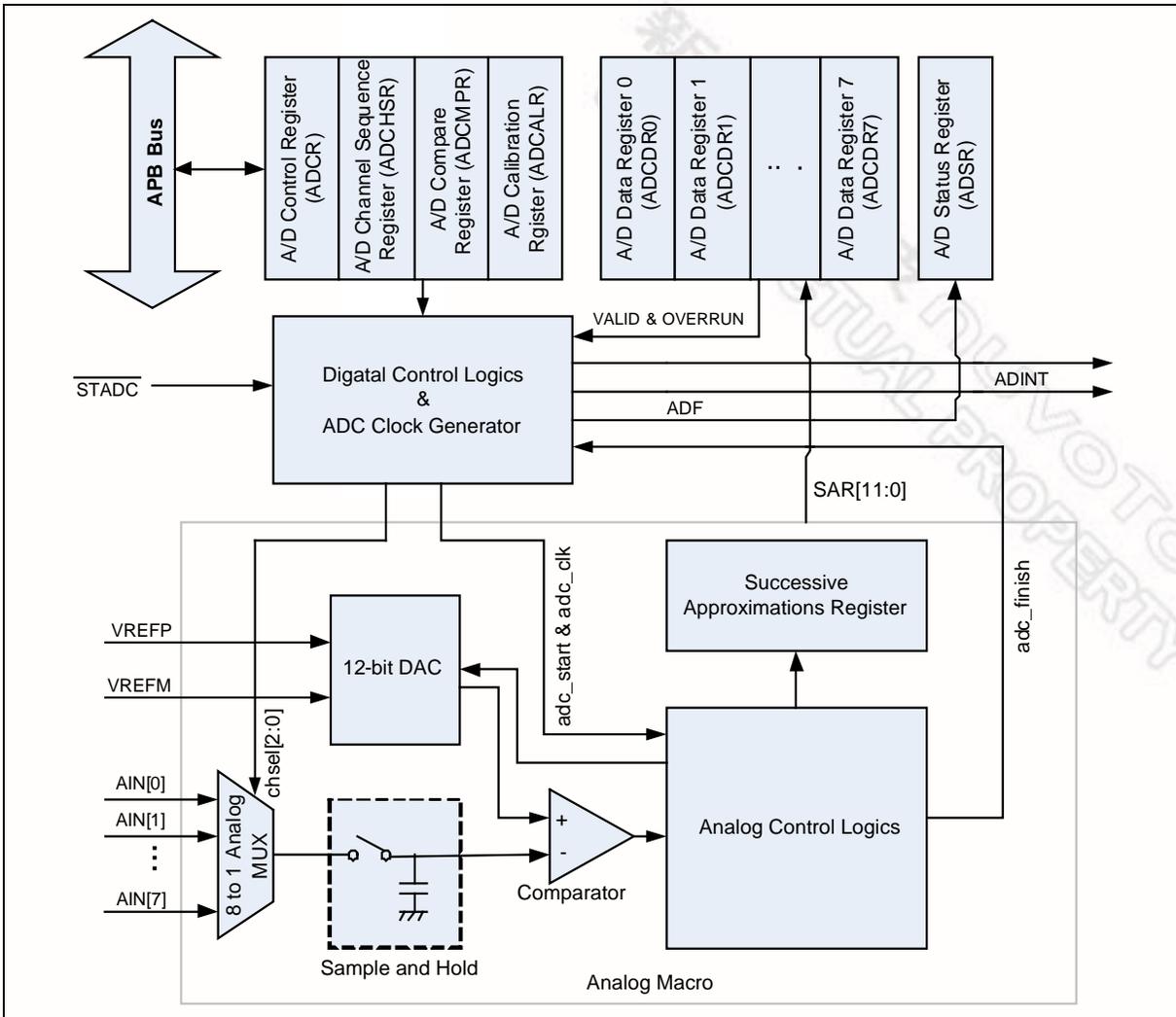


Figure 5-24 ADC Controller Block Diagram

5.13.3 ADC Operation Procedure

The A/D converter operates by successive approximation with 12-bit resolution. This A/D converter equips with self-calibration function to minimum conversion error, user can write 1 to CALEN bit in ADC_CAL register to enable calibration function, while internal calibration is finish the CAL_DONE bit will assert. The ADCs have two operation modes: single mode and scan mode. There are two kinds of scan mode: continuous mode and single-cycle mode. When changing the operating mode or analog input channel enable, in order to prevent incorrect operation, software must clear SWTRG bit to 0 in ADC_CTL register. The A/D converter discards current conversion immediately and enters idle state while SWTRG bit is cleared.

5.13.3.1 Self-Calibration

User can write 1 to CALEN bit in ADC_CAL register to enable self-calibration. The operation is process internally and it needs 127 ADC clocks to complete calibration. After CALEN is set to 1, software must wait CAL_DONE bit set by internal hardware. The detail timing is shown as below:

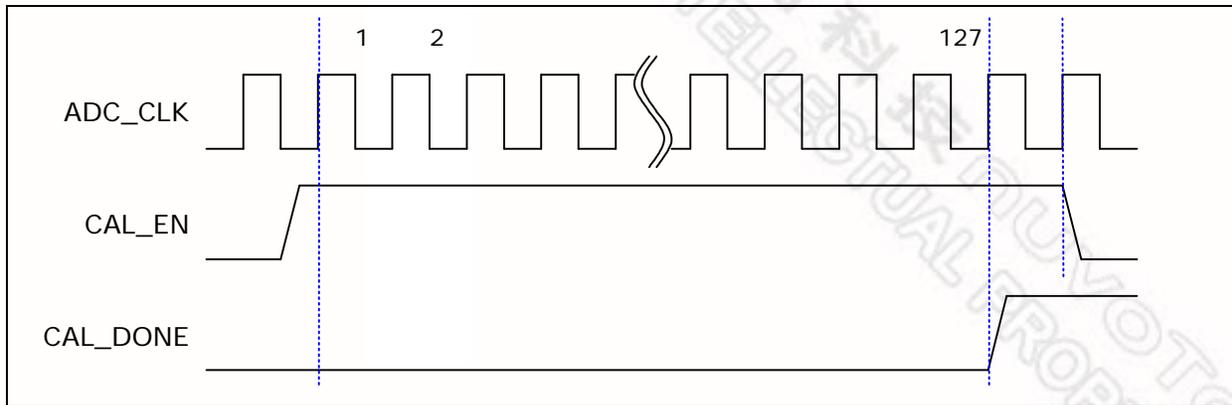


Figure 5-25 ADC Converter Self-Calibration Timing Diagram

5.13.3.2 ADC Clock Generator

The maximum sampling rate is up to 200K SPS. It needs 25 ADC clocks to complete an A/D conversion. The maximum clock rate to A/D conversion module should be less than 25 x sampling rate. In this ADC controller, APB bus clock (PCLK) is divided by (ADC_N+1) to produce the clock to A/D converter which should be less than or equal to 5MHz. In generally, software can set these bits to be 5MHz or slightly less.

5.13.3.3 Single Mode

In single mode, A/D conversion is to be performed only once on the specified single channel. The operations are as follows:

1. The channel defined in CHSEQ0 is the channel to be converted.
2. A/D conversion is started when the SWTRG bit in ADC_CTL is set to 1 by software or external trigger input.
3. When A/D conversion is finished, the result is transferred to the A/D data register ADC_DATn corresponding to the channel.
4. On completion of conversion, the ADIF bit in ADC_STATUS is set to 1. If the ADCIE bit is 1, an ADINT interrupt request is generated.
5. The SWTRG bit remains "1" during A/D conversion. When A/D conversion ends, the SWTRG bit is automatically cleared to 0 and the A/D converter enters the idle state.
6. When the SWTRG bit is cleared to 0 during A/D conversion, A/D conversion stops and the A/D converter enters the idle state.

