NUC505 Series CMSIS BSP Guide

Directory Introduction for 32-bit NuMicro™ Family

Directory Information

Please extract the “NUC505_Series_BSP_CMSIS_V3.03.000.zip” file firstly, and then put the “NUC505_Series_BSP_CMSIS_V3.03.000” folder into the working folder (e.g. .\Nuvoton\BSP Library\).

This BSP folder contents:

Document	Device driver reference manual and reversion history.
Library	Device driver header and source files.
SampleCode	Device driver sample code.
ThirdParty	Third party source code, including FatFs, LibMAD, and FreeRTOS™.

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

www.nuvoton.com

1 .\Document\

CMSIS.html	Introduction of CMSIS version 4.5.0. CMSIS components included CMSIS-CORE, CMSIS-Driver, CMSIS-DSP, etc. ● CMSIS-CORE: API for the Cortex-M4 processor core and peripherals. ● CMSIS-Driver: Defines generic peripheral driver interfaces for middleware making it reusable across supported devices.505 ● CMSIS-DSP: DSP Library Collection with over 60 Functions for various data types: fix-point (fractional q7, q15, q31) and single precision floating-point (32-bit).
NuMicro NUC505 Series CMSIS BSP Revision History.pdf	The revision history of NUC505 Series BSP.
NuMicro NUC505 Series Driver Reference Guide.chm	The usage of drivers in NUC505 Series 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.
UsbHostLib	USB host library source code.

3 .\SampleCode\

BootTemplate	Different boot sample code.
CortexM4	Cortex®-M4 sample code.
Hard_Fault_Sample	Accessing the memory space to generate bus fault exception is not supported in NUC505. If bus fault hander cannot execute the exception, Hard fault exception will take care of it.
ISP	This demo code provides a solution for firmware update.
Semihost	Show how to print and get characters with IDE console window.
StdDriver	A sample code to demonstrate the usage of NUC505 series MCU peripheral driver APIs.
Template	A project template for NUC505 series MCU.

4 .\ThirdParty\

FATFS	A generic FAT file system module for small embedded systems. Its official website is: http://elm-chan.org/fsw/ff/00index_e.html
FreeRTOS	A real time operating system available for free download. Its official website is: http://www.freertos.org/.
LibMAD	A MPEG audio decoder library which currently supports MPEG-1 and the MPEG-2 extension to lower sampling frequencies, as well as the de facto MPEG 2.5 format. All three audio layers — Layer I, Layer II, and Layer III (i.e. MP3) are fully implemented. This library is distributed under GPL license. Please contact Underbit Technologies (http://www.underbit.com/) for the commercial license.

5 .\SampleCode\BootTemplate

CritcalOnSRAM	Demonstrate how to locate a program mainly on SPI Flash for typical use, except critical parts on SRAM for fast execution.
FullOnSRAM	Demonstrate how to locate a program fully on SRAM.
Loader	Demonstrate how to launch a program via loader which is located fully on SRAM.
MainOnSRAM	Demonstrate how to locate a program mainly on SRAM for fast execution, except startup parts on SPI Flash for initialization.
Overlay	Demonstrate how to use overlay to run a large program in small size SRAM, through which a large program is divided into smaller ones which are located in the same SRAM address for execution.

6 \SampleCode\CortexM4

BitBand	Demonstrate the usage of Cortex®-M4 BitBand.
DSP_FFT	Demonstrate how to call ARM CMSIS DSP library to calculate FFT (Fast Fourier Transform).
MPU	Demonstrate the usage of Cortex®-M4 MPU.

