NUC029FAE Series CMSIS BSP Guide

Directory Introduction for 32-bit NuMicro® Family

Directory Information

Document	Driver reference guide and revision history.
Library	Driver header and source files.
SampleCode	Driver sample code.

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

www.nuvoton.com

TABLE OF CONTENTS

1 Document

CMSIS.html	Document of CMSIS version 4.5.0.
NuMicro NUC029FAE Driver Reference Guide.html	This document describes the usage of drivers in NUC029FAE BSP.
NuMicro NUC029FAE Series CMSIS BSP Revision History.pdf	This document shows the revision history of NUC029FAE BSP.

2 Library

CMSIS	Cortex® Microcontroller Software Interface Standard (CMSIS) V4.5.0 definitions by Arm® Corp.
Device	CMSIS compliant device header file.
StdDriver	All peripheral driver header and source files.

3 SampleCode

Hard_Fault_Sample	Show hard fault information when hard fault happened.
ISP	ISP firmware samples.
NuTiny-NUC029FAE	Same codes for NUC029FAE Tiny Board
RegBased	Sample codes implemented without access standard library but access registers directly.
Semihost	Show how to print and get character through IDE console window.
StdDriver	Demonstrate the usage of NUC029FAE MCU peripheral driver APIs.
Template	A project template for NUC029FAE MCU.

4 SampleCode\ISP

ISP_I2C	In-System-Programming sample code through I^2C interface.
ISP_RS485	In-System-Programming sample code through RS485 interface.
ISP_SPI	In-System-Programming sample code through SPI interface.
ISP_UART	In-System-Programming sample code through UART interface.

5 SampleCode\NuTiny-NUC029FAE

LED	Toggle P2.4 to turn on / off the board LED.

6 SampleCode\RegBased

Flash Memory Controller (FMC)

FMC_RW	Show FMC read flash IDs, erase, read, and write functions.

General Purpose I/O (GPIO)

GPIO	Use GPIO driver to control the GPIO pin direction, control their high/low state, and how to use GPIO interrupts.

Timer Controller (TIMER)

Timer_Periodic	Use the timer periodic mode to generate timer interrupt every 1 second.
Timer_TriggerCountingMode	Use the timer pin P3.2 to demonstrate timer trigger counting mode function. And displays the measured input frequency to UART console.

Watchdog Timer (WDT)

WDT_Polling	Use polling mode to check WDT time-out state and reset WDT after time out occurs.

PWM Generator (PWM)

PWM_DoubleBuffer	Demonstrate the PWM double buffer feature.

UART Interface Controller (UART)

UART_IrDA	Show how to transmit and receive UART data in UART IrDA mode.
UART_TxRx_Function	Transmit and receive data from PC terminal through RS232 interface.

Serial Peripheral Interface (SPI)

SPI_Loopback	Demonstrate SPI function by connect MOSI (P0.5) with MISO (P0.6).

I²C Serial Interface Controller (I²C)

I2C_ Interrupt_EEPROM	Read/write EEPROM via I²C interface using interrupt mode.

Analog-to-Digital Converter (ADC)

ADC_Convert	Demonstrate ADC function by repeatedly convert the input of ADC channel 0 (P5.3) and shows the result on UART console.

Analog Comparator Controller (ACMP)

ACMP	Demonstrate Analog comparator (ACMP) comparison by comparing CPP0 (P1.5) with Band-gap voltage and shows the result on UART console.

7 SampleCode\StdDriver

System Manager (SYS)

SYS	Demonstrate how to get PDID, get and clear reset source, configure BOD, and output system clock to CKO pin with the system clock / 4 frequency.
SYS_PowerDown_MinCurrent	Demonstrate how to minimize power consumption when entering power down mode.

Flash Memory Controller (FMC)

FMC_IAP	This sample code includes LDROM image (fmc_ld_iap) and APROM image (fmc_ap_main). It shows how to branch between APROM and LDROM. To run this sample code, the boot mode must be “Boot from APROM with IAP”.
FMC_RW	Show FMC read flash IDs, erase, read, and write functions.

General Purpose I/O (GPIO)

GPIO	Use GPIO driver to control the GPIO pin direction, control their high/low state, and how to use GPIO interrupts.

Timer Controller (TIMER)

Timer_Delay	Demonstrate the usage of TIMER_Delay() API to generate a 1 second delay
Timer_EventCounter	Use pin P3.4 to demonstrates timer event counter function.
Timer_FreeCountingMode	Use the timer pin P3.2 to demonstrate timer free counting mode function. Also display the measured input frequency to UART console.

Timer_ToggleOut	Demonstrate the timer 0 toggle out function on pin P3.4.

Watchdog Timer (WDT)

WDT_Wakeup	Use WDT to wake up system from Power-down mode periodically.

PWM Generator (PWM)

PWM_DeadZone	Demonstrate the dead-zone feature with PWM.

UART Interface Controller (UART)

UART_IrDA	Show how to transmit and receive UART data in UART IrDA mode.
UART_TxRx_Function	Transmit and receive data from PC terminal through RS232 interface.

Serial Peripheral Interface (SPI)

SPI_LoopBack	Demonstrate SPI function by connect MOSI (P0.5) with MISO (P0.6).

I²C Serial Interface Controller (I²C)

I2C_Interrupt_EEPROM	Read/write EEPROM via I²C interface using interrupt mode.

Analog-to-Digital Converter (ADC)

ADC_Compare	Demonstrate ADC conversion and comparison function by monitoring the conversion result of channel 0.

Analog Comparator Controller (ACMP)

ACMP	Demonstrate Analog comparator (ACMP) comparison by

	comparing CPP0 (P1.5) with Band-gap voltage and shows the result on UART console.

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.