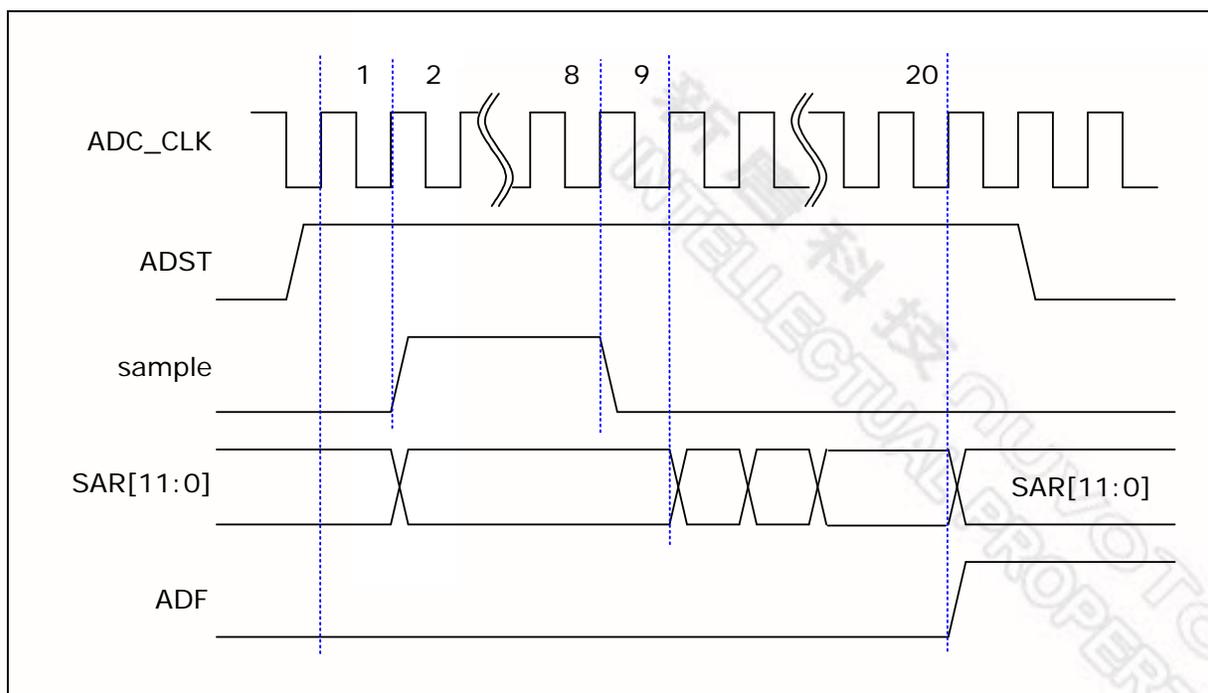


Figure 5-26 Single Mode Conversion Timing Diagram

5.13.3.4 Continuous Scan Mode

In continuous scan mode, A/D conversion is to be performed sequentially on the specified channels that are defined by CHSEQx bits in ADC_CHSEQ register (8 channels maximum). The operations are as follows:

1. When the SWTRG bit in ADC_CTL is set to 1 by software or external trigger input, A/D conversion starts on the channel with the lowest number.
2. When A/D conversion for each channel selected by channel sequence register (ADC_CHSEQ) is completed, the converted result is sequentially transferred to the A/D data register corresponding to channel sequence.
3. When the conversion for all the selected channels is completed, the ADIF bit in ADC_STATUS is set to 1. If the ADCIE bit is 1, an ADINT interrupt is requested after A/D conversion ends. Conversion of the channel defined in CHSEQ0 starts again.
4. Steps 2 to 3 are repeated as long as the SWTRG bit remains to 1. When the SWTRG bit is cleared to 0, A/D conversion stops and the A/D converter enters the idle state.

An example timing diagram for continuous scan mode on 3 channels (CHSEQ0=0, CHSEQ1=3, CHSEQ2=0, CHSEQ3=6 and CHSEQ4=F) is shown as below:

(This example is only appropriate for ADC)

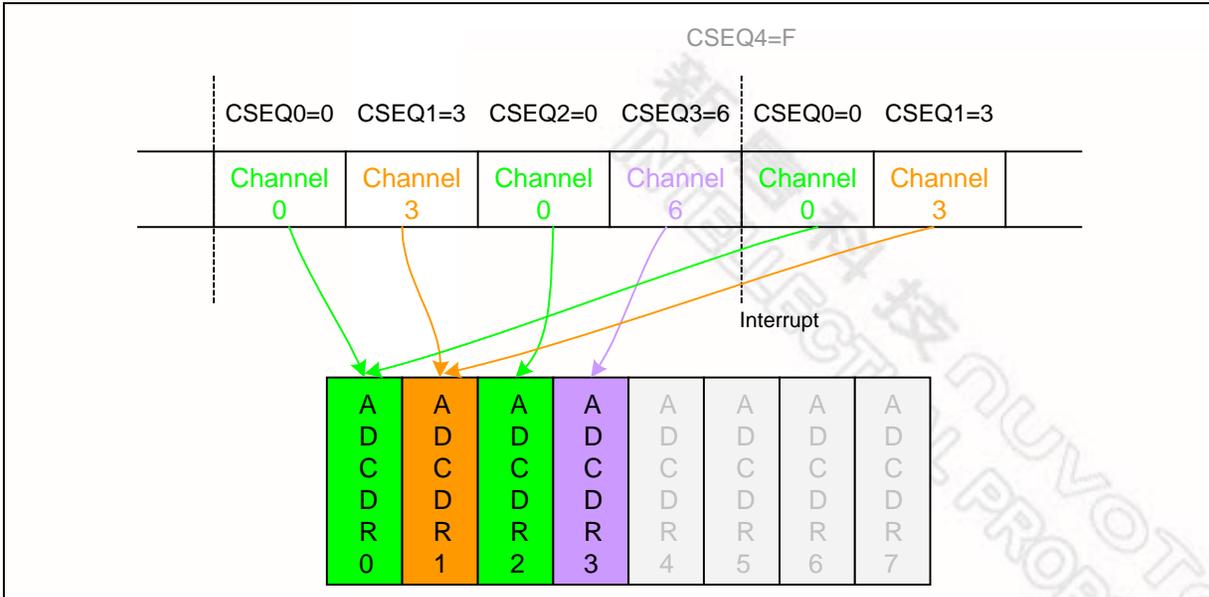


Figure 5-27 Continuous Scan on Selected Channels

5.13.3.5 Single-Cycle Scan Mode

In single-cycle scan mode, A/D conversion is to be performed once on the specified channels that are defined by CHSEQx bits in ADC_CHSEQ register (8 channels maximum for ADC). Operations are as follows:

1. When the SWTRG bit in ADC_CTL is set to 1 by software, or external trigger input, A/D conversion starts on the channel selected by CHSEQ0.
2. When A/D conversion for channel selected by channel sequence registers (CHSEQx) is completed, the result is sequentially transferred to the A/D data register corresponding to channel sequence.
3. When the conversion for all the selected channels is completed, the ADIF bit in ADC_STATUS is set to 1. If the ADCIE bit is 1, an ADINT interrupt is requested after A/D conversion ends.
4. After A/D conversion ends, the SWTRG bit is automatically cleared to 0 and the A/D converter enters the idle state. When the SWTRG bit is cleared to 0 during A/D conversion, A/D conversion stops and the A/D converter enters the idle state.

An example timing diagram for single-cycle scan on 4 channels (CHSEQ0=0, CHSEQ1=2, CHSEQ2=3, CHSEQ3=7 and CHSEQ4=F) is shown as below:

(This example is only appropriate for ADC)

