

1 概述

这篇应用笔记阐述了如何使用 I.MXRT 系列芯片上的 FlexIO 模块来模拟 SPI 总线的主机与从机应用。

FlexIO 是 I.MXRT 系列 MCU 上板载的一种外设。它是一个非常灵活并且可以随意配置的模块，它可以模拟出类似于 UART, I2C, I2S 等常用通信接口，当然，它也可以成功的模拟 SPI 总线。

这篇应用笔记基于 I.MXRT1010 平台创建了一个简单的例程来演示如何使用 FlexIO 模块去模拟出 SPI 总线的主机与从机。

2 FlexIO 概述

I.MXRT1010 的 FlexIO 模块具有以下的一些特性：

- 多组 32 位的移位寄存器，具有发送、接收以及数据拟合等模式。
- 具有双重缓存区的移位寄存器支持连续的数据传输。
- 自动生成并插入起始位及停止位。
- 中断，直接存储器存取或轮询的方式进行收发。
- 独立于总线时钟频率的可编程波特率，支持停止模式下的异步操作。
- 高度灵活的 16 位定时器，支持各种内部或外部触发器、复位、使能与失能方式。
- 可编程的逻辑模式，通过集成外部数字逻辑功能芯片或结合引脚/移位器/定时器功能，以产生复杂的输出。
- 支持当 CPU 进入失能各项系统控制功能模式下的可编程状态机，支持最多 8 个状态，8 个输出和每个状态 3 个可选择的输入。

图 1 为整个 FlexIO 模块的移位器定时器及引脚的结构图。

目录

1 概述.....	1
2 FlexIO 概述.....	1
3 利用 FlexIO 模拟 SPI.....	2
3.1 SPI 主机配置.....	2
3.2 SPI 从机配置.....	6
4 示例的运行.....	8
4.1 示例平台.....	8
4.2 运行示例.....	9
5 参考资料.....	10



