

NUC230/NUC240 BSP Directory Introduction

Rev.3.00.001

Directory Information

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

Document Information

BSP Revision History	Show all the revision history about specific BSP.
Driver Reference Guide	Describe the definition, input and output of each API.

Library Information

CMSIS	CMSIS definitions by ARM® Corp.
Device	CMSIS compliant device header file.
StdDriver	All peripheral driver header and source files.

Sample Code Information

\SampleCode\Template	Software Development Template.
\SampleCode\Semihost	A sample code to show how to debug with semihost message print.
\SampleCode\RegBased	The sample codes that only include driver header files and access control registers directly.
\SampleCode\StdDriver	NUC200 Series Driver Samples

\SampleCode\RegBased

ACMP	This sample code shows how ACMP works with internal band-gap voltage.
ADC_ContinuousScanMode	ADC operates in continuous scan mode and finishes two cycles of conversion for the specified channels.
ADC_PwmTrigger	This sample code shows how to trigger ADC by PWM.
ADC_ResultMonitor	This sample code demonstrates how to use the digital compare function to monitor the conversion result of channel 2.
ADC_SingleCycleScanMode	In this example, ADC operates in single cycle scan mode and finishes one cycle of conversion for the specified channels.
ADC_SingleMode	In this example, ADC finishes the conversion of the specified channel with single mode.
CAN_Set_MaskFilter	Needs to work with CAN_Test_MaskFilter. This sample code demonstrates how to use MaskFilter to receive message in Normal mode.
CAN_Test_MaskFilter	Needs to work with CAN_Set_MaskFilter. This sample code use message Object No.1 to send message objects(ID=0x700~0x70F).
CRC_8	To show how to perform CRC-8 operation and get the CRC checksum result.
CRC_CCITT	To show how to perform CRC_CCITT operation and get the CRC checksum result.
FMC_IAP	To show how to call LDROM functions from APROM.
FMC_MultiBoot	To show how to implement multi-boot system to boot from different applications in APROM.
FMC_RW	To show how to read/program embedded flash by ISP function.
GPIO_EINTAndDebounce	Check EINT0/1 interrupt functions with enabling interrupt debounce function.

GPIO_INT	Select two pins to test interrupt rising-edge and falling-edge functions.
GPIO_OutputInput	Select two pins to test GPIO input mode and output mode.
GPIO_PowerDown	Select one pin to test GPIO wake-up by GPIO interrupt.
I2C_EEPROM	To show how to access EEPROM by I2C interface.
I2C_GCMode_MASTER	Needs to work with I2C_GCMode_SLAVE. To show how a Master use I2C address 0x0 to write data to I2C Slave.
I2C_GCMode_SLAVE	Needs to work with I2C_GCMode_MASTER. To show how to receive Master data in GC (General Call) mode.
I2C_MASTER	Needs to work with I2C_SLAVE. To show how a Master access Slave.
I2C_SLAVE	Needs to work with I2C_MASTER. To show how to set I2C in slave mode to receive the data of a Master.
I2C_Wakeup_Master	Needs to work with I2C_Wakeup_Slave. To show how to wake up MCU from power down.
I2C_Wakeup_Slave	Needs to work with I2C_Wakeup_Master. To show how to set I2C to wake up MCU from power down mode.
I2S_Master	Needs to work with I2S_Slave sample code. This sample code shows how I2S works in master mode.
I2S_Slave	Needs to work with I2S_Master sample code. This sample code shows how I2S works in slave mode.
PDMA	This sample uses PDMA channel 6 to demonstrate memory to memory transfer.
PS2	This sample code is demonstration of how to control PS/2 mouse movement on the screen.
PWM_Capture	This sample code demonstrates PWMB Channel 2 capture PWMB Channel 1 Waveform.
PWM_DeadZone	This sample code demonstrates PWM Dead Zone function.

