SPI Use PDMA to Transfer

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	SPI use PDMA to transfer data
BSP Version	NUC123Series_BSP_CMSIS_v3.01.001
Hardware	NuTiny-EVB-123-LQFP64 Ver1.0

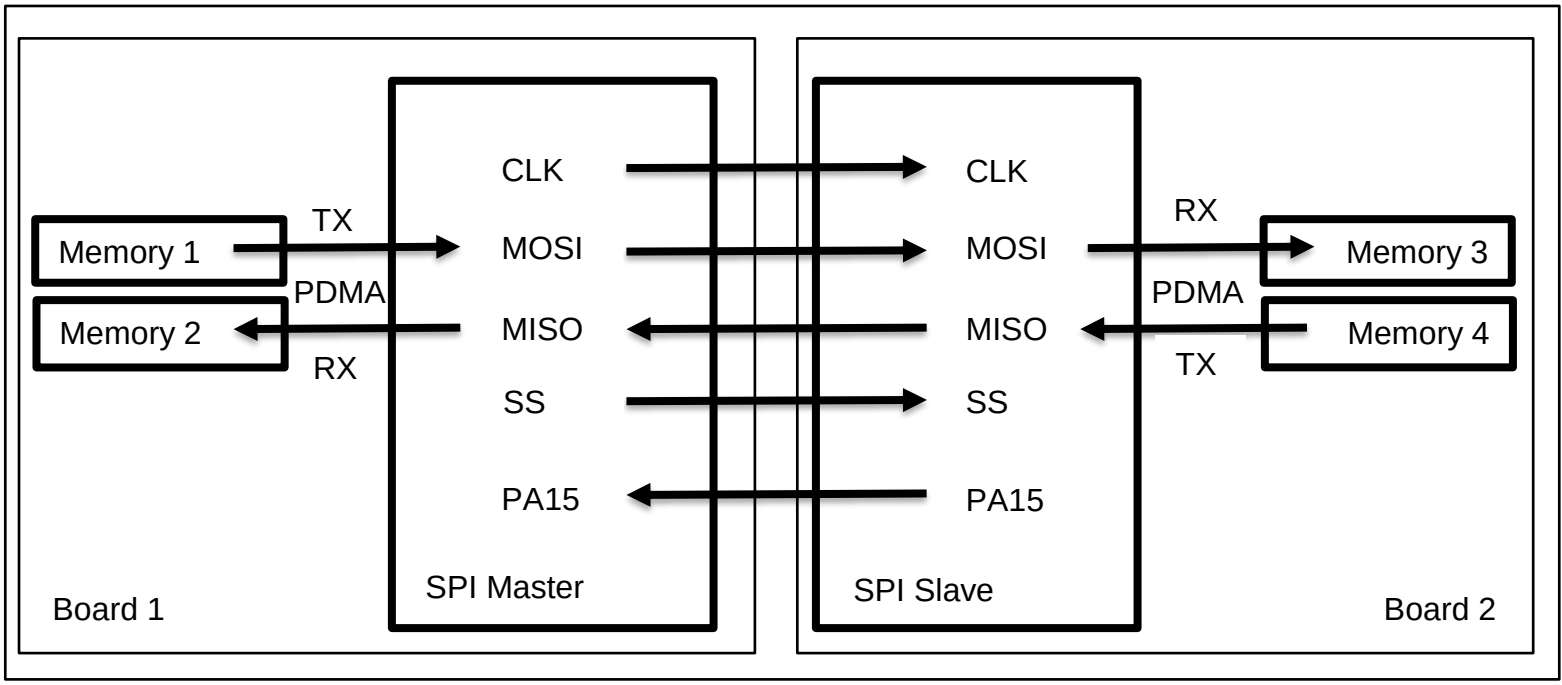
1 Function Description

1.1 Introduction

Two boards are used. One board uses PDMA for SPI Master and another board uses PDMA for SPI Slave. The two boards transmit data to each other.

