M058S CMSIS BSP Directory

Directory Introduction for 32-bit NuMicro™ Family

Directory Information

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

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

www.nuvoton.com

TABLE OF CONTENTS

1 Document

Revision History.pdf	Show all the revision history about specific BSP.
NuMicro M058S BSP Driver Reference Guide.chm	Describe the definition, input and output of each API.

2. Library

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

3 Sampel Code

NuTiny-EVB	The sample codes for NuTiny-EVB board.
Hard_Fault_Sample	Show hard fault information when hard fault happened.
ISP	Sample codes for In-System-Programming.
Template	Software Development Template.
Semihost	Show how to debug with semi-host message print.
RegBased	The sample codes which access control registers directly.
StdDriver	M058S Driver Samples

4 SampleCode\ISP

ISP_I2C	In-System-Programming Sample code through I2C 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\RegBased

System Manager (SYS)

SYS	Change system clock to different PLL frequency and output system clock from CLKO pin.
SYS_PowerDown_MinCurrent	Demonstrate how to minimize power consumption when entering power down mode.

Fash Memory Controller (FMC)

FMC_IAP	Show how to call LDROM functions from APROM. The code in APROM will look up the table at 0x100E00 to get the address of function of LDROM and call the function.
FMC_MultiBoot_SwReset	Show how to use software reset to implement multi-boot system to boot from different applications in APROM.
FMC_RW	Show how to read/program embedded flash by ISP function.

General Purpose I/O (GPIO)

GPIO_EINTAndDebounce	Show the usage of GPIO external interrupt function and de-bounce function.
GPIO_INT	Show the usage of GPIO interrupt function.
GPIO_OutputInput	Show how to set GPIO pin mode and use pin data input/output control.
GPIO_PowerDown	Show how to wake up system from Power-down mode by GPIO interrupt.

Timer Controller (TIMER)

TIMER_Capture	Show how to use the timer1 capture function to capture timer1 counter value.
TIMER_Counter	Implement timer1 event counter function to count the external input event.

TIMER_PeriodicINT	Implement timer counting in periodic mode.
TIMER_PowerDown	Use timer0 toggle-output time-out interrupt event to wake up system.

Watchdog Timer (WDT)

WDT_PowerDown	Use WDT time-out interrupt event to wake-up system.
WDT_TimeoutINT	Implement periodic WDT time-out interrupt event.
WDT_TimeoutReset	Show how to generate time-out reset system event while WDT time-out reset delay period expired.

Window Watchdog Timer (WWDT)

WWDT_CompareINT	Show how to reload the WWDT counter value.

PWM Generator and Capture Timer (PWM)

PWM_Capture	Capture the PWMA Channel 1 waveform by PWMA Channel 2.
PWM_DeadZone	Demonstrate how to use PWM Dead Zone function.
PWM_DoubleBuffer	Change duty cycle and period of output waveform by PWM Double Buffer function.

UART Interface Controller (UART)

UART_Autoflow_Master	Transmit and receive data with auto flow control. This sample code needs to work with UART_Autoflow_Slave.
UART_Autoflow_Slave	Transmit and receive data with auto flow control. This sample code needs to work with UART_Autoflow_Master.
UART_IrDA_Master	Transmit and receive data in UART IrDA mode. This sample code needs to work with UART_IrDA_Slave.
UART_IrDA_Slave	Transmit and receive data in UART IrDA mode. This sample

	code needs to work with UART_IrDA_Master.
UART_LIN	Transmit LIN frame including header and response in UART LIN mode.
UART_RS485_Master	Transmit and receive data in UART RS485 mode. This sample code needs to work with UART_RS485_Slave.
UART_RS485_Slave	Transmit and receive data in UART RS485 mode. This sample code needs to work with UART_RS485_Master.
UART_TxRx_Function	Transmit and receive data from PC terminal through RS232 interface.
UART_Wakeup	Show how to wake up system from Power-down mode by UART interrupt.