PWM_DoubleBuffer	This sample uses PWM Double Buffer function to change duty cycle and period of output waveform.
RTC_PowerDown	Use RTC alarm interrupt event to wake-up system.
RTC_TimeAndTick	This sample code shows how to get the current data/time per tick.
SCUART_TxRx	This sample code demonstrates how to use smartcard interface UART mode to print "Hello World!".
SPI_MasterFifoMode	Needs to work with SPI_SlaveFifoMode sample code. This sample code shows how to communicate with an off-chip SPI slave device with FIFO mode.
SPI_SlaveFifoMode	Needs to work with SPI_MasterFifoMode sample code. This sample code shows how to communicate with an off-chip SPI master device with FIFO mode.
SYS	This sample code demostartes change system clock to different PLL frequency and output system clock from CLK0(PB.8) pin. The CLK0 clock frequency is system clock/4.
TIMER_Capture	This sample code is showing how to use capture function on timer-2 capture input pin.
TIMER_Counter	This sample code is showing how to use counter function on timer-1 counter input pin.
TIMER_PeriodicINT	Enable all timer with difference clock source and set to periodic mode then check the interrupt period time is reasonable or not.
TIMER_PowerDown	Use timer-0 toggle-out interrupt event to wake-up system.
UART_AutoFlow_Master	Needs to work with UART_AutoFlow_Slave. The sample code shows how to transmit and receive data using auto flow control.
UART_AutoFlow_Slave	Needs to work with UART_AutoFlow_Master. The sample code shows how to transmit and receive data using auto flow control.
UART_IrDA_Master	Needs to work with UART_IrDA_Slave. The sample code shows how to transmit and receive data in UART IrDA mode.

UART_IrDA_Slave	Needs to work with UART_IrDA_Master. The sample code shows how to transmit and receive data in UART IrDA mode.
UART_LIN	The sample code shows how to transmit LIN header and response.
UART_RS485_Master	Needs to work with UART_RS485_Slave. The sample code shows how to transmit and receive data in UART RS485 mode.
UART_RS485_Slave	Needs to work with UART_RS485_Master. The sample code shows how to transmit and receive data in UART RS485 mode.
UART_TxRx_Function	The sample code shows how UART transmit and receive data from PC terminal through RS232 interface.
WDT_PowerDown	Use WDT time-out interrupt event to wake-up system.
WDT_TimeoutINT	Select one WDT time-out interval period time and check the time-out interrupt signal.
WDT_TimeoutReset	Check system can be reset while WDT time-out occurred and reset delay period expires.
WWDT_CompareINT	Select one WWDT window compare value then check the time-out period time and WWDT reset system can be work normally or not.

\SampleCode\StdDriver

ACMP	This sample code shows how ACMP works with internal band-gap voltage.
ADC_ContinuousScanMode	ADC operates in continuous scan mode and finishes two cycles of conversion for the specified channels.
ADC_PwmTrigger	This sample code shows how to trigger ADC by PWM.
ADC_ResultMonitor	This sample code demonstrates how to use the digital compare function to monitor the conversion result of channel 2.
ADC_SingleCycleScanMode	In this example, ADC operates in single cycle scan mode and

	finishes one cycle of conversion for the specified channels.
ADC_SingleMode	In this example, ADC finishes the conversion of the specified channel with single mode.
CAN_BasicMode_Receive	Needs to work with CAN_BasicMode_Transmit. This sample code demonstrates how to receive message in Basic mode.
CAN_BasicMode_Transmit	Needs to work with CAN_BasicMode_Receive. This sample code demonstrates how to transmit message in Basic mode.
CAN_NormalMode_Receive	Needs to work with CAN_NormalMode_Transmit. This sample code demonstrates how to receive message in Normal mode.
CAN_NormalMode_Transmit	Needs to work with CAN_NormalMode_Receive. This sample code demonstrates how to transmit message in Normal mode.
CRC_8	To show how to perform CRC-8 operation and get the CRC checksum result.
CRC_CCITT	To show how to perform CRC_CCITT operation and get the CRC checksum result.
FMC_IAP	To show how to call LDROM functions from APROM.
FMC_RW	To show how to read/program embedded flash by ISP function.
GPIO_EINTAndDebounce	Check EINT0/1 interrupt functions with enabling interrupt de-bounce function.
GPIO_INT	Select two pins to test interrupt rising-edge and falling-edge functions.
GPIO_OutputInput	Select two pins to test GPIO input mode and output mode.
GPIO_PowerDown	Select one pin to test GPIO wake-up by GPIO interrupt.
I2C_EEPROM	To show how to access EEPROM by I2C interface.
I2C_GCMode_MASTER	Needs to work with I2C_GCMode_SLAVE. To show how a Master use I2C address 0x0 to write data to I2C Slave.
I2C_GCMode_SLAVE	Needs to work with I2C_GCMode_MASTER. To show how to

