M451 Series Board Supporting Package Directory Introduction

Rev.3.00.003

Directory Information

Document	Driver reference manual and revision history.
Library	Driver header and source files.
SampleCode	Driver sample code.
ThirdParty	Library from third party

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.
SmartcardLib	Library for CCID smart card reader.
UsbHostLib	Library for USB Host.

Sample Code Information

\SampleCode\CardReader	CCID[1] smart card reader sample code.
\SampleCode\FreeRTOS	Simple FreeRTOS™ demo code.
\SampleCode\Hard_Fault_ Sample	Show hard fault information when hard fault happened.
\SampleCode\NuTiny-SDK-M453	Sample code for M453 Tiny Board.
\SampleCode\Template	Software Development Template.
\SampleCode\Semihost	The sample code to show how to debug with semihost message print.
\SampleCode\RegBased	The sample code able to access control registers directly.
\SampleCode\StdDriver	M451 Series Driver Samples

1. Circuit card interface device (CCID) is USB device that interface with integrated circuit cards.

\SampleCode\NuTiny-SDK-M453

LED	Toggle PC.9 to turn on / off the board LED.

\SampleCode\RegBased

ACMP	Demonstrate how ACMP[1] works with internal band-gap voltage.
ACMP_Wakeup	Demonstrate how to wake up from Power-down mode by ACMP wake-up function.
CAN_Set_MaskFilter	Demonstrate how to use MaskFilter to receive message in Normal mode. This sample code needs to work with CAN_Test_MaskFilter.
CAN_Test_MaskFilter	Demonstrate how to use message object No.1 to send message objects (ID=0x700~0x70F). This sample code needs to work with CAN_Set_MaskFilter.
CLK_ClockDetector	Demonstrate how to use clock fail detector and clock frequency monitor function.
CRC_CCITT	Perform CRC-CCITT operation and get the CRC checksum result.
CRC_CRC8	Perform CRC-8 operation and get the CRC checksum result.
DAC_PWMTrigger	Demonstrate how to trigger DAC by PWM.
DAC_TimerTrigger	Demonstrate how to trigger DAC by timer.
DAC_WriteDataTrigger	Demonstrate how to trigger DAC by writing DAC_DAT register.
EADC_ADINT_Trigger	Demonstrate how to use ADINT interrupt to do the ADC continuous scan conversion.
EADC_PWM_Trigger	Demonstrate how to trigger ADC by PWM.
EADC_ResultMonitor	Demonstrate how to use the digital compare function to monitor the conversion result of channel 2.

EADC_SWTRG_Trigger	Demonstrate how to trigger ADC by writing EADC_SWTRG register.
EADC_Timer_Trigger	Demonstrate how to trigger ADC by timer.
EBI_NOR	Demonstrate how to access MX29LV320T (NOR Flash) through EBI bus.
EBI_SRAM	Demonstrate how to access BS616LV4017 (SRAM) through EBI bus.
FMC_ExeInSRAM	Demonstrate how to execute code in SRAM and program embedded Flash. (Support KEIL® MDK Only)
FMC_IAP	Demonstrate 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	Demonstrate how to implement 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	Demonstrate how to use GPIO external interrupt function and de-bounce function.
GPIO_INT	Demonstrate how to use GPIO interrupt function.
GPIO_OutputInput	Demonstrate how to set GPIO pin mode and use pin data input/output control.
GPIO_PowerDown	Demonstrate how to wake up system form Power-down mode by GPIO interrupt.
I2C_EEPROM	Demonstrate how to access EEPROM by 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 access Slave. This sample code needs to work with I2C_SLAVE.
I2C_Slave	Demonstrate how to set I2C in Slave mode to receive the data of a Master. This sample code needs to work with I2C_MASTER.
I2C_SMBus	Demonstrate how to control I2C SMBus and use SMBus protocol between Host and Slave.
I2C_Wakeup_Master	Demonstrate how to wake up MCU from power-down. This sample code needs to work with I2C_Wakeup_Slave.
I2C_Wakeup_Slave	Demonstrate how to set I2C to wake up MCU from Power-down mode. This sample code needs to work with I2C_Wakeup_Master.
I2S_Master	Demonstrate how I2S works in Master mode. This sample code needs to work with I2S_Slave.
I2S_Slave	Demonstrate how I2S works in Slave mode. This sample code needs to work with I2S_Master.
PDMA	Demonstrate how to use PDMA channel 2 to transfer data from memory to memory.
PDMA_Scatter_Gather	Demonstrate how to use PDMA channel 5 to transfer data from memory to memory by scatter-gather mode.
PWM_Capture	Demonstrate how to use PWM1 Channel 2 to capture the PWM1 Channel 0 waveform.
PWM_DeadZone	Demonstrate how to use PWM Dead Zone function.
PWM_DoubleBuffer	Demonstrate how to use PWM Double Buffer function to change duty cycle and period of output waveform.
RTC_AlarmWakeup	Demonstrate how to use RTC alarm interrupt event to wake up system.