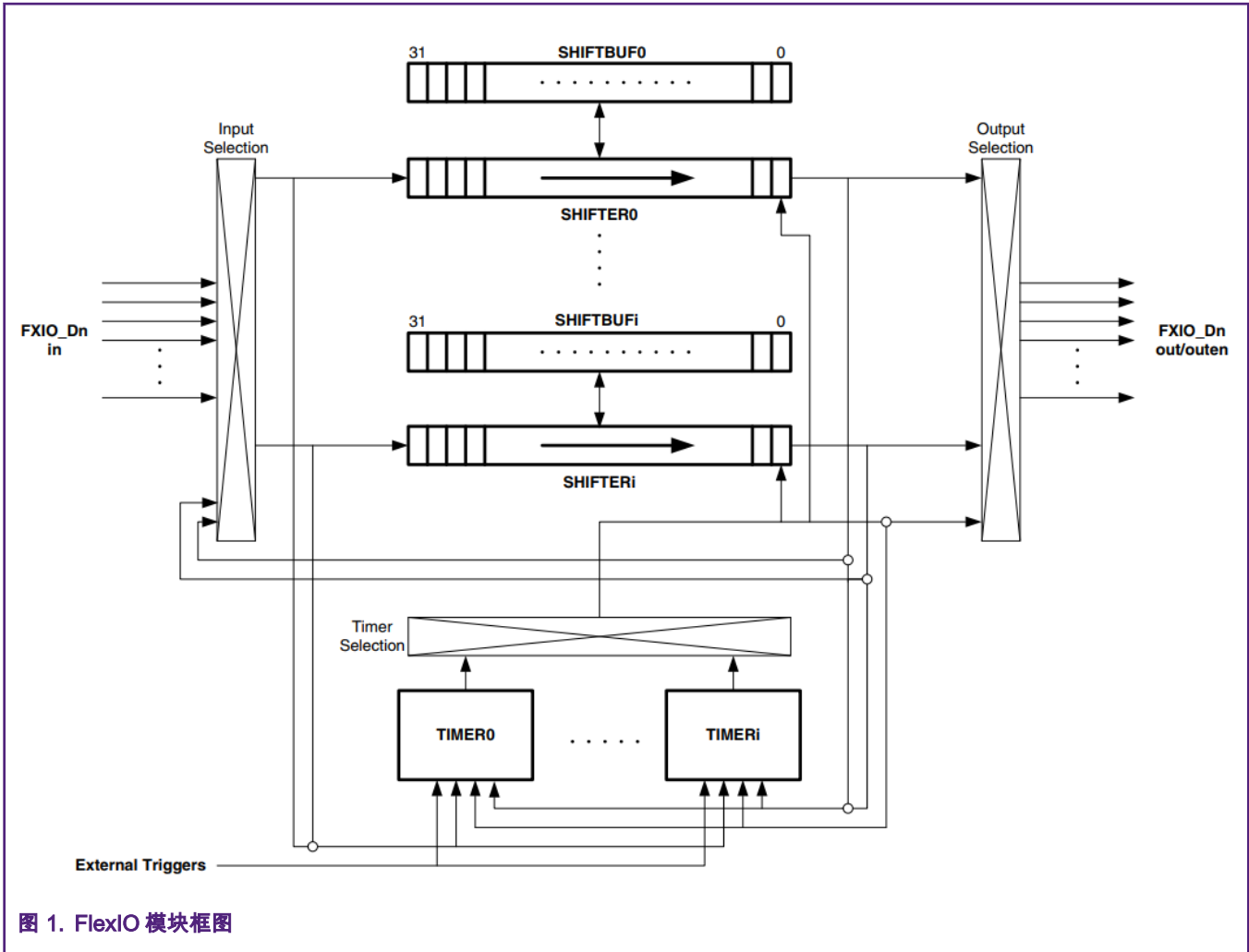


图 1. FlexIO 模块框图

I.MXRT 系列 MCU 的 FlexIO 模块所包含的资源不尽相同，用户可根据 FLEXIO_PARAM 寄存器读取所用芯片 FlexIO 模块包含的移位器，定时器，引脚及触发器数量，本例程中所使用的 I.MXRT1010 芯片中包含 8 个寄存器，8 个定时器，32 个引脚以及两个外部触发器（由于芯片设计引脚的限制，RT1010 的实际使用的 FlexIO 引脚只有 27 个）。

3 利用 FlexIO 模拟 SPI

本章主要介绍如何使用 FlexIO 模拟 SPI 总线的主机及从机，将会详细阐述 FlexIO 模块的相关配置项。

3.1 SPI 主机配置

模拟 SPI 主机时，需要用到以下的一些 FlexIO 模块资源：

- 两个定时器 – 一个用于产生 SPI_CS 输出信号，另一个用于产生 SPI_SCK 信号输出并且控制所使用的两个移位器的数据的读取/写入/移位操作。
- 两个移位器 – 分别用作数据的发送与接收。
- 四个引脚 – 分别接到上面使用的定时器与移位器用作 SPI_CS，SPI_SCK, SPI_MOSI 和 SPI_MISO 引脚。