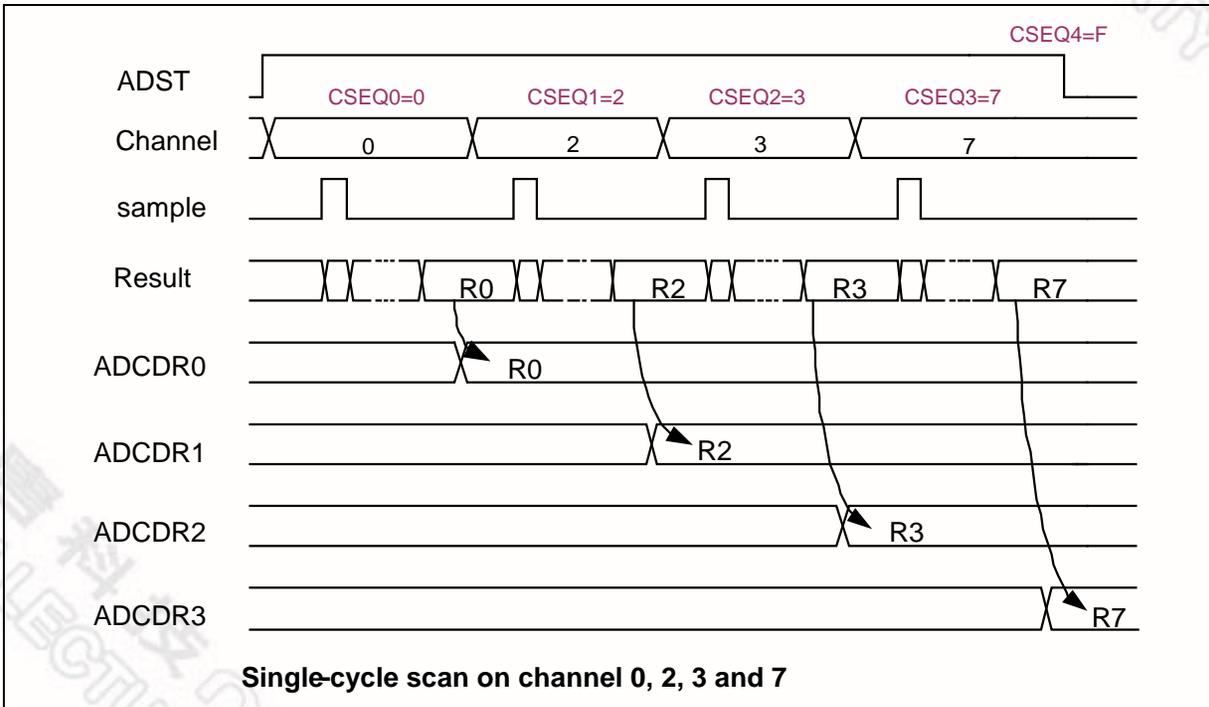


Figure 5-28 Single-Cycle Scan on selected Channels

5.13.3.6 Input Sampling and A/D Conversion Time

The A/D converter has a built-in sample-and-hold circuit for each module, the A/D samples the analog input when A/D conversion start delay time (T_d) has passed after SWTRG bit in ADC_CTL is set to 1, then start conversion. Due to ADC clock is generated by PCLK divided by (N+1), the maximum delay time from APB write to A/D start sampling analog input time is 2N PCLKs. The start delay time is shown as below:

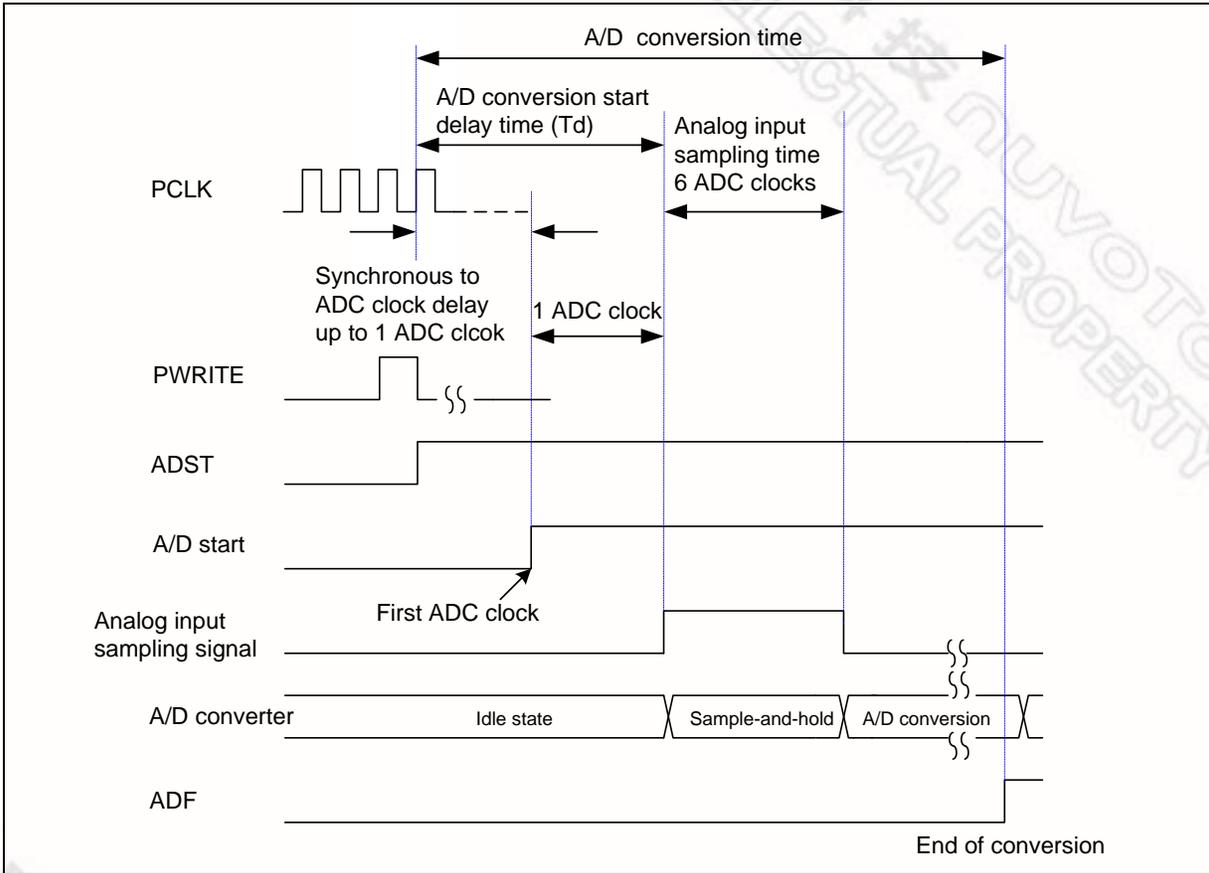


Figure 5-29 Conversion Start Delay Timing Diagram

A/D conversion can be triggered by external pin request. External trigger input is enabled at the STADC pin. Software can set HWTRGCOND to select trigger condition is falling/rising edge or low/high level. An 8-bit sampling counter is used to deglitch. If level trigger condition is selected, the STADC pin must be kept at defined state at least 8 PCLKs. The SWTRG bit will be set to 1 at the 9th PCLK and start to conversion. Conversion is continuous if external trigger input is pull at low (or high state) in level trigger mode. It is stopped only when external trigger condition disappears. If edge trigger condition is selected, the high and low state must be kept at least 4 PLCKs. Pulse that is shorter than this specification will be ignored. Other operations, in both single and scan modes, are the same as when the SWTRG bit has been set to 1.

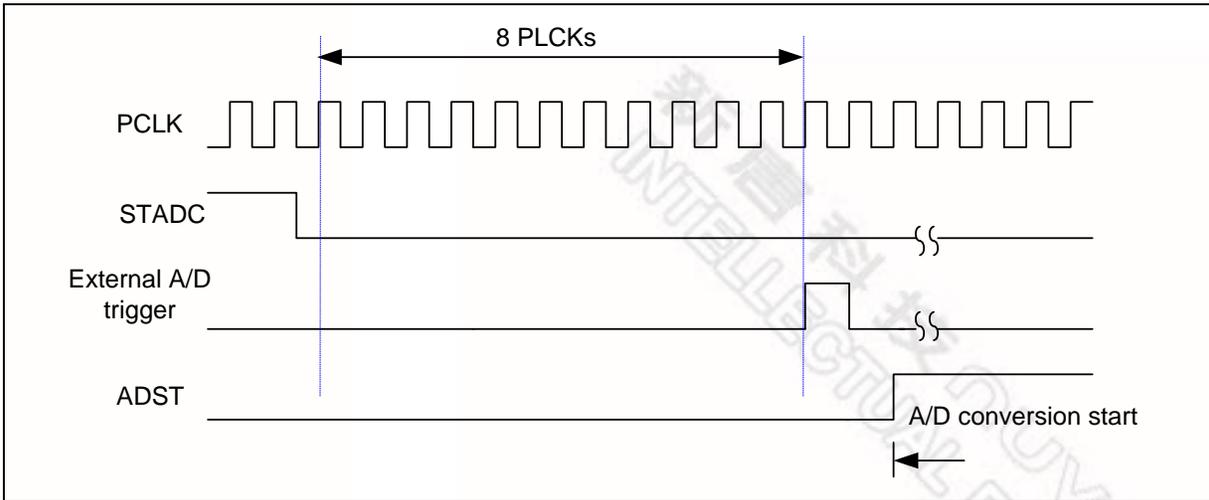


Figure 5-30 External A/D Conversion Trigger Timing Diagram

5.13.3.7 Conversion Result Monitor

N572F065 ADC controller provides two compare registers ADC_CMP0/1 to monitor specified channel conversion result from A/D conversion module (see figure below). Software can select which channel to be monitored by set CMPCH and CMPCOND bit is used to check conversion result is less than specify value or greater than (equal to) value specified in CMPDAT. When the compared result meets the setting, compare match counter will increase 1. When counter value reaches the setting of CMPMCNT then ADCMPF bit will be set to 1. If ADCMPIE bit is set, then an ADINT interrupt request is generated. The block diagram is shown as below:

