M0564 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.

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

www.nuvoton.com

1 Document Information

Revision History.pdf	Show all the revision history about specific BSP.
NuMicro M0564 Series Driver Reference Guide.chm	Describe the definition, input and output of each API.

2 Library Information

CMSIS	CMSIS 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	Show hard fault information when hard fault happened.
ISP	Sample codes for In-System-Programming.
Template	Software Development Template.
Semihost	A sample code to show how to debug with semihost message print.
RegBased	The sample codes which access control registers directly.
StdDriver	M0564 Series 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

ACMP	Demonstrate how ACMP[1] works with internal band-gap voltage.
ACMP_Wakeup	Show how to wake up MCU from Power-down mode by ACMP wake-up function.
ADC_ContinuousScanMode	Perform A/D Conversion with ADC continuous scan mode.
ADC_PDMA_SingleCycleScanMode	Perform A/D Conversion with ADC single cycle scan mode and transfer result by PDMA.
ADC_PwmTrigger	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.
CLK_ClockDetector	Show the usage of clock fail detector and clock frequency monitor function.
CRC_CCITT	Implement CRC in CRC-CCITT mode and get the CRC checksum result.
CRC_CRC32	Implement CRC in CRC-32 mode with PDMA transfer.
CRC_CRC8	Implement CRC in CRC-8 mode and get the CRC checksum result.
EBI_NOR	Configure EBI interface to access MX29LV320T (NOR Flash) on EBI interface.
EBI_SRAM	Configure EBI interface to access BS616LV4017 (SRAM) on EBI interface.
FMC_ExeInSRAM	Implement a code and execute in SRAM to program embedded Flash(Suuport KEIL MDK only).
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	Implement a multi-boot system to boot from different applications in APROM. A LDROM code and 4 APROM code are implemented in this sample code.
FMC_RW	Demonstrate how to read/program embedded flash by ISP function.
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.
HDIV	Show how to use divider API and how to use hardware divider by control registers.
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_Loopback	Demonstrate how a Master access Slave.
I2C_Master	Demonstrate how a Master access Slave. This sample code needs to work with I2C_Slave.
I2C_Master_PDMA	Demonstrate how a Master access Slave using PDMA TX mode and PDMA RX mode.
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_Slave_PDMA	Demonstrate how a Slave use PDMA Rx mode receive data from a Master.
I2C_Wakeup_Slave	Demonstrate how to set I2C to wake up MCU from Power-down mode. This sample code needs to work with I2C_Master.
I2S_Master	Configure SPI1 as I2S Master mode and demonstrate how I2S works in Master mode. This sample code needs to work with I2S_Slave sample code.
I2S_PDMA_NAU8822	This is a I2S demo with PDMA function connected with NAU8822 codec.
I2S_PDMA_Play	This is a I2S demo for playing data and demonstrate how I2S works with PDMA.
I2S_PDMA_PlayRecord	This is a I2S demo for playing and recording data with PDMA function.
I2S_PDMA_Record	This is a I2S demo for recording data and demonstrate how I2S works with PDMA.
I2S_Slave	Configure SPI1 as I2S Slave mode and demonstrate how I2S works in Slave mode. This sample code needs to work with I2S_Master sample code.
PDMA	Use PDMA channel 2 to transfer data from memory to memory.
PDMA_ScatterGather_PingPongBuffer	Use PDMA to implement Ping-Pong buffer by scatter-gather mode(memory to memory).
PDMA_Scatter_Gather	Use PDMA channel 4 to transfer data from memory to memory by scatter-gather mode.
PWM_Capture	Capture the PWM1 Channel 0 waveform by PWM1 Channel 2.
PWM_DeadZone	Demonstrate how to use PWM Dead Zone function.
PWM_DoubleBuffer_PeriodLoadingMode	Change duty cycle and period of output waveform by PWM Double Buffer function(Period loading mode).