图 2 为 FlexIO SPI 主机的结构框图。

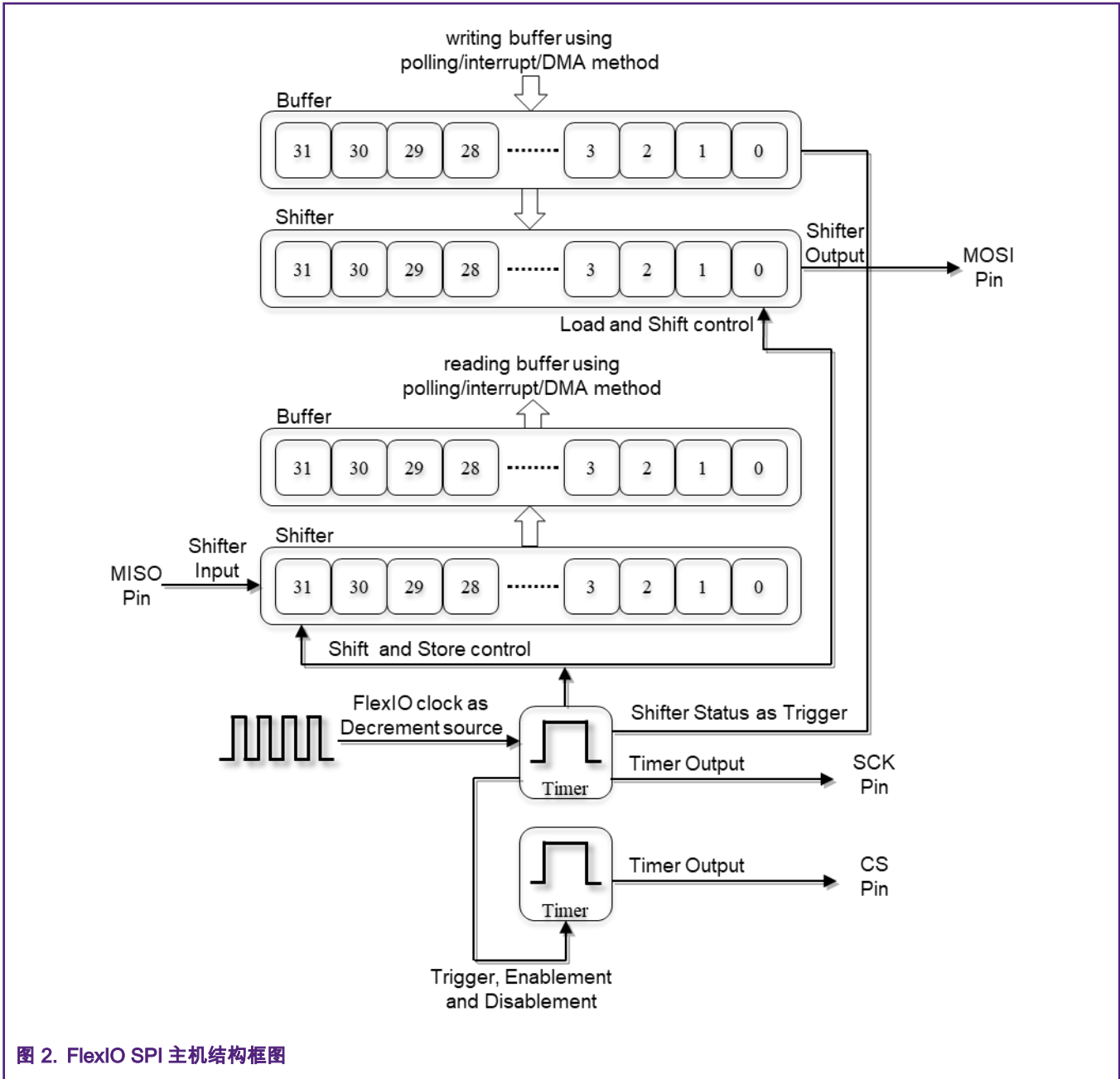


图 2. FlexIO SPI 主机结构框图

在本应用笔记中，Timer 0 被用作生成 SPI_SCK 信号，并通过 FlexIO_D26 引脚输出；Timer 1 被用作生成 SPI_CS 信号，并通过 FlexIO_D0 输出；Shifter 0 连接到 FlexIO_D21 在 SPI_SCK 信号的每个上升沿发出数据；Shifter 1 连接到 FlexIO_D22 在 SPI_SCK 的每个下降沿以接收数据。表 1 所示为主机 Timer 0 的具体配置。

表 1. 主机 Timer 0 配置表

Items	Configurations
Trigger Select	Shifter 0 status flag
Trigger Polarity	active low
Trigger Source	internal trigger

Table continues on the next page...

表 1. 主机 Timer 0 配置表 (continued)

Items	Configurations
Pin Config	Output
Pin Select	FlexIO_D26
Pin Polarity	active high
Timer mode	dual 8-bit counters baud mode
Timer Output	Timer output is logic zero when enabled and is not affected by timer reset
Timer Decrement	decrement counter on FlexIO clock, shift clock equals Timer output
Timer Reset	Timer never reset
Timer Disable	Timer disabled on Timer compare
Timer Enable	Timer enabled on Trigger high
Timer Stop Bit	enabled on timer disable
Timer Start Bit	Enable
Timer Compare	$((\text{bitCountPerChar}^1 * 2 - 1) \ll 8) (\text{baudrate_divider}^2 / 2 - 1)$

1. bitCountPerChar 是每个字的数据所包含的位数。
2. baudrate_divider 可以由 FlexIO clock 除以 SPI 总线的波特率得到。

Timer 0 被配置为 8-bit baud counter 模式，这种模式下，16 位长度的计数器以及比较寄存器（计数器记到 0 时会重新加载比较寄存器内的值）被分割成两个 8 位长度的计数器。其中，低 8 位的计数器用来配置波特率（移位器时钟），高 8 位的计数器用来配置传输中的字长。当低 8 位的计数器自减到 0 时，Timer 的输出会翻转，然后低 8 位的计数器会重载比较寄存器中的值，此时高 8 位的计数器会自减 1。

表 2 所示为主机 Timer 1 的配置。

表 2. 主机 Timer 1 配置表

Items	Configurations
Trigger Select	trigger from Timer 0
Trigger Polarity	active high
Trigger Source	internal trigger
Pin Config	Output
Pin Select	FlexIO_D0
Pin Polarity	active low
Timer mode	single 16-bit counter mode
Timer Output	Timer output is logic one when enabled and is not affected by timer reset
Timer Decrement	decrement counter on FlexIO clock, shift clock equals Timer output
Timer Reset	Timer never reset
Timer Disable	Timer disabled on Timer 0 disable
Timer Enable	Timer enabled on Timer 0 enable

Table continues on the next page...