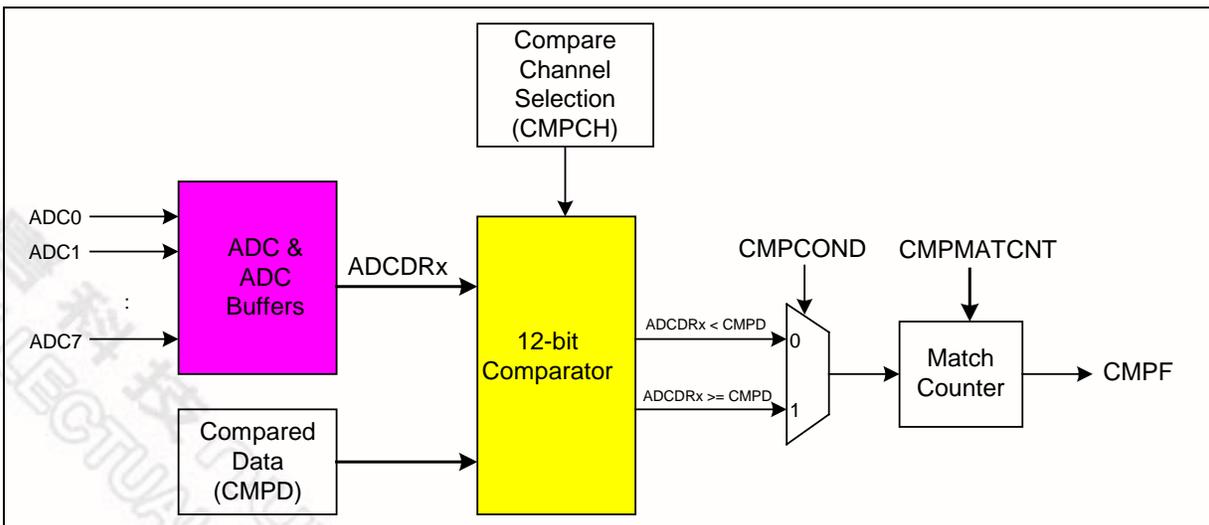


Figure 5-31 A/D Conversion Result Comparison

5.13.3.8 Interrupt Sources

The A/D converter generates a conversion end ADIF in ADC_STATUS register upon the end of A/D conversion. If ADCIE bit in ADC_CTL is set then conversion end interrupt request ADINT is generated. If ADCMPn bit is enabled, when A/D conversion result meets setting in ADC_CMPn register, monitor interrupt is generated, ADINT will be set also. CPU can clear ADCMPF and ADIF to stop interrupt request.

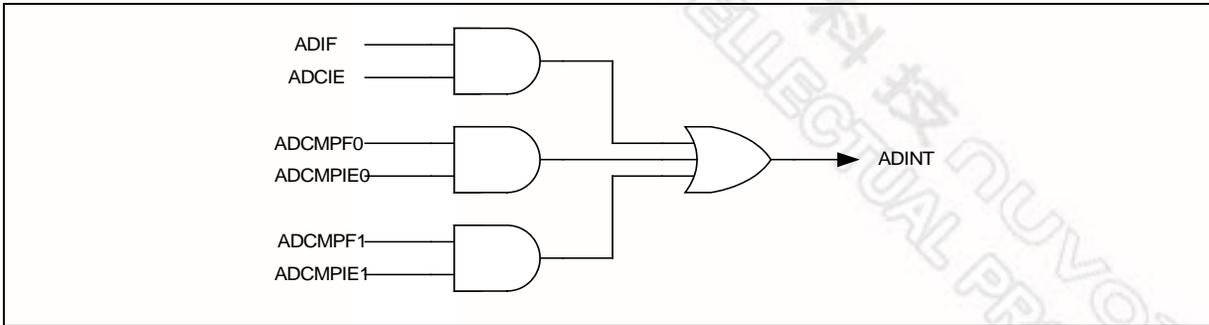


Figure 5-32 A/D Controller Interrupt



5.13.4 ADC and Pre-amplifier Register Map

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
ADC Base Address: ADC_BA = 0x400E_0000				
ADC_DAT0	ADC_BA+0x00	R	A/D Data Register for the channel defined in CHSEQ0	0x0000_0000
ADC_DAT1	ADC_BA+0x04	R	A/D Data Register for the channel defined in CHSEQ1	0x0000_0000
ADC_DAT2	ADC_BA+0x08	R	A/D Data Register for the channel defined in CHSEQ2	0x0000_0000
ADC_DAT3	ADC_BA+0x0C	R	A/D Data Register for the channel defined in CHSEQ3	0x0000_0000
ADC_DAT4	ADC_BA+0x10	R	A/D Data Register for the channel defined in CHSEQ4	0x0000_0000
ADC_DAT5	ADC_BA+0x14	R	A/D Data Register for the channel defined in CHSEQ5	0x0000_0000
ADC_DAT6	ADC_BA+0x18	R	A/D Data Register for the channel defined in CHSEQ6	0x0000_0000
ADC_DAT7	ADC_BA+0x1C	R	A/D Data Register for the channel defined in CHSEQ7	0x0000_0000
ADC_CTL	ADC_BA+0x20	R/W	A/D Control Register	0x0000_0000
ADC_CHSEQ	ADC_BA+0x24	R/W	A/D Channel Sequence Register	0xFFFF_FFFF
ADC_CMP0	ADC_BA+0x28	R/W	A/D Compare Register 0	0x0000_0000
ADC_CMP1	ADC_BA+0x2C	R/W	A/D Compare Register 1	0x0000_0000
ADC_STATUS	ADC_BA+0x30	R/W	A/D Status Register	0x0000_0000
ADC_CAL	ADC_BA+0x34	R/W	A/D Calibration Register	0x0000_0000
ADC_PGCTL	ADC_BA+0x3C	R/W	ADC Pre-amplifier Gain Control Register	0x0000_0000



5.13.5 ADC and Pre-amplifier Register Description

A/D Data Registers (ADC_DATn)

Register	Offset	R/W	Description	Reset Value
ADC_DAT0	ADC_BA+0x00	R	A/D Data Register for the channel defined in CHSEQ0	0x0000_0000
ADC_DAT1	ADC_BA+0x04	R	A/D Data Register for the channel defined in CHSEQ1	0x0000_0000
ADC_DAT2	ADC_BA+0x08	R	A/D Data Register for the channel defined in CHSEQ2	0x0000_0000
ADC_DAT3	ADC_BA+0x0C	R	A/D Data Register for the channel defined in CHSEQ3	0x0000_0000
ADC_DAT4	ADC_BA+0x10	R	A/D Data Register for the channel defined in CHSEQ4	0x0000_0000
ADC_DAT5	ADC_BA+0x14	R	A/D Data Register for the channel defined in CHSEQ5	0x0000_0000
ADC_DAT6	ADC_BA+0x18	R	A/D Data Register for the channel defined in CHSEQ6	0x0000_0000
ADC_DAT7	ADC_BA+0x1C	R	A/D Data Register for the channel defined in CHSEQ7	0x0000_0000

The A/D converted results are saved in these ADC_DATn registers sequentially. The channel scan sequence is defined in ADC_HSR register.

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved						VALID	OV
15	14	13	12	11	10	9	8
EXTS				RESULT[11:8]			
7	6	5	4	3	2	1	0
RESULT[7:0]							

Bits	Description	
[31:18]	Reserved	Reserved.
[17]	VALID	<p>Valid Flag</p> <p>0 = Data in RESULT are not valid.</p> <p>1 = Data in RESULT are valid.</p> <p>This bit is set to 1 when corresponding channel analog input conversion is completed and cleared by hardware after ADC_DAT register is read.</p>

NuVoice™ N572F065 Technical Reference Manual

[16]	OV	<p>Over Run Flag</p> <p>0 = Data in RESULT are recent conversion result. 1 = Data in RESULT are overwritten.</p> <p>If converted data in RESULT[11:0] have not been read before new conversion result is loaded to this register, OV is set to 1. It is cleared by hardware after ADC_DAT register is read.</p>
[15:12]	EXTS	<p>Extension Bits Of RESULT for Different Data Format</p> <p>If ADCFM is "0", EXTS all are read as "0". If ADCFM is "1", EXTS all are read as bit RESULT[11].</p>
[11:0]	RESULT	<p>A/D Conversion Result</p> <p>This field contains the 12-bit conversion result. Its data format is defined by ADCFM bit.</p>



A/D Control Register (ADC_CTL)

Register	Offset	R/W	Description	Reset Value
ADC_CTL	ADC_BA+0x20	R/W	A/D Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved			ADCFM	SWTRG	Reserved		HWTRGEN
7	6	5	4	3	2	1	0
HWTRGCOND		Reserved		OPMODE		ADCIE	ADCEN