RTC_SpareRegisterRW	Demonstrate how to access RTC spare registers.
RTC_TimeAndTick	Demonstrate how to get the current RTC data/time per tick.
SC_ReadATR	Demonstrate how to get smart card ATR data.
SCUART_TxRx	Demonstrate how to use smartcard interface UART mode to print "Hello World!"
SPI_Loopback	Demonstrate 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_MasterMode	Demonstrate how to communicate with an off-chip SPI slave device. This sample code needs to work with SPI_SlaveMode.
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_SlaveMode	Demonstrate how to communicate with an off-chip SPI master device. This sample code needs to work with SPI_MasterMode.
SYS_BODWakeup	Demonstrate how to wake up system form Power-down mode by brown-out detector interrupt.
SYS_PLLClockOutput	Demonstrate how to change system clock to different PLL frequency and output system clock from CLKO pin.
TIMER_CaptureCounter	Demonstrate how to use the timer2 capture event to capture timer2 counter value.
TIMER_EventCounter	Demonstrate how to use timer2 counter input function to count the input event.
TIMER_PeriodicINT	Demonstrate how to perform timer counting in Periodic mode.
TIMER_TimeoutWakeup	Demonstrate how to use timer0 periodic timeout interrupt

	event to wake up system.
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 transmit and receive data from PC terminal through RS232 interface.
UART_Wakeup	Demonstrate how to wake up system form Power-down mode by UART interrupt.
WDT_TimeoutWakeupAndReset	Demonstrate how to use 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	Select one WWDT window compare value to generate window compare match interrupt event.

1. Analog Comparator (ACMP).

\SampleCode\StdDriver

ACMP	Demonstrate how ACMP works with internal band-gap voltage.
ACMP_Wakeup	Demonstrate how to wake up from Power-down mode by ACMP wake-up function.
CAN_BasicMode_Receive	Demonstrate how to receive message in Basic mode. This sample code needs to work with CAN_BasicMode_Transmit.
CAN_BasicMode_Transmit	Demonstrate how to transmit message in Basic mode. This sample code needs to work with CAN_BasicMode_Receive.
CAN_NormalMode_Receive	Demonstrate how to receive message in Normal mode. This sample code needs to work with CAN_NormalMode_Transmit.
CAN_NormalMode_Transmit	Demonstrate how to transmit message in Normal mode. This sample code needs to work with CAN_NormalMode_Receive.
CAN_Wakeup	Demonstrate how to wake up system form Power-down mode by detecting a transition.
CLK_ClockDetector	Demonstrate how to use clock fail detector and clock frequency monitor function.
CRC_CCITT	Perform CRC-CCITT operation and get the CRC checksum result.
CRC_CRC8	Perform CRC-8 operation and get the CRC checksum result.
DAC_PWMTrigger	Demonstrate how to trigger DAC by PWM.

DAC_SoftwareTrigger	Demonstrate how to trigger DAC conversion by software method.
DAC_TimerTrigger	Demonstrate how to trigger DAC by timer.
DSP_FFT	Demonstrate how to call ARM CMSIS DSP library to calculate FFT.
EADC_ADINT_Trigger	Demonstrate how to use ADINT interrupt to do the ADC continuous scan conversion.
EADC_PWM_Trigger	Demonstrate how to trigger ADC by PWM.
EADC_ResultMonitor	Demonstrate how to use the digital compare function to monitor the conversion result of channel 2.
EADC_SWTRG_Trigger	Demonstrate how to trigger ADC by writing EADC_SWTRG register.
EADC_Timer_Trigger	Demonstrate how to trigger ADC by timer.
EBI_NOR	Demonstrate how to access MX29LV320T (NOR Flash) through EBI bus.
EBI_SRAM	Demonstrate how to access BS616LV4017 (SRAM) through EBI bus.
FMC_ExeInSRAM	Demonstrate how to execute code in SRAM and program embedded Flash. (Support KEIL® MDK Only)
FMC_IAP	Demonstrate how to reboot to LDROM functions from APROM. This sample code set VECMAP to LDROM and reset to re-boot to LDROM.
FMC_RW	Demonstrate how to read/program embedded flash by ISP function.
GPIO_EINTAndDebounce	Demonstrate how to use GPIO external interrupt function and de-bounce function.
GPIO_INT	Demonstrate how to use GPIO interrupt function.
GPIO_OutputInput	Demonstrate how to set GPIO pin mode and use pin data