表 2. 主机 Timer 1 配置表 (continued)

Items	Configurations
Timer Stop Bit	Disable
Timer Start Bit	Disable
Timer Compare	0xFFFF

Timer 1 被配置为随着 Timer 0 的使能而使能。Timer 1 被配置为 16-bit counter 模式，而这种情况下的计数器及比较寄存器会设置为 0xFFFF，在这种配置下，计数器从不进行比较，并且当 Timer 1 使能时一直会输出高电平。

表 3 所示为主机 Shifter 0 的配置表。

表 3. 主机 Shifter 0 配置表

Items	Configurations
Timer Select	Timer 0
Timer Polarity	shift on negedge of shift clock
Pin Config	Shifter pin output
Pin Select	FlexIO_D21
Pin Polarity	active high
Shifter Mode	transmit mode
Input Source	input from pin
Shifter Stop Bit	Disable
Shifter Start Bit	disable, transmitter loads data on enable

表 4 所示为主机 Shifter 1 的配置表。

表 4. 主机 Shifter 1 配置表

Items	Configurations
Timer Select	Timer 0
Timer Polarity	shift on posedge of shift clock
Pin Config	output disable
Pin Select	FlexIO_D22
Pin Polarity	active high
Shifter Mode	receive mode
Input Source	input from pin
Shifter Stop Bit	disable
Shifter Start Bit	disable, transmitter loads data on enable

Shifter 0 与 Shifter 1 分别作为 SPI 主机中的移位输出与输入，它们同时受到 Timer 0 即 SPI_SCK 信号的控制，根据 SPI_SCK 的输出进行移位。

3.2 SPI 从机配置

模拟 SPI 从机时，需要用到以下的一些 FlexIO 模块资源：

- 一个定时器 – 用于控制所使用的两个移位器的数据的读取/写入/移位操作。
- 两个移位器 – 分别用作数据的发送与接收。
- 四个引脚 – 用作连接 SPI_CS，SPI_SCK, SPI_MOSI 和 SPI_MISO 各信号的引脚。