	receive Master data in GC (General Call) mode.
I2C_MASTER	Needs to work with I2C_SLAVE. To show how a Master access Slave.
I2C_SLAVE	Needs to work with I2C_MASTER. To show how to set I2C in slave mode to receive the data of a Master.
I2C_Wakeup_Master	Needs to work with I2C_Wakeup_Slave. To show how to wake up MCU from power down.
I2C_Wakeup_Slave	Needs to work with I2C_Wakeup_Master. To show how to set I2C to wake up MCU from power down mode.
I2S_Master	Needs to work with I2S_Slave sample code. This sample code shows how I2S works in master mode.
I2S_Slave	Needs to work with I2S_Master sample code. This sample code shows how I2S works in slave mode.
PDMA	This sample uses PDMA channel 6 to demonstrate memory to memory transfer.
PS2	This sample code is demonstration of how to control PS/2 mouse movement on the screen.
PWM_Capture	This sample code demonstrates PWMB Channel 2 capture PWMB Channel 1 Waveform.
PWM_DeadZone	This sample code demonstrates PWM Dead Zone function.
PWM_DoubleBuffer	This sample uses PWM Double Buffer function to change duty cycle and period of output waveform.
RTC_PowerDown	Use RTC alarm interrupt event to wake-up system.
RTC_TimeAndTick	This sample code shows how to get the current data/time per tick.
SC_ReadATR	This sample code demonstrates how to get smart card ATR data.
SCUART_TxRx	This sample code demonstrates how to use smartcard interface

	UART mode to print "Hello World!".
SPI_LoopBackTest	This sample code demonstrates the data transfer in SPI master mode.
SYS	This sample code demonstrates change system clock to different PLL frequency and output system clock from CLK0(PB.8) pin. The CLK0 clock frequency is system clock/4.
TIMER_Capture	This sample code is showing how to use capture function on timer-2 capture input pin.
TIMER_Counter	This sample code is showing how to use counter function on timer-1 counter input pin.
TIMER_PeriodicINT	Enable all timer with difference clock source and set to periodic mode then check the interrupt period time is reasonable or not.
TIMER_PowerDown	Use timer-0 toggle-out interrupt event to wake-up system.
UART_AutoFlow_Master	Needs to work with UART_AutoFlow_Slave. The sample code shows how to transmit and receive data using auto flow control.
UART_AutoFlow_Slave	Needs to work with UART_AutoFlow_Master. The sample code shows how to transmit and receive data using auto flow control.
UART_IrDA_Master	Needs to work with UART_IrDA_Slave. The sample code shows how to transmit and receive data in UART IrDA mode.
UART_IrDA_Slave	Needs to work with UART_IrDA_Master. The sample code shows how to transmit and receive data in UART IrDA mode.
UART_LIN	The sample code shows how to transmit LIN header and response.
UART_RS485_Master	Needs to work with UART_RS485_Slave. The sample code shows how to transmit and receive data in UART RS485 mode.
UART_RS485_Slave	Needs to work with UART_RS485_Master. The sample code shows how to transmit and receive data in UART RS485 mode.
UART_TxRx_Function	The sample code shows how UART transmit and receive data from PC terminal through RS232 interface.

WDT_PowerDown	Use WDT time-out interrupt event to wake-up system.
WDT_TimeoutINT	Select one WDT time-out interval period time and check the time-out interrupt signal.
WDT_TimeoutReset	Check system can be reset while WDT time-out occurred and reset delay period expires.
WWDT_CompareINT	Select one WWDT window compare value then check the time-out period time and WWDT reset system can be work normally or not.

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.