

## Mini55 Series CMSIS BSP Guide

Directory Introduction for 32-bit NuMicro® Family

### Directory Information

<b>Document</b>	Driver reference guide and revision history.
<b>Library</b>	Driver header and source files.
<b>SampleCode</b>	Driver sample code.

*The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.*

*Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.*

*All data and specifications are subject to change without notice.*

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

[www.nuvoton.com](http://www.nuvoton.com)

## TABLE OF CONTENTS

<b>1</b>	<b>DOCUMENT.....</b>	<b>4</b>
<b>2</b>	<b>LIBRARY .....</b>	<b>5</b>
<b>3</b>	<b>SAMPLECODE .....</b>	<b>6</b>
<b>4</b>	<b>SAMPLECODE\ISP .....</b>	<b>7</b>
<b>5</b>	<b>SAMPLECODE\NUTINY-SDK-MINI55.....</b>	<b>8</b>
<b>6</b>	<b>SAMPLECODE\REGBASED .....</b>	<b>9</b>
	Clock Controller (CLK).....	9
	Flash Memory Controller (FMC).....	9
	General Purpose I/O (GPIO).....	9
	Timer Controller (TIMER).....	9
	Watchdog Timer (WDT) .....	10
	PWM Generator (PWM).....	10
	UART Interface Controller (UART).....	10
	Serial Peripheral Interface (SPI).....	10
	I <sup>2</sup> C Serial Interface Controller (I <sup>2</sup> C) .....	10
	Hardware Divider (HDIV).....	11
	Analog-to-Digital Converter (ADC) .....	11
	Analog Comparator Controller (ACMP) .....	11
<b>7</b>	<b>SAMPLECODE\STDDRIVER.....</b>	<b>12</b>
	System Manager (SYS) .....	12
	Clock Controller (CLK).....	12
	Flash Memory Controller (FMC).....	12
	General Purpose I/O (GPIO).....	12
	Timer Controller (TIMER).....	12
	Watchdog Timer (WDT) .....	13
	PWM Generator (PWM).....	13
	UART Interface Controller (UART).....	13

---

Serial Peripheral Interface (SPI) .....	14
I <sup>2</sup> C Serial Interface Controller (I <sup>2</sup> C) .....	14
Hardware Divider (HDIV).....	14
Analog-to-Digital Converter (ADC) .....	14
Analog Comparator Controller (ACMP) .....	14

## 1 Document

<b>CMSIS.html</b>	Document of CMSIS version 4.5.0.
<b>NuMicro Mini55 Driver Reference Guide.html</b>	This document describes the usage of drivers in Mini55 BSP.
<b>NuMicro Mini55 Series CMSIS BSP Revision History.pdf</b>	This document shows the revision history of Mini55 BSP.

## 2 Library

<b>CMSIS</b>	Cortex® Microcontroller Software Interface Standard (CMSIS) V4.5.0 definitions by Arm® Corp.
<b>Device</b>	CMSIS compliant device header file.
<b>StdDriver</b>	All peripheral driver header and source files.

### 3 SampleCode

<b>Hard_Fault_Sample</b>	Show hard fault information when hard fault happened.
<b>ISP</b>	ISP firmware samples.
<b>NuTiny-SDK-MINI55</b>	Sample code for Mini55 Tiny Board
<b>RegBased</b>	Sample codes implemented without access standard library but access registers directly.
<b>Semihost</b>	Show how to print and get character through IDE console window.
<b>StdDriver</b>	Demonstrate the usage of Mini55 MCU peripheral driver APIs.
<b>Template</b>	A project template for Mini55 MCU.

## 4 SampleCode\ISP

ISP_I2C	In-System-Programming sample code through I <sup>2</sup> C 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\NuTiny-SDK-Mini55**

**LED\_Toggle**

This sample toggles P1.5 to turn on board LED on and off.



## 6 SampleCode\RegBased

### Clock Controller (CLK)

CLK_SwitchHCLK	Demonstrate how to switch HCLK between HIRC and HXT.
----------------	--

### Flash Memory Controller (FMC)

FMC_RW	Show FMC read flash IDs, erase, read, and write functions.
--------	--

### General Purpose I/O (GPIO)

GPIO	Use GPIO driver to control the GPIO pin direction, control their high/low state, and how to use GPIO interrupts.
------	--

### Timer Controller (TIMER)

Timer_EventCounter	Use pin P3.4 to demonstrates timer event counter function.
Timer_FreeCountingMode	Use the timer pin P3.2 to demonstrate timer free counting mode function. Also display the measured input frequency to UART console.
Timer_Periodic	Use the timer periodic mode to generate timer interrupt every 1 second.
Timer_ToggleOut	Demonstrate the timer 0 toggle out function on pin P3.4.
Timer_TriggerCountingMode	Use the timer pin P3.2 to demonstrate timer trigger counting mode function. And displays the measured input frequency to UART console.
Timer_Wakeup	Use Timer to wake up system from Power-down mode periodically.

## Watchdog Timer (WDT)

WDT_Polling	Use polling mode to check WDT time-out state and reset WDT after time out occurs.
WDT_Wakeup	Use WDT to wake up system from Power-down mode periodically.

## PWM Generator (PWM)

PWM_DeadZone	Demonstrate the dead-zone feature with PWM.
PWM_DoubleBuffer	Demonstrate the PWM double buffer feature.