Serial Peripheral Interface (SPI)

SPI_Loopback	Implement SPI Master loop back transfer. This sample code needs to connect MISO pin and MOSI pin together. It will compare the received data with transmitted data.
SPI_MasterFifoMode	Configure SPI0 as Master mode and demonstrate how to communicate with an off-chip SPI Slave device with FIFO mode. This sample code needs to work with SPI_SlaveFifoMode sample code.
SPI_SlaveFifoMode	Configure SPI0 as Slave mode and demonstrate how to communicate with an off-chip SPI Master device with FIFO mode. This sample code needs to work with SPI_MasterFifoMode sample code.

I^{2}C Serial Interface Controller (I^{2}C)

I2C_EEPROM	Demonstrate how to access EEPROM through a I2C interface
I2C_GCMode_Master	Demonstrate how a Master uses I2C address 0x0 to write data to I2C Slave. This sample code needs to work with I2C_GCMode_Slave.
I2C_GCMode_Slave	Demonstrate how to receive Master data in GC (General Call) mode. This sample code needs to work with

	I2C_GCMode_Master.
I2C_Master	Demonstrate how a Master accesses a Slave. This sample code needs to work with I2C_Slave.
I2C_Slave	Demonstrate how to set I2C in slave mode to receive data from a Master. This sample code needs to work with I2C_Master.
I2C_Wakeup_Master	Demonstrate how to wake-up MCU from power-down. Needs to work with I2C_Wakeup_Slave sample code.
I2C_Wakeup_Slave	Demonstrate how to set I^{2}C to wake-up MCU from power-down mode. Needs to work with I2C_Wakeup_Master sample code.

Analog-to-Digital Converter (ADC)

ADC_BurstMode	Demonstrate A/D conversion with burst mode. In burst mode, ADC will sample and convert a specified channel continuously and store the conversion result in FIFO buffers.
ADC_ContinuousScanMode	Perform A/D Conversion with ADC continuous scan mode.
ADC_PwmTrigger	Measure AVDD voltage by ADC.
ADC_MeasureAVDD	Demonstrate how to trigger ADC by PWM.
ADC_ResultMonitor	Monitor the conversion result of channel 2 by the digital compare function.
ADC_SingleCycleScanMode	Perform A/D Conversion with ADC single cycle scan mode.
ADC_SingleMode	Perform A/D Conversion with ADC single mode.

6 SampleCode\StdDriver

System Manager (SYS)

SYS	Change system clock to different PLL frequency and output system clock from CLKO pin..
SYS_PowerDown_MinCurrent	Demonstrate how to minimize power consumption when entering power down mode.

Flash Memory Controller (FMC)

FMC_IAP	Show how to call LDROM functions from APROM. The code in APROM will look up the table at 0x100E00 to get the address of function of LDROM and call the function.
FMC_RW	Show how to read/program embedded flash by ISP function.

General Purpose I/O (GPIO)

GPIO_EINTAndDebounce	Show the usage of GPIO external interrupt function and de-bounce function.
GPIO_INT	Show the usage of GPIO interrupt function.
GPIO_OutputInput	Show how to set GPIO pin mode and use pin data input/output control.
GPIO_PowerDown	Show how to wake up system from Power-down mode by GPIO interrupt.

Timer Controller (TIMER)

TIMER_Capture	Show how to use the timer1 capture function to capture timer1 counter value.
TIMER_Counter	Implement timer1 event counter function to count the external input event.
TIMER_PeriodicINT	Implement timer counting in periodic mode.

TIMER_PowerDown	Use timer-0 toggle-output interrupt event to wake-up system.

Watchdog Timer (WDT)

WDT_PowerDown	Use WDT time-out interrupt event to wake-up system.
WDT_TimeoutINT	Implement periodic WDT time-out interrupt event.
WDT_TimeoutReset	Show how to generate time-out reset system event while WDT time-out reset delay period expired.