Bits	Description	
[31:13]	Reserved	Reserved.
[12]	ADCFM	Data Format Of ADC Conversion Result 0 = Unsigned 1 = 2'Complement
[11]	SWTRG	A/D Conversion Start 0 = Conversion is stopped and A/D converter enters idle state. 1 = Start conversion. Note: SWTRG bit can be set to 1 from three sources: software write and external pin STADC. SWTRG is cleared to 0 by hardware automatically at the end of single mode and single-cycle scan mode on specified channel. In continuous scan mode, A/D conversion is continuously performed sequentially until software writes 0 to this bit or chip resets.
[10:9]	Reserved	Reserved.
[8]	HWTRGEN	Trigger Enable Enable or disable triggering of A/D conversion by external STADC pin. 0 = Disable. 1 = Enable.
[7:6]	HWTRGCOND	Trigger Condition These two bits decide external pin STADC trigger event is level or edge. The signal must be kept at stable state at least 8 PCLKs for level trigger and 4 PCLKs at high and low state. 00 = Low level 01 = High level 10 = Falling edge 11 = Positive edge

NuVoice™ N572F065 Technical Reference Manual



[5:4]	Reserved	Reserved.
[3:2]	OPMODE	A/D Converter Operation Mode 00 = Single conversion 01 = Reserved 10 = Single-cycle scan 11 = Continuous scan When changing the operation mode, software should disable SWTRG bit firstly.
[1]	ADCIE	A/D Interrupt Enable 0 = Disable A/D interrupt function 1 = Enable A/D interrupt function A/D conversion end interrupt request is generated if ADCIE bit is set to 1.
[0]	ADCEN	A/D Converter Enable 0 = Disable 1 = Enable Before starting A/D conversion function, this bit should be set to 1. Clear it to 0 to disable A/D converter analog circuit power consumption.

A/D Channel Sequence Register (ADC CHSEQ)

Register	Offset	R/W	Description	Reset Value
ADC_CHSEQ	ADC_BA+0x24	R/W	A/D Channel Sequence Register	0xFFFF_FFFF

31	30	29	28	27	26	25	24
CHSEQ7				CHSEQ6			
23	22	21	20	19	18	17	16
CHSEQ5				CHSEQ4			
15	14	13	12	11	10	9	8
CHSEQ3				CHSEQ2			
7	6	5	4	3	2	1	0
CHSEQ1				CHSEQ0			

Bits	Description	
[31:28]	CHSEQ7	Select Channel N As The 8th Conversion In Scan Sequence The definition of channel selection is the same as CHSEQ0.
[27:24]	CHSEQ6	Select Channel N As The 7th Conversion In Scan Sequence The definition of channel selection is the same as CHSEQ0.
[23:20]	CHSEQ5	Select Channel N As The 6th Conversion In Scan Sequence The definition of channel selection is the same as CHSEQ0.
[19:16]	CHSEQ4	Select Channel N As The 5th Conversion In Scan Sequence The definition of channel selection is the same as CHSEQ0.
[15:12]	CHSEQ3	Select Channel N As The 4th Conversion In Scan Sequence The definition of channel selection is the same as CHSEQ0.
[11:8]	CHSEQ2	Select Channel N As The 3rd Conversion In Scan Sequence The definition of channel selection is the same as CHSEQ0.
[7:4]	CHSEQ1	Select Channel N As The 2nd Conversion In Scan Sequence The definition of channel selection is the same as CHSEQ0.

[3:0]	CHSEQ0	<p>Select Channel N As The 1st Conversion In Scan Sequence</p> <p>If CHSEQ0[3]=0, one of the following channel is selected according to CHSEQ0[2:0].</p> <table border="1" data-bbox="678 363 1167 785"> <thead> <tr> <th>CHSEQ0</th> <th>Selected channel to ADC</th> </tr> </thead> <tbody> <tr><td>0000</td><td>Channel 0</td></tr> <tr><td>0001</td><td>Channel 1</td></tr> <tr><td>0010</td><td>Channel 2</td></tr> <tr><td>0011</td><td>Channel 3</td></tr> <tr><td>0100</td><td>Channel 4</td></tr> <tr><td>0101</td><td>Channel 5</td></tr> <tr><td>0110</td><td>Channel 6</td></tr> <tr><td>0111</td><td>Channel 7</td></tr> </tbody> </table> <p>If CHSEQ0[3]=1,. For CHSEQ0[0].=0,. Output of pre-amplifier with differential input pair is selected as ADC input. The input pair of pre-amplifier is defined in CHSEQ0[2:1].</p> <table border="1" data-bbox="755 961 1356 1199"> <thead> <tr> <th>CHSEQ0</th> <th>Channel pair to pre-amplifier input</th> </tr> </thead> <tbody> <tr><td>1000</td><td>Channel 0 and 1</td></tr> <tr><td>1010</td><td>Channel 2 and 3</td></tr> <tr><td>1100</td><td>Channel 4 and 5</td></tr> <tr><td>1110</td><td>None</td></tr> </tbody> </table> <p>For CHSEQ0[0].=1, 1x1 = no channel is selected, scan sequence end.</p>	CHSEQ0	Selected channel to ADC	0000	Channel 0	0001	Channel 1	0010	Channel 2	0011	Channel 3	0100	Channel 4	0101	Channel 5	0110	Channel 6	0111	Channel 7	CHSEQ0	Channel pair to pre-amplifier input	1000	Channel 0 and 1	1010	Channel 2 and 3	1100	Channel 4 and 5	1110	None
CHSEQ0	Selected channel to ADC																													
0000	Channel 0																													
0001	Channel 1																													
0010	Channel 2																													
0011	Channel 3																													
0100	Channel 4																													
0101	Channel 5																													
0110	Channel 6																													
0111	Channel 7																													
CHSEQ0	Channel pair to pre-amplifier input																													
1000	Channel 0 and 1																													
1010	Channel 2 and 3																													
1100	Channel 4 and 5																													
1110	None																													



A/D Compare Register 0/1 (ADC CMPn)

Register	Offset	R/W	Description	Reset Value
ADC_CMP0	ADC_BA+0x28	R/W	A/D Compare Register 0	0x0000_0000
ADC_CMP1	ADC_BA+0x2C	R/W	A/D Compare Register 1	0x0000_0000

31	30	29	28	27	26	25	24
Reserved				CMPDAT[11:8]			
23	22	21	20	19	18	17	16
CMPDAT[7:0]							
15	14	13	12	11	10	9	8
Reserved				CMPMCNT			
7	6	5	4	3	2	1	0
Reserved		CMPCH			CMPCOND	ADCMPIE	ADCM PEN

Bits	Description	Description
[31:28]	Reserved	Reserved.
[27:16]	CMPDAT	<p>Compare Data</p> <p>The 12 bits data are used to compare with conversion result of specified channel. Software can use it to monitor the external analog input pin voltage transition in scan mode without imposing a load on software.</p> <p>The data format should be consistent with the setting of ADCFM bit.</p>
[15:12]	Reserved	Reserved.
[11:8]	CMPMCNT	<p>Compare Match Count</p> <p>When the specified A/D channel analog conversion result matches the comparing condition, the internal match counter will increase 1. When the internal counter achieves the setting, (CMPMCNT+1) hardware will set the ADCMPF bit.</p>
[7:6]	Reserved	Reserved.
[5:3]	CMPCH	<p>Compare Channel Selection</p> <p>000 = Channel 0 conversion result is selected to be compared. 001 = Channel 1 conversion result is selected to be compared. 010 = Channel 2 conversion result is selected to be compared. 011 = Channel 3 conversion result is selected to be compared. 100 = Channel 4 conversion result is selected to be compared. 101 = Channel 5 conversion result is selected to be compared. 110 = Channel 6 conversion result is selected to be compared. 111 = Channel 7 conversion result is selected to be compared.</p>
[2]	CMPCOND	<p>Compare Condition</p> <p>0 = ADCMPF_x bit is set if conversion result is less than CMPDAT[11:0]. 1 = ADCMPF_x bit is set if conversion result is greater or equal to CMPDAT[11:0],</p>

[1]	ADCMPIE	<p>Compare Interrupt Enable</p> <p>0 = Disable 1 = Enable</p> <p>When converted data in RESULT is less (or greater) than the compare data CMPDAT[11:0], ADCMPF bit is asserted. If ADCMPIE is set to 1, a compare interrupt request is generated.</p>
[0]	ADCMPE	<p>Compare Enable</p> <p>0 = Disable compare. 1 = Enable compare.</p> <p>Set this bit to 1 to enable the comparison CMPDAT[11:0] with specified channel conversion result when converted data is loaded into ADC_DAT register.</p>



A/D Status Register (ADC STATUS)

Register	Offset	R/W	Description	Reset Value
ADC_STATUS	ADC_BA+0x30	R/W	A/D Status Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
OV							
15	14	13	12	11	10	9	8
VALID							
7	6	5	4	3	2	1	0
Reserved	CHANNEL			BUSY	ADCMPPF1	ADCMPPF0	ADIF

