

NM1120 CMSIS BSP Directory

Directory Introduction for 32-bit NuMicro™ Family

Directory Information

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

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 NuMicro 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 Document Information

NuMicro NM1120 CMSIS BSP Revision History.pdf	This document shows all the revision history about NM1120 BSP.
NuMicro NM1120 Series Driver Reference Guide.chm	This document describes the usage of drivers in NM1120 Board Support Package (BSP).

2 Library Information

CMSIS	Cortex [®] Microcontroller Software Interface Standard (CMSIS) V3.01 definitions by ARM [®] Corp.
Device	CMSIS compliant device header file.
StdDriver	All peripheral driver header and source files.

3 Sample Code Information

Hard_Fault_Sample	Accessing the memory space to generate bus fault exception is not supported in NM1120. If bus fault handler cannot execute the exception, Hard fault exception will take care of it.
Semihost	Show how to print and get characters with IDE console window.
StdDriver	A sample code to demonstrate the usage of NM1120 MCU peripheral driver APIs.
Template	A project template for NM1120 MCU.

4 \SampleCode\StdDriver

ACMP	Demonstrate analog comparator (ACMP) comparison by comparing ACMP0_P0 input and VBG voltage and shows the result on UART console.
BPWM_DeadZone	Demonstrate the dead-zone feature with BPWM.
BPWM_DoubleBuffer	Demonstrate the BPWM double buffer feature.
EADC_Ind2SH	Convert ADC0 channel 0 and ADC1 channel 0 in Independent 2SH mode and print conversion results.
EADC_IndSimple	Convert ADC0 channel 0 and ADC1 channel 0 in Independent Simple mode and print conversion results.
EADC_IndSimple_BandGap	Convert ADC0 channel 6 (Band-Gap) in Independent Simple mode and print conversion results.
EADC_IndSimple_TempSensor	Convert ADC1 channel 6 (Temperature Sensor) in Independent Simple mode and print conversion results.
EADC_PWMTrigger	Configure PWM0 to trigger ADC0 channel 0 periodically and print conversion results.
EADC_SimSeq3R	Convert ADC0 channel 0, channel 6, and ADC1 channel 0 in EADC Simultaneous Sequential 3R mode and print conversion results.
EADC_SimSeq4R	Convert ADC0 channel 0, channel 6, ADC1 channel 0, and channel 3 in EADC Simultaneous Sequential 4R mode and print conversion results.
EADC_SimSimple	Convert ADC0 channel 0 and ADC1 channel 0 in Simultaneous Simple mode and print conversion results.
EADC_TimerTrigger	Configure Timer0 to trigger ADC0 channel 0 periodically and print conversion results.
EADC_WCompare	Demonstrate EADC conversion and window comparison function by monitoring the conversion result of ADC0 channel 0.
ECAP_Capture	Configure ECAP channel 0 to capture input square wave and print capture results. The input square wave is

	generated by Timer0 and GPIO output pin.
ECAP_CmpMatch	Demonstrate ECAP capture and compare match function by monitoring the capture result of ECAP channel 0.
EPWM_DeadZone	Demonstrate the dead-zone feature with EPWM.
EPWM_DoubleBuffer	Demonstrate the EPWM double buffer feature.
FMC_CRC32	Show FMC CRC32 calculating capability.
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.
GPIO_IOTest	Use GPIO driver to control the GPIO pin direction and the high/low state, and show how to use GPIO interrupts.
GPIO_PowerDown	Demonstrate how to wake up system form Power-down mode by GPIO interrupt.
HDIV	Demonstrate how to divide two signed integer by HDIV engine.
PGA_PGAO	Demonstrate how to amplify input signal with different gain level and output to PGA_O output pin.
SYS_CLKO	Demonstrate how to output different clocks one after another to the same CLKO (PA0) pin.
SYS_Control	Demonstrate some functions about system manager controller. It includes read PDID, get reset source, system write-protection, power-down wake up by Watchdog timer, and CPU reset.
TIMER_Delay	Demonstrate the usage of TIMER_Delay() API to generate a 1 second delay.
TIMER_EventCounter	Use pin PB.3 to demonstrates timer event counter function.

TIMER_FreeCountingMode	Use the ACMP0 positive input pin to demonstrate timer free counting mode function. And displays the measured input frequency to console.
TIMER_Periodic	Use the timer periodic mode to generate timer interrupt every 1 second.
TIMER_ToggleOut	Demonstrate the timer 0 toggle out function on pin PB.3.
TIMER_Wakeup	Use timer to wake up system from Power-down mode periodically.
USCI_I2C_EEPROM	Show how to use USCI_I2C interface to access EEPROM.
USCI_I2C_Master	Show a Master how to access Slave. This sample code needs to work with USCI_I2C_Slave.
USCI_I2C_Master	Show a Master how to access Slave. This sample code needs to work with USCI_I2C_Slave.
USCI_SPI_Loopback	Implement USCI_SPI1 Master loop back transfer. This sample code needs to connect USCI_SPI1_MISO pin and USCI_SPI1_MOSI pin together. It will compare the received data with transmitted data.
USCI_SPI_MasterMode	Configure USCI_SPI1 as Master mode and demonstrate how to communicate with an off-chip SPI Slave device. Needs to work with USCI_SPI_SlaveMode sample code.
USCI_SPI_SlaveMode	Configure USCI_SPI1 as Slave mode and demonstrate how to communicate with an off-chip SPI Master device. This sample code needs to work with USCI_SPI_MasterMode sample code.
USCI_UART_TxRxFunction	Transmit and receive data from PC terminal through RS232 interface.
WDT_Polling	Use polling mode to check WDT time-out state and reset WDT after time out occurs.
WDT_Wakeup	Use WDT to wake system up from power-down mode periodically.

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.*