Window Watchdog Timer (WWDT)

WWDT_CompareINT	Show how to reload the WWDT counter value.

PWM Generator and Capture Timer (PWM)

PWM_Capture	Capture the PWMA Channel 1 waveform by PWMA Channel 2.
PWM_DeadZone	Demonstrate how to use PWM Dead Zone function.
PWM_DoubleBuffer	Change duty cycle and period of output waveform by PWM Double Buffer function.

UART Interface Controller (UART)

UART_Autoflow_Master	Transmit and receive data with auto flow control. This sample code needs to work with UART_Autoflow_Slave.
UART_Autoflow_Slave	Transmit and receive data with auto flow control. This sample code needs to work with UART_Autoflow_Master.
UART_IrDA_Master	Transmit and receive data in UART IrDA mode. This sample code needs to work with UART_IrDA_Slave.
UART_IrDA_Slave	Transmit and receive data in UART IrDA mode. This sample code needs to work with UART_IrDA_Master.

UART_LIN	Transmit LIN frame including header and response in UART LIN mode.
UART_RS485_Master	Transmit and receive data in UART RS485 mode. This sample code needs to work with UART_RS485_Slave.
UART_RS485_Slave	Transmit and receive data in UART RS485 mode. This sample code needs to work with UART_RS485_Master.
UART_TxRx_Function	Transmit and receive data from PC terminal through RS232 interface.
UART_Wakeup	Show how to wake up system from Power-down mode by UART interrupt.

Serial Peripheral Interface (SPI)

SPI_Loopback	Implement SPI Master loop back transfer. This sample code needs to connect MISO pin and MOSI pin together. It will compare the received data with transmitted data.
SPI_MasterFIFOMode	Configure SPI0 as Master mode and demonstrate how to communicate with an off-chip SPI Slave device with FIFO mode. This sample code needs to work with SPI_SlaveFifoMode sample code.
SPI_SlaveFIFOMode	Configure SPI0 as Slave mode and demonstrate how to communicate with an off-chip SPI Master device with FIFO mode. This sample code needs to work with SPI_MasterFifoMode sample code.

I^{2}C Serial Interface Controller (I^{2}C)

I2C_EEPROM	Demonstrate how to access EEPROM through a I2C interface
I2C_GCMode_Master	Demonstrate how a Master uses I2C address 0x0 to write data to I2C Slave. This sample code needs to work with I2C_GCMode_Slave.
I2C_GCMode_Slave	Demonstrate how to receive Master data in GC (General Call) mode. This sample code needs to work with I2C_GCMode_Master.

I2C_Master	Demonstrate how a Master accesses Slave. This sample code needs to work with I2C_Slave.
I2C_Slave	Demonstrate how to set I2C in slave mode to receive the data from a Master. This sample code needs to work with I2C_Master.
I2C_Wakeup_Master	Demonstrate how to wake-up MCU from power-down. Needs to work with I2C_Wakeup_Slave sample code.
I2C_Wakeup_Slave	Demonstrate how to set I^2C to wake-up MCU from power-down mode. Needs to work with I2C_Wakeup_Master sample code.

Analog-to-Digital Converter (ADC)

ADC_BurstMode	Demonstrate A/D conversion with burst mode. In burst mode, ADC will sample and convert a specified channel continuously and store the conversion result in FIFO buffers.
ADC_ContinuousScanMode	Perform A/D Conversion with ADC continuous scan mode.
ADC_PwmTrigger	Demonstrate how to trigger ADC by PWM.
ADC_MeasureAVDD	Measure AVDD voltage by ADC.
ADC_ResultMonitor	Monitor the conversion result of channel 2 by the digital compare function.
ADC_SingleCycleScanMode	Perform A/D Conversion with ADC single cycle scan mode.
ADC_SingleMode	Perform A/D Conversion with ADC single mode.

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.