Bits	Description	
[31:24]	Reserved	Reserved.
[23:16]	OV	Over Run Flag It is a mirror to OV bit in ADC_DATn.
[15:8]	VALID	Data Valid Flag It is a mirror of VALID bit in ADC_DATn.
[7]	Reserved	Reserved.
[6:4]	CHANNEL	Current Conversion Channel This field reflects current conversion channel when BUSY=1. When BUSY=0, it shows the next channel will be converted. It is read only.
[3]	BUSY	BUSY/IDLE 0 = A/D converter is in idle state. 1 = A/D converter is busy at conversion. This bit is mirror of SWTRG bit in ADC_CTL. It is read only.
[2]	ADCMPPF1	Compare Flag When the selected channel A/D conversion result meets setting conditions in ADC_CMP1, then this bit is set to 1. And it is cleared by write 1. 0 = Converted result RESULT in ADC_DAT does not meet ADC_CMP1 setting. 1 = Converted result RESULT in ADC_DAT meets ADC_CMP1 setting,
[1]	ADCMPPF0	Compare Flag When the selected channel A/D conversion result meets setting conditions in ADC_CMP0, then this bit is set to 1. And it is cleared by write 1. 0 = Converted result RESULT in ADC_DAT does not meet ADC_CMP0 setting. 1 = Converted result RESULT in ADC_DAT meets ADC_CMP0 setting,

[0]	ADIF	A/D Conversion End Flag A status flag that indicates the end of A/D conversion. ADIF is set to 1 under the following two conditions: 1. When A/D conversion ends in single mode, 2. When A/D conversion ends on all channels specified by channel sequence register in scan mode. And it is cleared when 1 is written.
-----	------	--

A/D Calibration Register (ADC_CAL)

Register	Offset	R/W	Description	Reset Value
ADC_CAL	ADC_BA+0x34	R/W	A/D Calibration Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved						CALDONE	CALEN

Bits	Description	
[31:2]	Reserved	Reserved.
[1]	CALDONE	<p>Calibration is Done (read only)</p> <p>0 = A/D converter has not been calibrated or calibration is in progress if CALEN bit is set.</p> <p>1 = A/D converter self-calibration is done</p> <p>When 0 is written to CALEN bit, CALDONE bit is cleared by hardware immediately and be set to 1 after 96 ADC clocks. It is a read only bit.</p>
[0]	CALEN	<p>Self-Calibration Enable</p> <p>0 = Disable self-calibration</p> <p>1 = Enable self-calibration</p> <p>Software can set this bit to 1 enables A/D converter to do self-calibration function. It needs 127 ADC clocks to complete calibration. This bit must be kept at 1 after CALDONE asserted. Clearing this bit will disable self-calibration function.</p>



Pre-amplifier Gain Control Register (ADC_PGCTL)

Register	Offset	R/W	Description	Reset Value
ADC_PGCTL	ADC_BA+0x3C	R/W	ADC Pre-amplifier Gain Control Register	0x0000_0000

31	30	29	28	27	26	25	24	
Reserved								
23	22	21	20	19	18	17	16	
Reserved								
15	14	13	12	11	10	9	8	
Reserved			PAG_I		OS[4:2]			
7	6	5	4	3	2	1	0	
OS[1:0]		PAG_II					OPMUTE	

Bits	Description	
[31:13]	Reserved	Reserved.
[12:11]	PAG_I	Gain Setting Bits for the First Stage Of Pre-Amp 00 = -6 dB, 01 = 0 dB, 10 = 8 dB, 11 = 14 dB,
[10:6]	OS	Configuration for Pre-Amp OP Offset Bias Compensation Voltage There are 32 levels and 2mV per level @ 5V condition.
[5:1]	PAG_II	Gain Setting Bits for the Second Stage Of Pre-Amp 00000 = 0 dB, 00001 = 16 dB, 00010 = 17 dB, 00011 = 18 dB, 11111 = 46 dB.
[0]	OPMUTE	Mute Control of First Stage Pre-Amp for Offset Bias Calibration When this bit is set as "1", two input end of first stage pre-amp will be shorted, and feedback resistor of this stage will be shorted. 0 = open 1 = short

6 FLASH MEMORY CONTROLLER (FMC)

6.1 Overview

N572F065 equips with 64K bytes on chip embedded flash for application program memory (APROM) that can be updated thru ISP (In System Programming) procedure. ISP function could enable user to update program memory when chip is soldered on PCB. ISP also could update user configuration (register CONFIG) but new configurations are only effective after POR (Power-on Reset).

6.2 Features

- AHB interface compatible
- Run up to 24MHz with zero wait state for discontinuous address read access
- 64K bytes application program memory (APROM)
- In System Programming (ISP) to update on chip Flash EPROM

6.3 Flash Memory Organization

The N572F065 flash memory consists of 64K bytes flash and user configuration.

User configuration block provides several bytes to control system logic, like flash security lock, voltage detector voltage level, and so on. It works like a fuse for power on setting. It is loaded from flash memory to its corresponding control registers during chip power on. User can set these bits according to application request.

Table 6-1 Memory Address Map

Block Name	Size	Start Address	End Address
APROM	64KB	0x0000_0000	0x0000_FFFF
User Configuration	1 word	0x0030_0000	0x0030_0003

The Flash memory organization is shown as below:

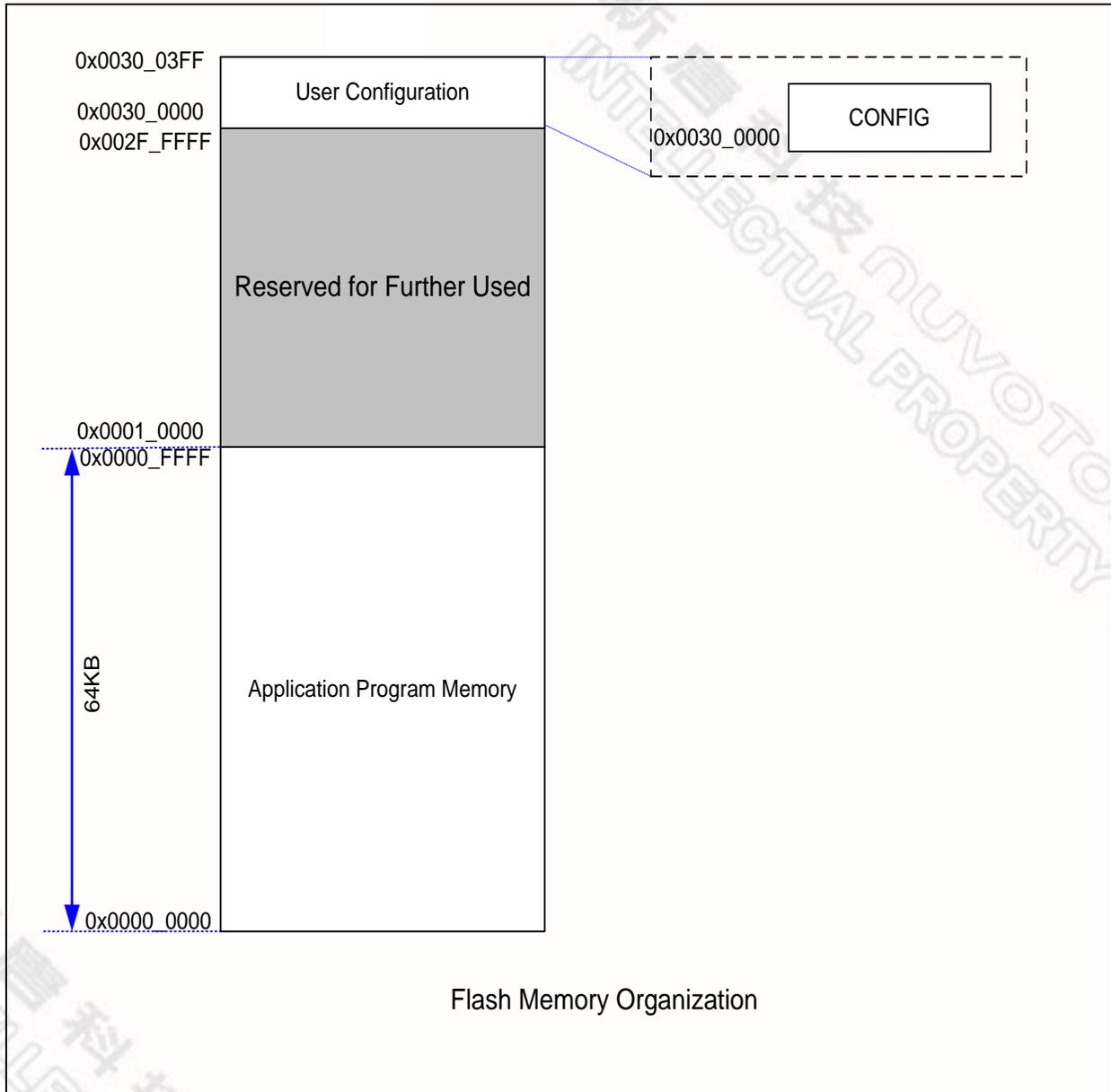


Figure 6-1 Flash Memory Organization



6.4 User Configuration

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
CONFIGURE Base Address: CFG_BA=0x0030_0000				
CONFIG	CFG_BA+0x000	R/W	User Configuration Memory	0x0000_0000