	input/output control.
GPIO_PowerDown	Demonstrate how to wake up system form Power-down mode by GPIO interrupt.
I2C_EEPROM	Demonstrate how to access EEPROM by 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 access Slave. This sample code needs to work with I2C_SLAVE.
I2C_Slave	Demonstrate how to set I2C in Slave mode to receive the data of a Master. This sample code needs to work with I2C_MASTER.
I2C_SMBus	Demonstrate how to control I2C SMBus and use SMBus protocol between Host and Slave.
I2C_Wakeup_Master	Demonstrate how to wake up MCU from power-down. This sample code needs to work with I2C_Wakeup_Slave.
I2C_Wakeup_Slave	Demonstrate how to set I2C to wake up MCU from power-down mode. This sample code needs to work with I2C_Wakeup_Master.
I2S_Master	Demonstrate how I2S works in Master mode. This sample code needs to work with I2S_Slave.
I2S_Slave	Demonstrate how I2S works in Slave mode. This sample code needs to work with I2S_Master.
PDMA	Demonstrate how to use PDMA channel 2 to transfer data from memory to memory.
PDMA_Scatter_Gather	Demonstrate how to use PDMA channel 5 to transfer data from memory to memory by scatter-gather mode.

PWM_Capture	Demonstrate how to use PWM1 Channel 2 to capture the PWM1 Channel 0 waveform.
PWM_DeadZone	Demonstrate how to use PWM Dead Zone function.
PWM_DoubleBuffer	Demonstrate how to use PWM Double Buffer function to change duty cycle and period of output waveform.
RTC_AlarmWakeup	Demonstrate how to use RTC alarm interrupt event to wake up system.
RTC_SpareRegisterRW	Demonstrate how to access RTC spare registers.
RTC_TimeAndTick	Demonstrate how to get the current RTC data/time per tick.
SC_ReadATR	Demonstrate how to get smart card ATR data.
SCUART_TxRx	Demonstrate how to use smartcard interface UART mode to print "Hello World!"
SPI_Loopback	Demonstrate 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_MasterMode	Demonstrate how to communicate with an off-chip SPI slave device. Needs to work with SPI_SlaveMode 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_SlaveMode	Demonstrate how to communicate with an off-chip SPI master device. This sample code needs to work with SPI_MasterMode.
SYS_BODWakeup	Demonstrate how to wake up system form Power-down mode by brown-out detector interrupt.
SYS_PLLClockOutput	Demonstrate how to change system clock to different PLL frequency and output system clock from CLKO pin.

TIMER_EventCounter	Demonstrate how to use timer1 counter input function to count the input event.
TIMER_CaptureCounter	Demonstrate how to use the timer2 capture event to capture timer2 counter value.
TIMER_Delay	Demonstrate how to use timer0 to create a precise delay loop.
TIMER_PeriodicINT	Demonstrate how to perform timer counting in periodic mode.
TIMER_TimeoutWakeup	Demonstrate how to use timer0 periodic time-out interrupt event to wake up system.
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 transmit and receive data from PC terminal through RS232 interface.
UART_Wakeup	Demonstrate how to wake up system form Power-down mode by UART interrupt.
USBD_Audio_HID_NAU8822	Demonstrate how to implement a USB audio class device with HID key. NAU8822 is used in this sample code to play the audio data from Host. It also supports to record data from NAU8822 to Host.
USBD_Audio_NAU8822	Demonstrate how to implement a USB audio class device. NAU8822 is used in this sample code to play the audio data from Host. It also supports to record data from NAU8822 to Host.
USBD_HID_Keyboard	Demonstrate how to implement a USB keyboard device. This sample code supports to use GPIO to simulate key input.
USBD_HID_Mouse	Demonstrate how to implement a USB mouse device. The mouse cursor will move automatically when this mouse device connecting to PC by USB.
USBD_HID_Transfer	Demonstrate how to transfer data between USB device and PC through USB HID interface. A windows tool is also included in this sample code to connect with USB device.
USBD_MassStorage_DataFlash	Demonstrate how to implement a USB Mass-Storage. It uses embedded data flash as storage.
USBD_VCOM	Demonstrate how to implement a USB virtual com port device. It supports one virtual comport.
USBH_HID	Demonstrate how to implement a USB Host and recognize a HID device when device plug-in.

USBH_UMAS	Demonstrate how to implement a USB Host with a file system to read/write a file on USB Mass Storage.
USBOTG_Dual_Role_UMAS	Demonstrate how USB works as a dual role device. If it works as USB Host, it can access a mass storage device. If it works as USB Device, It acts as a mass storage device.
WDT_TimeoutWakeupAndReset	Demonstrate how to use 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	Select one WWDT window compare value to generate the window compare match interrupt event.

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.