7 .\SampleCode\StdDriver

ADC	1. Demonstrate ADC conversion from channel 0. 2. Demonstrate analog keypad detection from channel 2.
GPIO	Use GPIO driver to control the GPIO pin direction, control their high/low state, and how to use GPIO interrupts.
GPIO_PowerDown	Show how to wake up system form Power-down mode by GPIO interrupt.
I2C_EEPROM	Show how to use I²C interface to access EEPROM.
I2C_Master	Show a Master how to access Slave. This sample code needs to work with I2C_Slave.
I2C_Slave	Show how to set I²C in Slave mode and receive the data from Master. This sample code needs to work with I2C_master.
I2C_Wakeup_Master	Show how to wake up MCU from Power-down mode. This sample code needs to work with I2C_Wakeup_Slave.
I2C_Wakeup_Slave	Show how to wake up MCU from Power-down mode through I2C interface. This sample code needs to work with I2C_Wakeup_Master.
I2S_InternalCODEC	An I²S demo using internal Audio CODEC used to playback the input from line-in or MIC interface.
I2S_Master	Demonstrate how I²S works in Master mode. This sample code needs to work with I2S_Slave.
I2S_MP3PLAYER_SD	A MP3 file player demo using internal Audio CODEC to play back a MP3 file stored in SD card.
I2S_MP3PLAYER_USB	A MP3 file player demo using internal Audio CODEC to play back a MP3 file stored in USB pen drive.

I2S_NAU8822	An I²S demo using NAU8822 Audio CODEC to playback the input from line-in or MIC interface.
I2S_NAU8822_MP3PLAYER_SD	A MP3 file player demo using NAU8822 Audio CODEC to play back a MP3 file stored in SD card.
I2S_NAU8822_MP3PLAYER_USB	A MP3 file player demo using NAU8822 Audio CODEC to play back a MP3 file stored in USB pen drive.
I2S_NAU8822_WAVPLAYER_SD	A WAV file player demo using NAU8822 Audio CODEC o play back a WAV file stored in SD card.
I2S_NAU8822_WAVPLAYER_USB	A WAV file player demo using NAU8822 Audio CODEC to play back a WAV file stored in USB pen drive.
I2S_NAU8822_WAVRECORDER_SD	A WAV file recorder demo using NAU8822 Audio CODEC to record a WAV file and save the file to SD card.
I2S_Slave	Demonstrate how I²S works in Slave mode. This sample code needs to work with I2S_Master.
I2S_WAVPLAYER_SD	A WAV file player demo using internal Audio CODEC to play back a WAV file stored in SD card.
I2S_WAVPLAYER_USB	A WAV file player demo using internal Audio CODEC to play back a WAV file stored in USB pen drive.
I2S_WAVRECORDER_SD	A WAV file recorder demo using internal Audio CODEC to record a WAV file and save the file to SD card.
PWM_Capture	Demonstrate PWM Capture function by using PWM channel 2 to capture the output of PWM channel 0. Please connect PB.10 and PB.12 to execute this code.
PWM_DeadZone	Demonstrate the dead-zone feature with PWM.
RTC_Alarm_Mask_Test	Demonstrate the RTC alarm function. This sample code sets a minute alarm mask after

	execution.
RTC_Alarm_Test	Demonstrate the RTC alarm function. This sample code sets an alarm 10 seconds after execution.
RTC_Time_Display	Demonstrate the RTC function and display the current time on the UART console.
SD_FATFS	Access a SD card formatted in the FAT file system.
SPI_Flash	Access the SPI Flash through a SPI interface.
SPI_Loopback	Demonstrate SPI master loop back transfer. This sample code needs to connect the SPI0_MISO0 pin and the SPI0_MOSI0 pin together. It will compare the received data with transmitted data.
SPI_MasterMode	Demonstrate how to communicate with an off-chip SPI slave device. This sample code needs to work with SPI_SlaveMode.
SPI_SlaveMode	Demonstrate how to communicate with an off-chip SPI master device. This sample code needs to work with SPI_MasterMode.
SPIM_CheckIF	Check which interface SPI Flash is attached to.
SPIM_DMA	Demonstrate how to read/write SPI Flash in SPIM DMA mode.
SPIM_DMM	Demonstrate how to read SPI Flash in SPIM DMM (Direct Memory Map) mode.
SPIM_IO	Demonstrate how to read/write SPI Flash in SPIM (SPI Master) I/O mode.
SPIM_SPIROM	Special notes for code running on SPI Flash: 1. Switch to different clock safely, especially higher system clock. 2. Embed MTP signature in the predefined location for security function.
SYS	1. Demonstrate delay function by systick. 2. Demonstrate core clock switching. 3. Demonstrate how to enable module clock