PWM_DutySwitch	Change duty cycle of output waveform by configured period.
PWM_OutputWaveform	Demonstrate how to use PWM output waveform.
PWM_PDMA_Capture	Capture the PWM1 Channel 0 waveform by PWM1 Channel 2, and use PDMA to transfer captured data.
PWM_SyncStart	Demonstrate how to use PWM counter synchronous start function.
RTC_AlarmWakeup	Use RTC alarm interrupt event to wake up system.
RTC_TimeAndTick	Show the current RTC data/time per tick.
SCUART_TxRx	Show Smartcard UART by connecting PA.0 and PA.1 pins.
SC_ReadATR	Read the Smartcard ATR from SC0 port.
SPI_Loopback	Implement SPI Master loop back transfer. This sample code needs to connect SPI0_MISO0 pin and SPI0_MOSI0 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_PDMA_LoopTest	Demonstrate SPI data transfer with PDMA. SPI0 will be configured as Master mode and SPI1 will be configured as Slave mode. Both TX PDMA function and RX PDMA function will be enabled.
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.
SYS_BODWakeup	Show how to wake up system form Power-down mode by brown-out detector interrupt.
SYS_PLLClockOutput	Change system clock to different PLL frequency and output system clock from CLKO pin.

SYS_VoltageDetector	Show how to use voltage detector to detect pin input voltage.
TIMER_CaptureCounter	Show how to use the timer2 capture function to capture timer2 counter value.
TIMER_EventCounter	Show how to use the timer2 capture function to capture timer2 counter value.
TIMER_PeriodicINT	Implement timer counting in periodic mode.
TIMER_PWM_ChangeDuty	Change duty cycle and period of output waveform in PWM down count type.
TIMER_PWM_DeadTime	Demonstrate Timer PWM Complementary mode and Dead-Time function.
TIMER_PWM_OuputWaveform	Demonstrate output different duty waveform in Timer0~Timer3 PWM.
TIMER_TimeoutWakeup	Use timer0 periodic time-out interrupt event to wake up system.
UART_AutoBaudRate_Master	Show how to use auto baud rate detection function. This sample code needs to work with UART_AutoBaudRate_Slave.
UART_AutoBaudRate_Slave	Show how to use auto baud rate detection function. This sample code needs to work with UART_AutoBaudRate_Master.
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_PDMA	Transmit and receive UART data with PDMA.
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_TxRxFunction	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.
USCI_I2C_EEPROM	Demonstrate how to access EEPROM through a USCI_I2C interface
USCI_I2C_Loopback	Demonstrate how a Master access Slave.
USCI_I2C_Loopback_10bit	Demonstrate how a Master use 10-bit addressing access Slave.
USCI_I2C_Master	Demonstrate how a Master access Slave. This sample code needs to work with I2C_Slave.
USCI_I2C_Master_10bit	Demonstrate how a Master use 10-bit addressing access Slave. This sample code needs to work with I2C_Slave.
USCI_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.
USCI_I2C_Slave_10bit	Demonstrate how to set I2C in 10-bit addressing slave mode to receive the data from a Master. This sample code needs to work with I2C_Master.
USCI_I2C_Wakeup_Slave	Demonstrate how to set I2C to wake up MCU from Power-down mode. This sample code needs to work with I2C_Master.
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_AutoBaudRate_Master	Show how to use auto baud rate detection function. This sample code needs to work with USCI_UART_AutoBaudRate_Slave.
USCI_UART_AutoBaudRate_Slave	Show how to use auto baud rate detection function. This sample code needs to work with USCI_UART_AutoBaudRate_Master.
USCI_UART_Autoflow_Master	Transmit and receive data with auto flow control. This sample code needs to work with USCI_UART_Autoflow_Slave.
USCI_UART_Autoflow_Slave	Transmit and receive data with auto flow control. This sample code needs to work with USCI_UART_Autoflow_Master.
USCI_UART_RS485_Master	Transmit and receive data in RS485 mode. This sample code needs to work with USCI_UART_RS485_Slave.
USCI_UART_RS485_Slave	Transmit and receive data in RS485 mode. This sample code needs to work with USCI_UART_RS485_Master.
USCI_UART_TxRxFunction	Transmit and receive data from PC terminal through RS232 interface.
USCI_UART_Wakeup	Show how to wake up system from Power-down mode by USCI interrupt in UART mode.
WDT_TimeoutWakeupAndReset	Implement WDT time-out interrupt event to wake up system and generate time-out reset system event while WDT time-out reset delay period expired.
WWDT_CompareINT	Show how to reload the WWDT counter value.