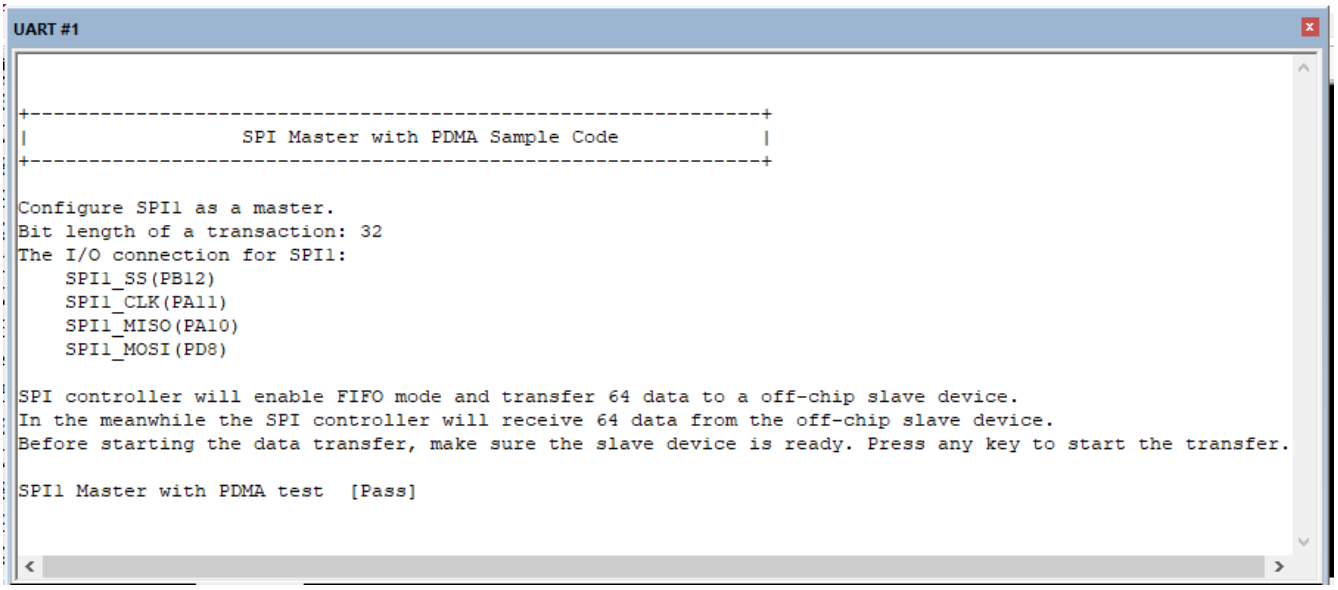
1.2 Principle

One boards as SPI1 Master to use PDMA Basic mode to transfer data to TX and use PDMA Basic mode to transfer RX data to memory, and another board as SPI1 Slave also use PDMA. Using PA15 as the synchronization signal between the two boards, the program will compare the correctness of the data and output the transmission result.

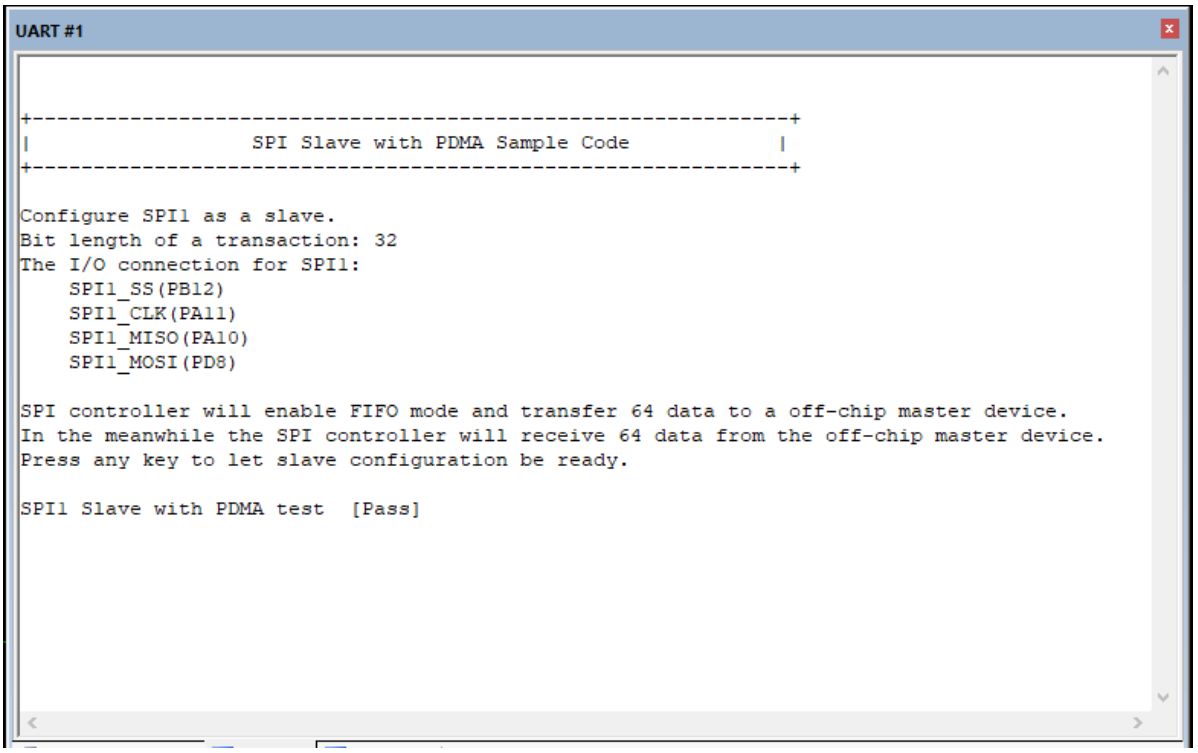


1.3 Demo Result

The transmission result of SPI1 Master as follows.



The transmission result of SPI1 slave as follows.