图 3 所示为 FlexIO SPI 从机的结构框图。

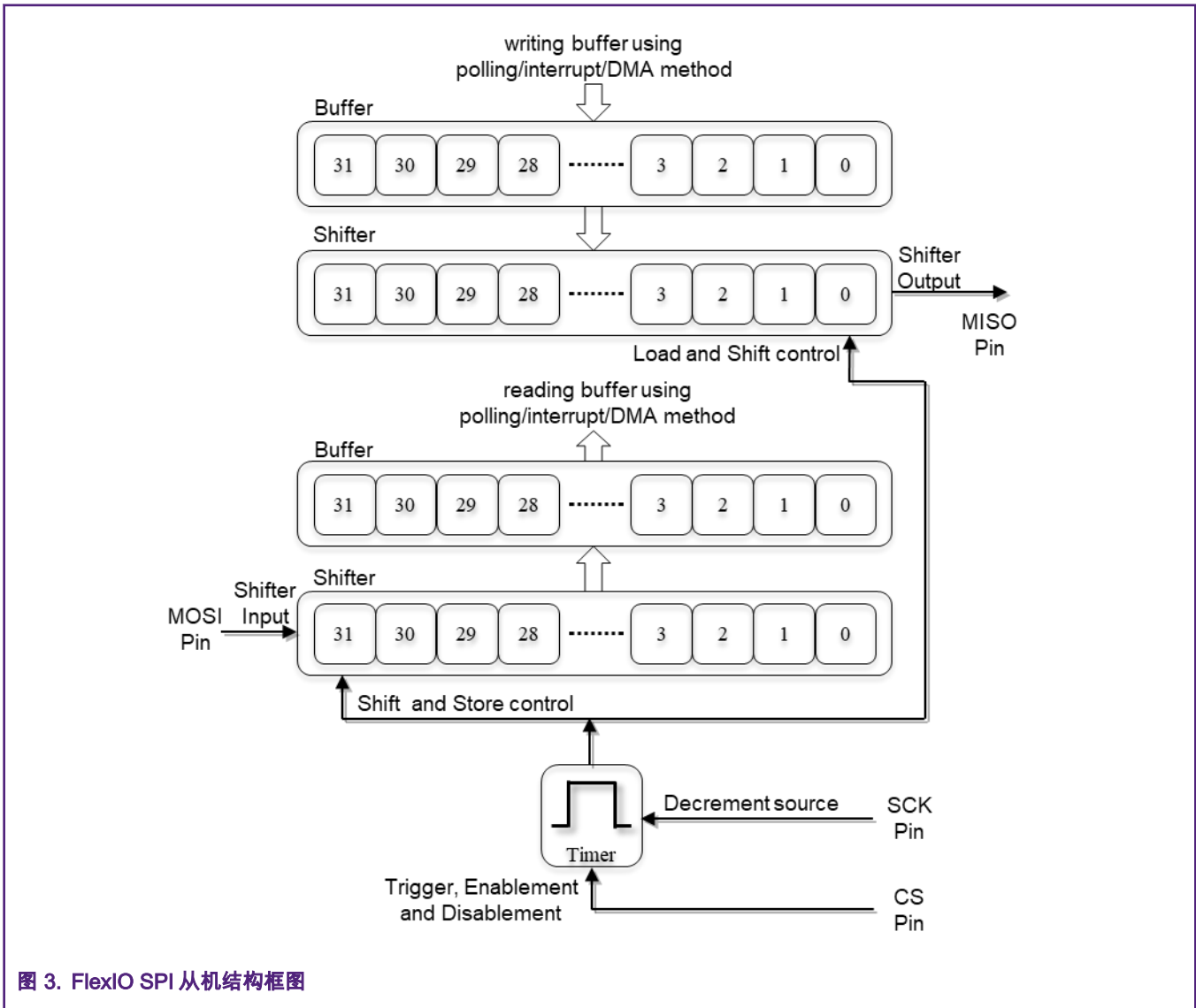


图 3. FlexIO SPI 从机结构框图

在从机模式下，Timer 0 连接到 FlexIO_D26 引脚被用作接收从 SPI 主机发出的 SPI_SCK 的时钟信号并且依据此信号来进行计数并且控制两个移位器数据的移位。当 SPI_CS 信号已经使能时，移位器内的数据将会在 SPI_SCK 信号的每个边沿进行移位。FlexIO_D0 作为 SPI_CS 信号与 SPI 主机的 SPI_CS 输出相连以作为 Timer 0 的触发源。Shifter 0 连接到 FlexIO_D21 作为从机的发射端，Shifter 1 连接到 FlexIO_D22 引脚作为从机的接收端。

表 5 所示为从机 Timer 0 的配置。

表 5. 从机 Timer 0 配置表

Items	Configurations
Trigger Select	trigger from FlexIO_D0
Trigger Polarity	active low
Trigger Source	internal trigger
Pin Config	output disable
Pin Select	FlexIO_D26
Pin Polarity	active high
Timer mode	single 16-bit counter mode
Timer Output	Timer output is logic zero when enabled and is not affected by timer reset
Timer Decrement	decrement counter on pin input, shift clock equals pin input
Timer Reset	Timer never reset
Timer Disable	Timer disabled on Timer compare
Timer Enable	Timer enabled on trigger rising edge
Timer Stop Bit	disable
Timer Start Bit	disable
Timer Compare	$((\text{bitCountPerChar} * 2 - 1))$

表 6 所示为从机 Shifter 0 的配置。

表 6. 从机 Shifter 0 配置表

Items	Configurations
Timer Select	Timer 0
Timer Polarity	shift on negedge of shift clock
Pin Config	Shifter pin output
Pin Select	FlexIO_D21
Pin Polarity	active high
Shifter Mode	transmit mode
Input Source	input from pin
Shifter Stop Bit	disable
Shifter Start Bit	disable, transmitter loads data on enable

表 7 所示为从机 Shifter 1 的配置项。

表 7. 从机 Shifter 1 配置表

Items	Configurations
Timer Select	Timer 0

Table continues on the next page...