1. Analog Comparator (ACMP).

6 \SampleCode\StdDriver

ACMP	Demonstrate how ACMP[1] works with internal band-gap voltage.
ACMP_Wakeup	Show how to wake up MCU from Power-down mode by ACMP wake-up function.
ADC_ContinuousScanMode	Perform A/D Conversion with ADC continuous scan mode.
ADC_PDMA_SingleCycleScanMode	Perform A/D Conversion with ADC single cycle scan mode and transfer result by PDMA.
ADC_PwmTrigger	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.
CLK_ClockDetector	Show the usage of clock fail detector and clock frequency monitor function.
CRC_CCITT	Implement CRC in CRC-CCITT mode and get the CRC checksum result.
CRC_CRC32	Implement CRC in CRC-32 mode with PDMA transfer.
CRC_CRC8	Implement CRC in CRC-8 mode and get the CRC checksum result.
EBI_NOR	Configure EBI interface to access MX29LV320T (NOR Flash) on EBI interface.
EBI_SRAM	Configure EBI interface to access BS616LV4017 (SRAM) on EBI interface.
FMC_ExeInSRAM	Implement a code and execute in SRAM to program embedded Flash(Suuport KEIL MDK only).
FMC_IAP	Show how to set VECMAP to LDROM and reboot to LDROM from APROM.

FMC_RW	Demonstrate how to read/program embedded flash by ISP function.
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.
HDIV	Show how to use divider API and how to use hardware divider by control registers.
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_Loopback	Demonstrate how a Master access Slave.
I2C_Master	Demonstrate how a Master access Slave. This sample code needs to work with I2C_Slave.
I2C_Master_PDMA	Demonstrate how a Master access Slave using PDMA TX mode and PDMA RX mode.
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_Slave_PDMA	Demonstrate how a Slave use PDMA Rx mode receive data from a Master.
I2C_Wakeup_Slave	Demonstrate how to set I2C to wake up MCU from Power-down mode. This sample code needs to work with

	I2C_Master.
I2S_Master	Configure SPI1 as I2S Master mode and demonstrate how I2S works in Master mode. This sample code needs to work with I2S_Slave.
I2S_PDMA_NAU8822	This is a I2S demo with PDMA function connected with NAU8822 codec.
I2S_PDMA_Play	This is a I2S demo for playing data and demonstrate how I2S works with PDMA.
I2S_PDMA_PlayRecord	This is a I2S demo for playing and recording data with PDMA function.
I2S_PDMA_Record	This is a I2S demo for recording data and demonstrate how I2S works with PDMA.
I2S_Slave	Configure SPI1 as I2S Slave mode and demonstrate how I2S works in Slave mode. This sample code needs to work with I2S_Master.
PDMA	Use PDMA channel 2 to transfer data from memory to memory.
PDMA_ScatterGather_PingPongBuffer	Use PDMA to implement Ping-Pong buffer by scatter-gather mode(memory to memory).
PDMA_Scatter_Gather	Use PDMA channel 4 to transfer data from memory to memory by scatter-gather mode.
PWM_Capture	Capture the PWM1 Channel 0 waveform by PWM1 Channel 2.
PWM_DeadZone	Demonstrate how to use PWM Dead Zone function.
PWM_DoubleBuffer_PeriodLoadingMode	Change duty cycle and period of output waveform by PWM Double Buffer function(Period loading mode).
PWM_DutySwitch	Change duty cycle of output waveform by configured period.
PWM_OutputWaveform	Demonstrate how to use PWM output waveform.
PWM_PDMA_Capture	Capture the PWM1 Channel 0 waveform by PWM1 Channel 2, and use PDMA to transfer captured data.