User Configuration Memory (CONFIG)

Register	Offset	R/W	Description	Reset Value
CONFIG	CFG_BA+0x000	R/W	User Configuration Memory	0x0000_0000

31	30	29	28	27	26	25	24
CWDTEN	-	-	CKF	-	CFOSC-		
23	22	21	20	19	18	17	16
CVDEN	-	CVDTV	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	-	-	-	-	LOCK	-

Bits	Description									
[31]	CWDTEN	Watchdog Enable 0 = Watchdog is disabled after power on. 1 = Watchdog is enabled after power on.								
[30:29]	Reserved	Reserved.								
[28]	CKF	Clock Filter Enable 0 = Disable clock filter 1 = Enable clock filter								
[27]	Reserved	Reserved.								
[26:24]	CFOSC	Power-on Clock Source Selection <table border="1"> <thead> <tr> <th>CFOSC[2:0]</th> <th>Clock Source</th> </tr> </thead> <tbody> <tr> <td>001</td> <td>External 12MHz/6MHz crystal clock</td> </tr> <tr> <td>111</td> <td>Internal RC 24MHz oscillator clock</td> </tr> <tr> <td>Others</td> <td>Reserved</td> </tr> </tbody> </table>	CFOSC[2:0]	Clock Source	001	External 12MHz/6MHz crystal clock	111	Internal RC 24MHz oscillator clock	Others	Reserved
CFOSC[2:0]	Clock Source									
001	External 12MHz/6MHz crystal clock									
111	Internal RC 24MHz oscillator clock									
Others	Reserved									

[23]	CVDEN	Voltage Detector Enable 0 = Enable Voltage Detector after power on. 1 = Disable Voltage Detector after power on.						
[22]	Reserved	Reserved.						
[21]	CVDTV	Voltage Detector Threshold Voltage Selection <table border="1" data-bbox="609 493 1088 640"> <thead> <tr> <th>CVDTV</th> <th>Threshold voltage</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>2.7V</td> </tr> <tr> <td>1</td> <td>3.0V</td> </tr> </tbody> </table>	CVDTV	Threshold voltage	0	2.7V	1	3.0V
CVDTV	Threshold voltage							
0	2.7V							
1	3.0V							
[20:2]	Reserved	Reserved.						
[1]	LOCK	Security Lock 0 = Flash data are locked. 1 = Flash data are not locked. When flash data is locked, (1) only device ID, CONFIG can be read by Writer and ICP thru serial debug interface. Other data are locked as 0xFFFFFFFF. (2) ISP can read data anywhere regardless of LOCK bit value. (3) SWD interface cannot access internal RAM and Flash						
[0]	Reserved	Reserved.						

6.5 In-System Programming (ISP)

The program and data flash memory support both in hardware In-Circuit Programming (ICP) and firmware based In-System programming (ISP). Hardware ICP programming mode uses the Serial-Wire Debug (SWD) port to program chip. Dedicated ICE Debug hardware (Nu-Link) or ICP gang-writers are available to reduce programming and manufacturing costs. For firmware updates in the field, the N572F065 provides an ISP mode allowing a device to be reprogrammed under software control.

ISP is performed without removing the device from the system. Two examples for the approach are shown below.

- (1) The ISP F/W is downloaded from USB to internal RAM, and CPU jumps from APROM to the downloaded F/W entrance to execute the ISP procedure. The data to be programmed to APROM come from USB.
- (2) The ISP F/W is stored in APROM initially. CPU copies the ISP F/W to internal RAM and then jumps from APROM to ISP F/W entrance in RAM to execute ISP procedure. The data to be programmed to APROM come from external SPI Flash.

6.5.1 ISP Procedure

To enable ISP functionality software must first ensure the ISP clock (CLK_AHBCLK.ISPCKEN) is present then set the FMC_ISPCTL.ISPEN bit. Software is required to write register SYS_REGLCTL (address 0x5000_0100) with 0x59, 0x16 and 0x88 sequentially before writing FMC_ISPCTL register. This procedure is used to protect flash memory from being destroyed owing to unintended write during power on/off duration.

Several error conditions are checked after software writes the FMC_ISPTRG[0](ISPGO) bit. If an error condition occurs, ISP operation is not been started and the ISP fail flag (FMC_ISPCTL.ISPFF) will be set instead. The ISPFF flag will remain set until it is cleared by software. Subsequent ISP procedure can be started even if ISPFF is set. It is recommended that software checks ISPFF bit and clear it after each ISP operation if it is set to "1".

When FMC_ISPTRG[0](ISPGO) bit is set, the CoretxM0 CPU will wait for ISP operation to finish, during this period; peripherals operate as usual. If any interrupt request occurs, CPU will not service it until ISP operation finishes.

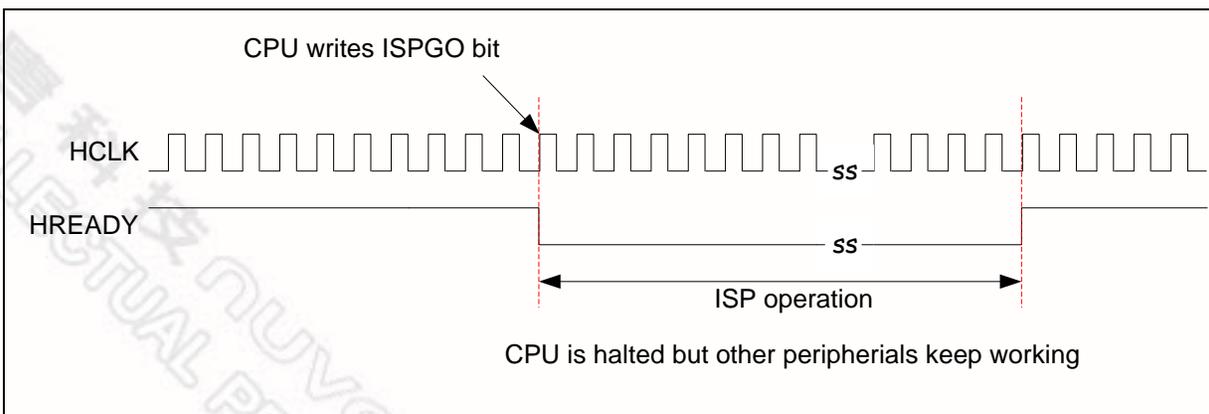


Figure 6-2 ISP Operation Timing

Table 6-2 ISP Command Set

ISP Mode	FMC_ISPCMD	FMC_ISPADDR			FMC_ISPDAT
	CMD[5:0]	A21	A20	A[19:0]	D[31:0]
Standby	0x3X	x	x	x	x
Read Company ID	0x0B	x	x	x	Returns "0x0000_00DA"
Read Device ID	0x0C	x	x	0x00000	Data output "Device ID"
FLASH Page Erase	0x22	0	A[20]	A[19:0]	x
FLASH Program	0x21	0	A[20]	A[19:0]	Data input
FLASH Read	0x00	0	A[20]	A[19:0]	Data output
CONFIG Page Erase	0x22	1	1	A[19:0]	x
CONFIG Program	0x21	1	1	A[19:0]	Data input
CONFIG Read	0x00	1	1	A[19:0]	Data output



6.6 Flash Control Register Map

R: read only, W: write only, R/W: both read and write

Register	Offset	R/W	Description	Reset Value
FMC Base Address: FMC_BA=0x5000_C000				
FMC_ISPCTL	FMC_BA+0x00	R/W	ISP Control Register	0x0000_0000
FMC_ISPADDR	FMC_BA+0x04	R/W	ISP Address Register	0x0000_0000
FMC_ISPDAT	FMC_BA+0x08	R/W	ISP Data Register	0x0000_0000
FMC_ISPCMD	FMC_BA+0x0C	R/W	ISP Command Register	0x0000_0000
FMC_ISPTRG	FMC_BA+0x10	R/W	ISP Trigger Control Register	0x0000_0000
FMC_FAT	FMC_BA+0x18	R/W	Flash Access Time Control Register	0x0000_0000



6.7 Flash Control Register Description

ISP Control Register (FMC ISPCTL)

This register is a protected register. To program this register needs an unlock sequence, writing “59h”, “16h”, “88h” sequentially to SYS_REGLCTL, to gain access. Refer to [Register Lock Control Register \(SYS_REGLCTL\)](#).