表 7. 从机 Shifter 1 配置表 (continued)

Items	Configurations
Timer Polarity	shift on posedge of shift clock
Pin Config	output disable
Pin Select	FlexIO_D22
Pin Polarity	active high
Shifter Mode	receive mode
Input Source	input from pin
Shifter Stop Bit	disable
Shifter Start Bit	disable, transmitter loads data on enable

4 示例的运行

4.1 示例平台

本应用笔记介绍了基于 I.MXRT1010-EVK 开发板的 FlexIO 模拟 LIN 总线的例程。I.MXRT1010-EVK 板如 图 4 所示。用户也可使用 I.MXRT 系列 MCU 的其他 EVK 板经过少许更改后实现这个例程。

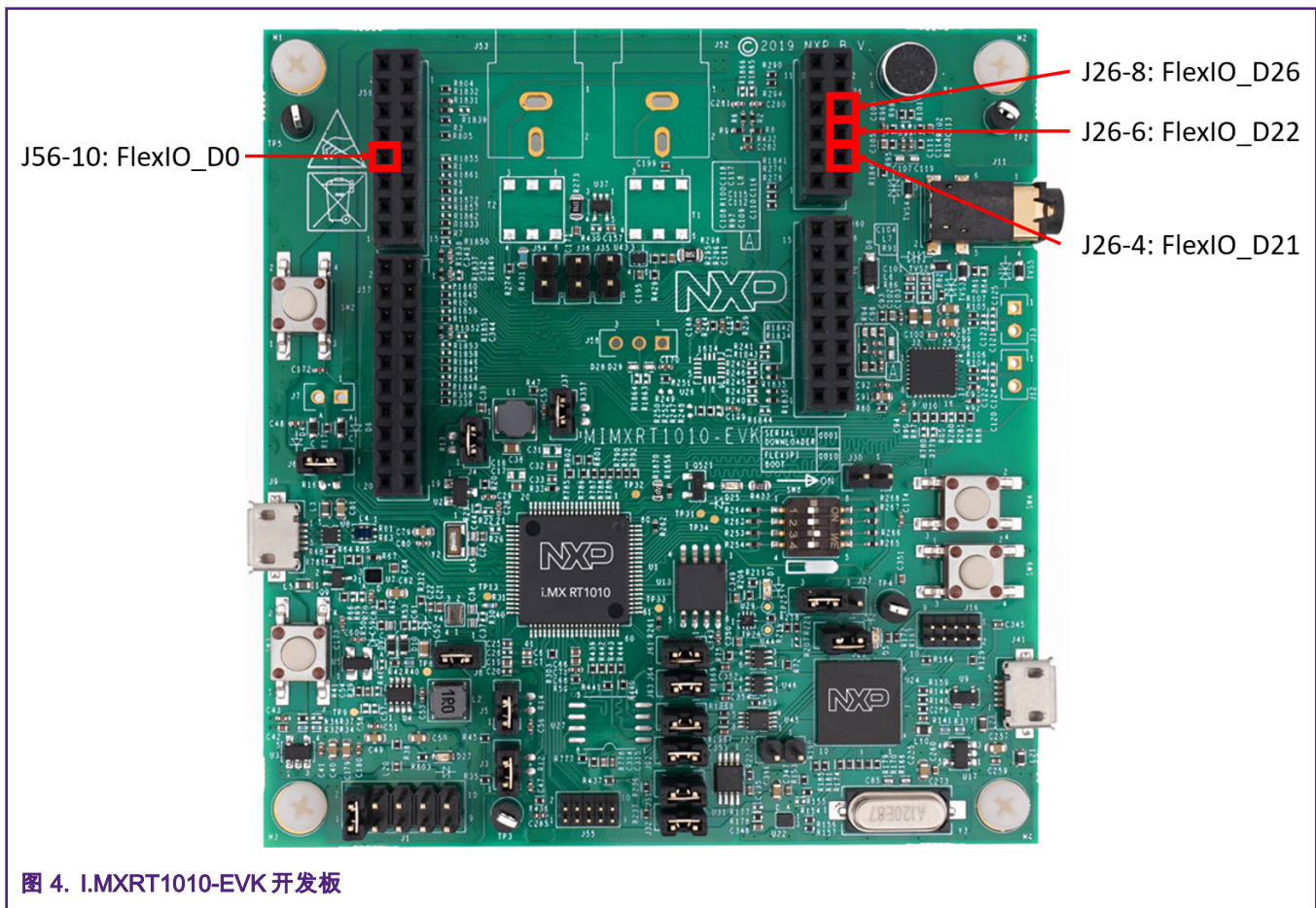


图 4. I.MXRT1010-EVK 开发板

在本例程中，FlexIO 使用 FlexIO_D21 引脚作为数据的输出引脚，使用 FlexIO_D22 作为数据的输入引脚，使用 FlexIO_D26 引脚作为 SPI_SCK 信号引脚，FlexIO_D0 作为 SPI_CS 信号引脚。