## UART Interface Controller (UART)

UART_AutoFlow	Show how to transmit and receive data using auto flow control.
UART_IrDA	Show how to transmit and receive UART data in UART IrDA mode.
UART_RS485	Transmit and receive data in UART RS485 mode.
UART_TxRx_Function	Transmit and receive data from PC terminal through RS232 interface.

## Serial Peripheral Interface (SPI)

SPI_FIFO_FLASH	Access SPI Flash using FIFO mode.
SPI_LoopBack	Demonstrate SPI function by connect MOSI (P0.5) with MISO (P0.6).

## I<sup>2</sup>C Serial Interface Controller (I<sup>2</sup>C)

I2C_FIFO_EEPROM	Read/write EEPROM via I <sup>2</sup> C interface using FIFO mode.
I2C_Interrupt_EEPROM	Read/write EEPROM via I <sup>2</sup> C interface using interrupt mode.

## Hardware Divider (HDIV)

HDIV	Show how to use hardware divider.
------	-----------------------------------

## Analog-to-Digital Converter (ADC)

ADC_Compare	Demonstrate ADC conversion and comparison function by monitoring the conversion result of channel 0.
ADC_Convert	Demonstrate ADC function by repeatedly convert the input of ADC channel 0 (P5.3) and shows the result on UART console.

## Analog Comparator Controller (ACMP)

ACMP	Demonstrate Analog comparator (ACMP) comparison by comparing CPP0 (P1.5) with Band-gap voltage and shows the result on UART console.
ACMP_TriggerTimerCapture	Show how to use Analog comparator (ACMP) state change to trigger timer capture function. P1.5 is used as comparator positive input and Band-gap voltage as negative input.

## 7 SampleCode\StdDriver

### System Manager (SYS)

<b>SYS_Control</b>	Demonstrate how to get PDID, get and clear reset source, configure BOD, and output system clock to CKO pin with the system clock / 4 frequency.
<b>SYS_TrimRC</b>	Demonstrate how to use LXT to trim HIRC.

### Clock Controller (CLK)

<b>CLK_SwitchHCLK</b>	Demonstrate how to switch HCLK between HIRC and HXT.
-----------------------	--

### Flash Memory Controller (FMC)

<b>FMC_IAP</b>	<p>This sample code includes LDROM image (fmc_ld_iap) and APROM image (fmc_ap_main).</p> <p>It shows how to branch between APROM and LDROM. To run this sample code, the boot mode must be "Boot from APROM with IAP".</p>
<b>FMC_RW</b>	Show FMC read flash IDs, erase, read, and write functions.

### General Purpose I/O (GPIO)

<b>GPIO</b>	Use GPIO driver to control the GPIO pin direction, control their high/low state, and how to use GPIO interrupts.
-------------	--

### Timer Controller (TIMER)

<b>Timer_Delay</b>	Demonstrate the usage of TIMER_Delay() API to generate a 1 second delay
<b>Timer_EventCounter</b>	Use pin P3.4 to demonstrates timer event counter function.

<b>Timer_FreeCountingMode</b>	Use the timer pin P3.2 to demonstrate timer free counting mode function. Also display the measured input frequency to UART console.
<b>Timer_Periodic</b>	Use the timer periodic mode to generate timer interrupt every 1 second.
<b>Timer_ToggleOut</b>	Demonstrate the timer 0 toggle out function on pin P3.4.
<b>Timer_TriggerCountingMode</b>	Use the timer pin P3.2 to demonstrate timer trigger counting mode function. And displays the measured input frequency to UART console.
<b>Timer_Wakeup</b>	Use Timer to wake up system from Power-down mode periodically.

### Watchdog Timer (WDT)

<b>WDT_Polling</b>	Use polling mode to check WDT time-out state and reset WDT after time out occurs.
<b>WDT_Wakeup</b>	Use WDT to wake up system from Power-down mode periodically.

### PWM Generator (PWM)

<b>PWM_DeadZone</b>	Demonstrate the dead-zone feature with PWM.
<b>PWM_DoubleBuffer</b>	Demonstrate the PWM double buffer feature.

### UART Interface Controller (UART)

<b>UART_AutoFlow</b>	Show how to transmit and receive data using auto flow control.
<b>UART_IrDA</b>	Show how to transmit and receive UART data in UART IrDA mode.
<b>UART_RS485</b>	Transmit and receive data in UART RS485 mode.
<b>UART_TxRx_Function</b>	Transmit and receive data from PC terminal through

	RS232 interface.
--	------------------

## Serial Peripheral Interface (SPI)

SPI_FIFO_FLASH	Access SPI Flash using FIFO mode.
SPI_LoopBack	Demonstrate SPI function by connect MOSI (P0.5) with MISO (P0.6).

## I<sup>2</sup>C Serial Interface Controller (I<sup>2</sup>C)

I2C_FIFO_EEPROM	Read/write EEPROM via I <sup>2</sup> C interface using FIFO mode.
I2C_Interrupt_EEPROM	Read/write EEPROM via I <sup>2</sup> C interface using interrupt mode.

## Hardware Divider (HDIV)

HDIV	Show how to use hardware divider.
------	-----------------------------------

## Analog-to-Digital Converter (ADC)

ADC_Compare	Demonstrate ADC conversion and comparison function by monitoring the conversion result of channel 0.
ADC_Convert	Demonstrate ADC function by repeatedly convert the input of ADC channel 0 (P5.3) and shows the result on UART console.

## Analog Comparator Controller (ACMP)

ACMP	Demonstrate Analog comparator (ACMP) comparison by comparing CPP0 (P1.5) with Band-gap voltage and shows the result on UART console.
------	--

### **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.*