Register	Offset	R/W	Description	Reset Value
FMC_ISPCTL	FMC_BA+0x00	R/W	ISP Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved	ET			Reserved	PT		
7	6	5	4	3	2	1	0
Reserved	ISPPF	Reserved	CFGUEN	Reserved		EWEN	ISPEN

Bits	Description
[31:15]	Reserved
[14:12]	<p>Flash Page Erase Time</p> <p>000 = 20 ms (default). 001 = 25 ms. 010 = 30 ms. 011 = 35 ms. 100 = 3 ms. 101 = 5 ms. 110 = 10 ms. 111 = 15 ms.</p>
[11]	Reserved
[10:8]	<p>Flash Program Time</p> <p>000 = 40 μs (default). 001 = 45 μs. 010 = 50 μs. 011 = 55 μs. 100 = 20 μs. 101 = 25 μs. 110 = 30 μs. 111 = 35 μs.</p>

[7]	Reserved	Reserved.
[6]	ISPPF	<p>ISP Fail Flag</p> <p>This bit is set by hardware when a triggered ISP meets any of the following conditions:</p> <ul style="list-style-type: none"> (1) MCU writes (or erase) to Flash when EWEN is "0". (2) Destination address is illegal, such as over an available range. <p>Write 1 to clear.</p>
[5]	Reserved	Reserved.
[4]	CFGUEN	<p>CONFIG Update Enable</p> <p>0 = Disable. 1 = Enable.</p> <p>When enabled, ISP functions can access the CONFIG address space and modify device configuration area.</p>
[3:2]	Reserved	Reserved.
[1]	EWEN	<p>Enable Erase/Write Of ISP Function</p> <p>0 = Disable erase/write. 1 = Enable erase/write.</p>
[0]	ISPEN	<p>ISP Enable</p> <p>0 = Disable ISP function. 1 = Enable ISP function.</p>



ISP Address Register (FMC ISPADDR)

Register	Offset	R/W	Description	Reset Value
FMC_ISPADDR	FMC_BA+0x04	R/W	ISP Address Register	0x0000_0000

31	30	29	28	27	26	25	24
ISPADDR[31:24]							
23	22	21	20	19	18	17	16
ISPADDR[23:16]							
15	14	13	12	11	10	9	8
ISPADDR[15:8]							
7	6	5	4	3	2	1	0
ISPADDR[7:0]							

Bits	Description
[31:0]	<p>ISPADDR</p> <p>ISP Address Register This is the memory address register that a subsequent ISP command will access. ISP operation are carried out on 32-bit word only, consequently ISPADDR [1:0] must be 00b for correct ISP operation. N572F065 equips with an 16Kx32 bits embedded flash.</p>

ISP Data Register (FMC_ISPDAT)

Register	Offset	R/W	Description	Reset Value
FMC_ISPDAT	FMC_BA+0x08	R/W	ISP Data Register	0x0000_0000

31	30	29	28	27	26	25	24
ISPDAT[31:24]							
23	22	21	20	19	18	17	16
ISPDAT[23:16]							
15	14	13	12	11	10	9	8
ISPDAT[15:8]							
7	6	5	4	3	2	1	0
ISPDAT[7:0]							

Bits	Description
[31:0]	<p>ISPDAT</p> <p>ISP Data Register</p> <p>Write data to this register before an ISP program operation.</p> <p>Read data from this register after an ISP read operation</p>



ISP Command (FMC ISPCMD)

Register	Offset	R/W	Description	Reset Value
FMC_ISPCMD	FMC_BA+0x0C	R/W	ISP Command Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved		CMD					

Bits	Description	
[31:6]	Reserved	Reserved.
[5:0]	CMD	ISP Command Operation Mode : CMD ----- Standby : 0x3X Read CID : 0x0B Read DID : 0x0C Flash Page Erase : 0x22 Flash Program : 0x21 Flash Read : 0x00



ISP Trigger Control Register (FMC ISPTRG)

Register	Offset	R/W	Description	Reset Value
FMC_ISPTRG	FMC_BA+0x10	R/W	ISP Trigger Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							ISPGO

Bits	Description	
[31:1]	Reserved	Reserved.
[0]	ISPGO	<p>ISP Start Trigger</p> <p>Write 1 to start ISP operation. This bit will be cleared to 0 by hardware automatically when ISP operation is finished.</p> <p>0 = ISP operation is finished.</p> <p>1 = ISP is ongoing.</p>

Flash Access Time Control Register (FMC FAT)

Register	Offset	R/W	Description	Reset Value
FMC_FAT	FMC_BA+0x18	R/W	Flash Access Time Control Register	0x0000_0000

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved				FATS[2:0]			FPSEN

Bits	Descriptions																			
[31:4]	Reserved	Reserved																		
[3:1]	FATS	<p>Flash Access Time Window Select</p> <p>These bits are used to decide flash sense amplifier active duration. For tsmc 0.25um embedded flash, it is about 40ns.</p> <table border="1"> <thead> <tr> <th>FATS</th> <th>Access Time window (ns)</th> </tr> </thead> <tbody> <tr><td>000</td><td>40</td></tr> <tr><td>001</td><td>50</td></tr> <tr><td>010</td><td>60</td></tr> <tr><td>011</td><td>70</td></tr> <tr><td>100</td><td>80</td></tr> <tr><td>101</td><td>90</td></tr> <tr><td>110</td><td>100</td></tr> <tr><td>111</td><td>reserved</td></tr> </tbody> </table>	FATS	Access Time window (ns)	000	40	001	50	010	60	011	70	100	80	101	90	110	100	111	reserved
FATS		Access Time window (ns)																		
000		40																		
001		50																		
010		60																		
011		70																		
100		80																		
101		90																		
110		100																		
111	reserved																			
[0]	FPSEN	<p>Flash Power Save Enable</p> <p>On chip flash memory access time is around 40ns. If CPU clock is slower than 50MHz, then S/W can enable flash power saving function.</p> <p>0 = Disable flash power saving. 1 = Enable flash power saving,</p>																		



7 FUNCTION REGISTER COMPARISON

Register	N572F072/P072	N572F065
APU base	APU_BA=0x5000_8000	APU_BA=0x400C_0000
CLK_PWRCTL	<ul style="list-style-type: none"> Remove bit 0. Default value of CLK_PWRCTL [3] is "1". 	Default value of CLK_PWRCTL[3] is "0".
CLK_AHBCLK	APUCKEN is moved to APUCKEN=CLK_AHBCLK[3].	APUCKEN=CLK_APBCLK[30]
SYS_GPA_HS	New added register	No
CLK_CLKSEL0	Add SysTick clock source selection, STCLKSEL = CLK_CLKSEL0[4:3].	
CLK_CLKSEL1	<ul style="list-style-type: none"> Modify clock source selection because 12MHz XTAL is removed, 24MHz RC is changed to 48MHz RC. Add SPI0 clock source selection. SPI0SEL = CLK_CLKSEL1[5:4]. 	
SYST_CSR SYST_RVR SYST_CVR	New added SysTick registers	No
CLK_PLLCON	Modify PLL2 clock source and output frequency	
SPI_CTL (SPI1 only)	Add master/slave mode selection. SLAVE=SPI_CTL[18].	
SPI_SSCTL (SPI1 only)	Add slave mode related bits. LTRIG_FLAG= SPI_SSCTL[5], SS_LTRIG=SPI_SSCTL[4].	
SPI0_RCLK	New added register	No
APU_CH0DAT0~ APU_CH0DAT7 APU_CH1DAT0	16-bit registers	13-bit registers
ADCALR	No	Yes
ADC_PGCTL	<ul style="list-style-type: none"> Definitions of control bits are changed. OS=ADC_PGCTL[16:8], PAG_I=ADC_PGCTL[28:24], PAG_II=ADC_PGCTL[2:1]. New added APPS, MICE bits. 	OS=ADC_PGCTL[10:6], PAG_I=ADC_PGCTL[12:11], PAG_II=ADC_PGCTL[5:1].
APU_VM	VOLUM=111: reserved	VOLUM=111: reserved and prohibited
ADC_MIBSCTL	New added register	No
FATCON	No	Yes



8 REVISION HISTORY

VERSION	DATE	PAGE/ CHAP.	DESCRIPTION
V1.00	Aug, 2014	-	Initial release.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

*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.*