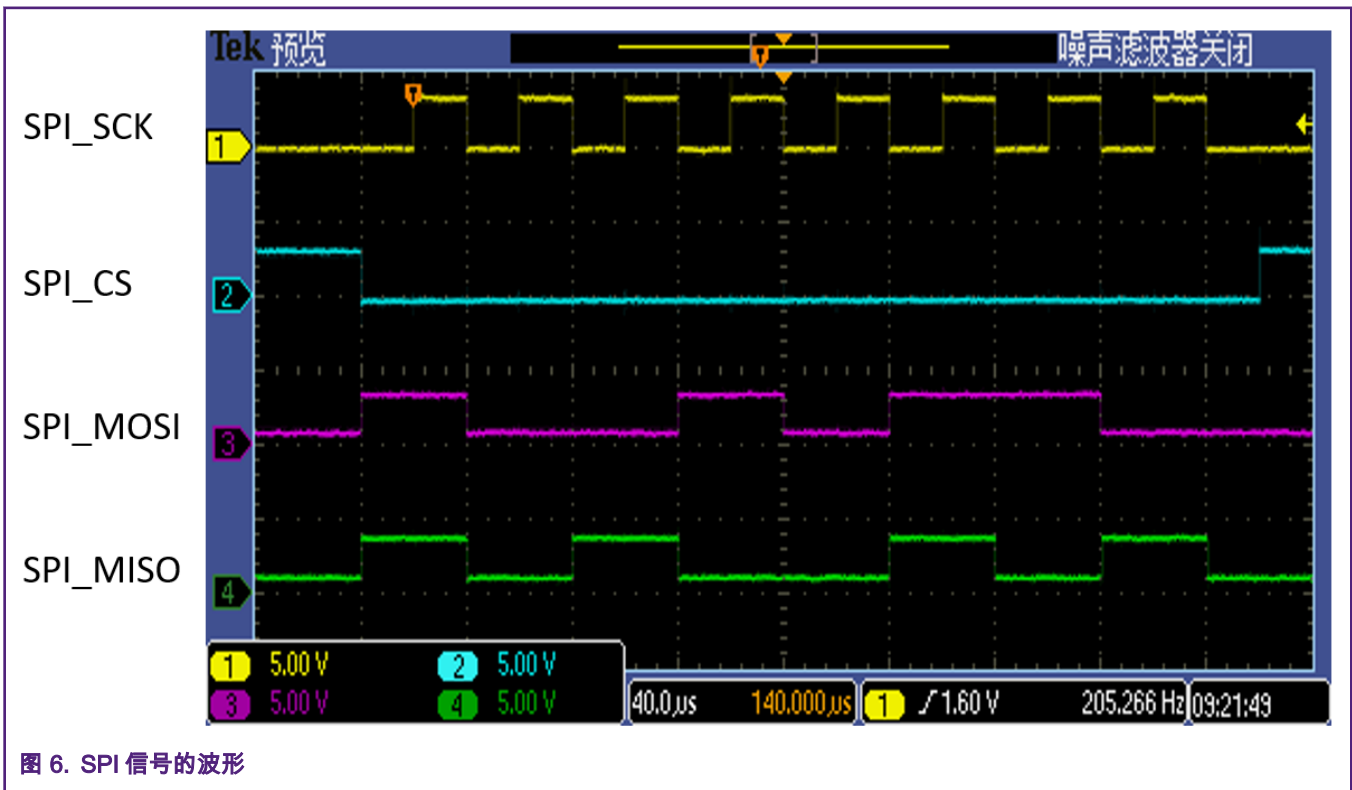
本例程需要使用到两块 EVK 板分别作为 SPI 的主机与从机，两块板子的连线如 图 5 所示。

Pin Name	Master Board		Pin Name	Slave Board
SPI_SCK	J26-8	←---→	SPI_SCK	J26-8
SPI_CS	J56-10	←---→	SPI_CS	J56-10
SPI_MOSI	J26-21	←---→	SPI_MOSI	J26-22
SPI_MISO	J26-22	←---→	SPI_MISO	J26-21
GND	J60-14	←---→	GND	J60-14

图 5. Board connection

4.2 运行示例

用户可从 NXP 官网下载附带的示例程序。找到名为 *flexio_spi_master* 和 *flexio_spi_slave* 的两个 IAR 工程。将两工程分别下载到两块 EVK 开发板上，按照上述的连接接好主机与从机，运行两块板上的示例程序即可。图 6 所示为示波器捕获的 SPI 的 4 根线上的波形。