2 Code Description

SPI1 Master uses PDMA Basic mode to transfer：

```
    /* Set SPI TX with PDMA configuration */
    PDMA_Open(1 << DMA_SPI1_TX);
    PDMA_SetTransferCnt(DMA_SPI1_TX, PDMA_WIDTH_8, PDMA_TEST_LENGTH);
    PDMA_SetTransferAddr(DMA_SPI1_TX, (uint32_t)u8SrcArray, PDMA_SAR_INC,
    (uint32_t)&SPI1->TX[0], PDMA_DAR_FIX);
    PDMA_EnableInt(DMA_SPI1_TX, PDMA_IER_BLKD_IE_Msk);
    PDMA2->CSR = (PDMA2->CSR & ~(PDMA_CSR_MODE_SEL_Msk) | (0x2 << PDMA_CSR_MODE_SEL_Pos));

    /* Set SPI RX with PDMA configuration */
    PDMA_Open(1 << DMA_SPI1_RX);
    PDMA_SetTransferCnt(DMA_SPI1_RX, PDMA_WIDTH_8, PDMA_TEST_LENGTH);
    PDMA_SetTransferAddr(DMA_SPI1_RX, (uint32_t)&SPI1->RX[0], PDMA_SAR_FIX,
    (uint32_t)u8DestArray, PDMA_DAR_INC);
    PDMA_EnableInt(DMA_SPI1_RX, PDMA_IER_BLKD_IE_Msk);
    PDMA3->CSR = (PDMA3->CSR & ~(PDMA_CSR_MODE_SEL_Msk) | (0x1 << PDMA_CSR_MODE_SEL_Pos));
    PDMA_GCR->PDSSR0 = 0x0FF2301;

    NVIC_EnableIRQ(PDMA_IRQn);

    /* Enable PDMA trigger function */
    PDMA_Trigger(DMA_SPI1_RX);
    PDMA_Trigger(DMA_SPI1_TX);

    /* Wait for slave */
    while (PA15 == 0);

    /* Enable SPI master DMA function */
    SPI_TRIGGER_RX_PDMA(SPI1);
    SPI_TRIGGER_TX_PDMA(SPI1);
```

SPI1 Slave uses PDMA Basic mode to transfer：

```
    /* Set SPI TX with PDMA configuration */
    PDMA_Open(1 << DMA_SPI1_TX);
    PDMA_SetTransferCnt(DMA_SPI1_TX, PDMA_WIDTH_8, PDMA_TEST_LENGTH);
    PDMA_SetTransferAddr(DMA_SPI1_TX, (uint32_t)SrcArray, PDMA_SAR_INC,
    (uint32_t)&SPI1->TX[0], PDMA_DAR_FIX);
    PDMA_EnableInt(DMA_SPI1_TX, PDMA_IER_BLKD_IE_Msk);
    PDMA2->CSR = (PDMA2->CSR & ~(PDMA_CSR_MODE_SEL_Msk) | (0x2 << PDMA_CSR_MODE_SEL_Pos));

    /* Set SPI RX with PDMA configuration */
    PDMA_Open(1 << DMA_SPI1_RX);
    PDMA_SetTransferCnt(DMA_SPI1_RX, PDMA_WIDTH_8, PDMA_TEST_LENGTH);
    PDMA_SetTransferAddr(DMA_SPI1_RX, (uint32_t)&SPI1->RX[0], PDMA_SAR_FIX,
    (uint32_t)DestArray, PDMA_DAR_INC);
    PDMA_EnableInt(DMA_SPI1_RX, PDMA_IER_BLKD_IE_Msk);
    PDMA3->CSR = (PDMA3->CSR & ~(PDMA_CSR_MODE_SEL_Msk) | (0x1 << PDMA_CSR_MODE_SEL_Pos));
    PDMA_GCR->PDSSR0 = 0x0FF2301;

    NVIC_EnableIRQ(PDMA_IRQn);

    /* Enable PDMA trigger function */
    PDMA_Trigger(DMA_SPI1_RX);
    PDMA_Trigger(DMA_SPI1_TX);
```

```
    /* Enable SPI slave PDMA function */
    SPI_TRIGGER_RX_PDMA(SPI1);
    SPI_TRIGGER_TX_PDMA(SPI1);

    /* Sync with Master */
    PA15 = 1;
```

3 Software and Hardware Environment

Software Environment

- BSP version
  - NUC123Series_BSP_CMSIS_v3.01.001
- IDE version
  - Keil uVersion 5.26

Hardware Environment

- Circuit components
  - NuTiny-EVB-123-LQFP64 Ver1.0

4 Directory Information

- EC_NUC123_SPI_WITH_PDMA_V1.00
  - Library — Sample code header and source files
    - CMSIS — Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
    - Device — CMSIS compliant device header file
    - SmartcardLib — Smartcard library binary and header file
    - StdDriver — All peripheral driver header and source files
    - UsbHostLib — USB host library source code
  - SampleCode
    - ExampleCode — Source file of example code

5 How to Execute Example Code

1. Browsing into sample code folder by Directory Information (section 4) and double click NUC123_SPI_WITH_PDMA_Master.uvproj and

   NUC123_SPI_WITH_PDMA_Slave.uvproj。

2. Enter Keil compile mode

   a. Build

   b. Download

   c. Start/Stop debug session

3. Enter debug mode

   a. Run

Revision History

Date	Revision	Description
Sep. 20, 2019	1.00	1. Initially issued.

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.