	and set module clock divider.
SYS_PowerDownConsumption	Demonstrate how to save power consumption in Power-down mode.
Timer_Delay	Demonstrate the usage of TIMER_Delay() API to generate a 1 second delay.
Timer_EventCounter	Use the pin PA.12 to demonstrate the timer event counter function.
Timer_FreeCountingMode	Use the timer pin PA.13 to demonstrate timer free counting mode function. Also display the measured input frequency on the UART console.
Timer_Periodic	Use the timer periodic mode to generate timer interrupt per one second.
Timer_ToggleOut	Demonstrate the timer 0 toggle out function on pin PA.12.
UART_AutoBaudRate_Master	Demonstrate how to use auto baud rate detection function. This sample code needs to work with UART_AutoBaudRate_Slave.
UART_AutoBaudRate_Slave	Demonstrate how to use auto baud rate detection function. This sample code needs to work with UART_AutoBaudRate_Master.
UART_Autoflow_Master	Demonstrate how to transmit and receive data with auto flow control. This sample code needs to work with UART_Autoflow_Slave.
UART_Autoflow_Slave	Demonstrate how to transmit and receive data with auto flow control. This sample code needs to work with UART_Autoflow_Master
UART_IrDA_Master	Demonstrate how to transmit and receive data in UART IrDA mode. This sample code needs to work with UART_IrDA_Slave.
UART_IrDA_Slave	Demonstrate how to transmit and receive data in UART IrDA mode. This sample code needs to work with UART_IrDA_Master.
UART_LIN	Demonstrate how to transmit LIN header and response.

UART_RS485_Master	Demonstrate how to transmit and receive data in UART RS485 mode. This sample code needs to work with UART_RS485_Slave.
UART_RS485_Slave	Demonstrate how to transmit and receive data in UART RS485 mode. This sample code needs to work with UART_RS485_Master.
UART_TxRxFunction	Demonstrate how UART transmits and receives data from PC terminal through RS232 interface.
UART_Wakeup	Demonstrate how to wake up system form Power-down mode by UART interrupt.
USBD_Audio_HeadSet	An UAC1.0/UAC2.0 sample code used to record the sound transmitted to PC and play the sound from PC through the USB interface.
USBD_Audio_Microphone	An UAC1.0/UAC2.0 sample code used to record the sound to PC through the USB interface.
USBD_Audio_Speaker	An UAC1.0/UAC2.0 sample code used to play the sound sent from PC through the USB interface.
USBD_HID_MOUSE	Simulate a USB mouse and draw a circle on the screen.
USBD_HID_Transfer	Demonstrate how to transfer user-defined data – Command / Data Read / Data Write by a HID device.
USBD_Mass_Storage_SD	Implement a mass storage class sample for a SD card reader.
USBD_Mass_Storage_ ShortPacket	Implement a mass storage class sample to demonstrate how to receive a USB short packet.
USBD_Mass_Storage_SRAM	Use internal SRAM as back-end storage media to simulate a 30KB USB pen drive.
USBD_Mass_Storage_ ScatterGather	Demonstrate the usage of USBD DMA scatter gather function.
USBD_VCOM_And_HID	Demonstrate how to implement a USB virtual com port and a HID composite device.

USBD_VCOM_SerialEmulator	Demonstrate how to implement a USB virtual com port device.
USBH_HID	Demonstrate reading inputs from a USB Mouse and displaying the input data (coordinate and button status) on the UART console. This sample includes a USB Mouse driver which is based on the HID driver.
USBH_UAC_HID	Demonstrate how to use an UAC+HID device.
USBH_UMAS	Use a USB Host core driver, a USB mass storage driver, and a FATFS file system to show a disk access shell interface.
WDT_Polling	Use polling mode to check WDT time-out state and reset WDT after a time-out occurs.
WDT_Wakeup	Use WDT to wake up system from Power-down mode periodically.
WWDT_Reload	Demonstrate the WWDT counter reload function.

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.