PWM_SyncStart	Demonstrate how to use PWM counter synchronous start function.
RTC_AlarmWakeup	Use RTC alarm interrupt event to wake up system.
RTC_TimeAndTick	Show the current RTC data/time per tick.
SCUART_TxRx	Show Smartcrd UART by connecting PA.0 and PA.1 pins.
SC_ReadATR	Read the Smartcard ATR from SC0 port.
SC_ReadSimPhoneBook	Read the SIM phone book from SC0 port.
SPI_Loopback	Implement SPI Master loop back transfer. This sample code needs to connect MISO_0 pin and MOSI_0 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_PDMA_LoopTest	Demonstrate SPI data transfer with PDMA. SPI0 will be configured as Master mode and SPI1 will be configured as Slave mode. Both TX PDMA function and RX PDMA function will be enabled.
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.
SYS_BODWakeup	Show how to wake up system form Power-down mode by brown-out detector interrupt.
SYS_PLLClockOutput	Change system clock to different PLL frequency and output system clock from CLKO pin.
SYS_VoltageDetector	Show how to use voltage detector to detect pin input voltage.
TIMER_ACMPTrigger	Show how to use ACMP0 to trigger Timer capture event.
TIMER_CaptureCounter	Show how to use the timer2 capture function to capture timer2 counter value.

TIMER_Delay	Show how to use timer0 to create various delay time.
TIMER_EventCounter	Implement timer1 event counter function to count the external input event.
TIMER_InterTimerTriggerMode	Demonstrate how to use Inter-Timer trigger function.
TIMER_PeriodicINT	Implement timer counting in periodic mode.
TIMER_PWM_Brake	Generate Timer brake event by Timer brake pin.
TIMER_PWM_ChangeDuty	Change duty cycle and period of output waveform in PWM down count type.
TIMER_PWM_DeadTime	Demonstrate Timer PWM Complementary mode and Dead-Time function.
TIMER_PWM_OuputWaveform	Demonstrate output different duty waveform in Timer0~Timer3 PWM.
TIMER_TimeoutWakeup	Use timer0 periodic time-out interrupt event to wake up system.
TIMER_ToggleOut	Implement timer counting in toggle-output mode.
UART_AutoBaudRate_Master	Show how to use auto baud rate detection function. This sample code needs to work with UART_AutoBaudRate_Slave.
UART_AutoBaudRate_Slave	Show how to use auto baud rate detection function. This sample code needs to work with UART_AutoBaudRate_Master.
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_PDMA	Transmit and receive UART data with PDMA.
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_TxRxFunction	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.
USCI_I2C_EEPROM	Demonstrate how to access EEPROM through a USCI_I2C interface.
USCI_I2C_Loopback	Demonstrate how a Master access Slave.
USCI_I2C_Loopback_10bit	Demonstrate how a Master use 10-bit addressing access Slave.
USCI_I2C_Master	Demonstrate how a Master access Slave. This sample code needs to work with I2C_Slave.
USCI_I2C_Master_10bit	Demonstrate how a Master use 10-bit addressing access Slave. This sample code needs to work with I2C_Slave.
USCI_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.
USCI_I2C_Slave_10bit	Demonstrate how to set I2C in 10-bit addressing slave mode to receive the data from a Master. This sample code needs to work with I2C_Master.
USCI_I2C_Wakeup_Slave	Demonstrate how to set I2C to wake up MCU from Power-down mode. This sample code needs to work with I2C_Master.

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_AutoBaudRate_Master	Show how to use auto baud rate detection function. This sample code needs to work with USCI_UART_AutoBaudRate_Slave.
USCI_UART_AutoBaudRate_Slave	Show how to use auto baud rate detection function. This sample code needs to work with USCI_UART_AutoBaudRate_Master.
USCI_UART_Autoflow_Master	Transmit and receive data with auto flow control. This sample code needs to work with USCI_UART_Autoflow_Slave.
USCI_UART_Autoflow_Slave	Transmit and receive data with auto flow control. This sample code needs to work with USCI_UART_Autoflow_Master.
USCI_UART_RS485_Master	Transmit and receive data in RS485 mode. This sample code needs to work with USCI_UART_RS485_Slave.
USCI_UART_RS485_Slave	Transmit and receive data in RS485 mode. This sample code needs to work with USCI_UART_RS485_Master.
USCI_UART_TxRxFunction	Transmit and receive data from PC terminal through RS232 interface.
USCI_UART_Wakeup	Show how to wake up system from Power-down mode by USCI interrupt in UART mode.
WDT_TimeoutWakeupAndReset	Implement WDT time-out interrupt event to wake up system and generate time-out reset system event while WDT time-out reset delay period expired.

WWDT_CompareINT	Show how to reload the WWDT counter value.

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.