本示例中 SPI 主机将会发送 0x96 到 SPI 从机，与此同时将会收到从机发送的 0xA5。图 7 所示为 SPI 主机与从机之间通信的数据。

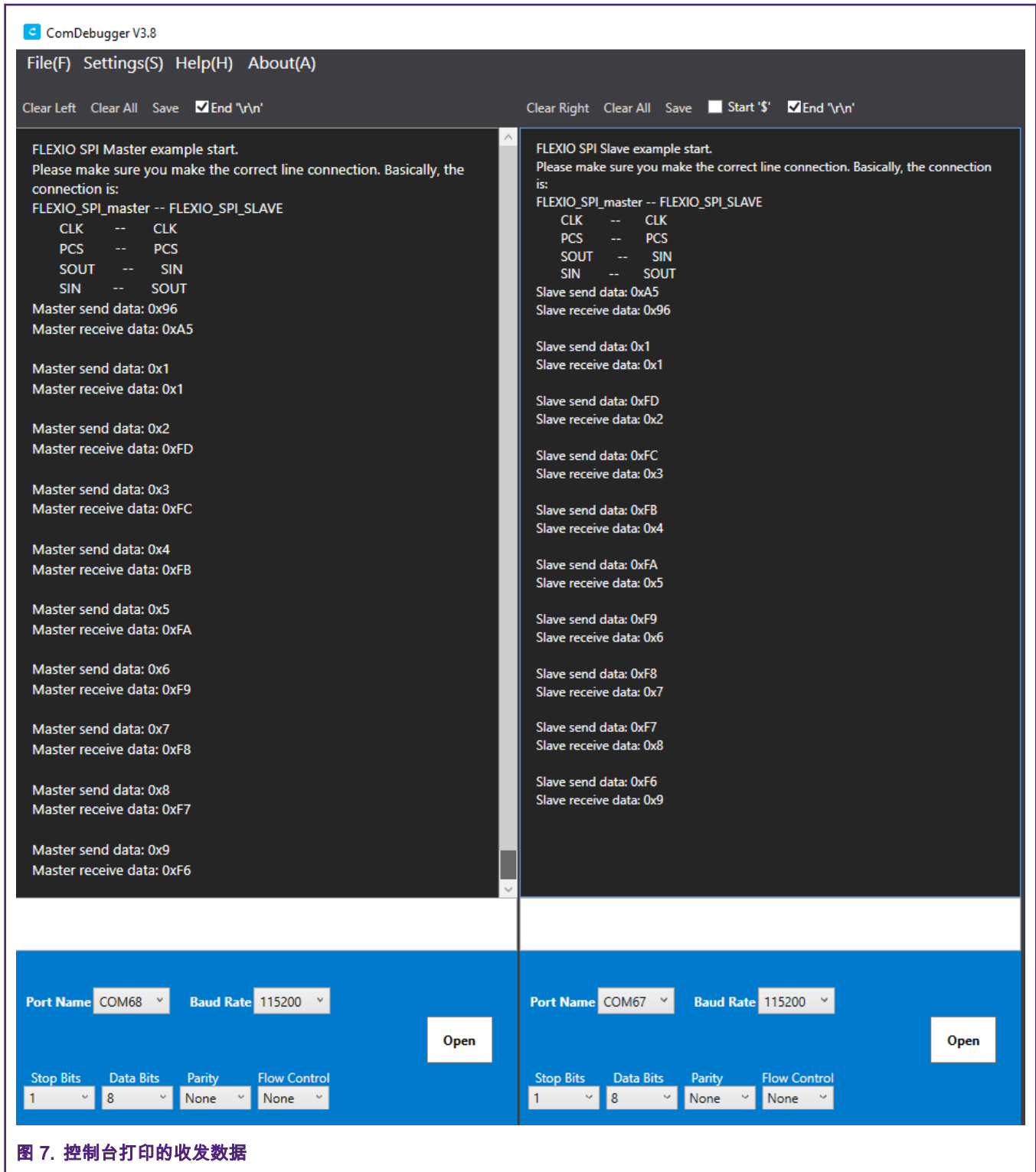


图 7. 控制台打印的收发数据

5 参考资料

1. *IMX RT1010 Processor Reference Manual* (document [I.MXRT1010RM](#))
2. MCUXpresso SDK: Software Development Kit for NXP MCUs <https://mcuxpresso.nxp.com/en/welcome>

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer's applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, I2C BUS, ICODE, JCOP, LIFE VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, Altivec, C-5, CodeTEST, CodeWarrior, ColdFire, ColdFire+, C-Ware, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, Ready Play, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, SMARTMOS, Tower, TurboLink, UMEMS, EdgeScale, EdgeLock, eIQ, and Immersive3D are trademarks of NXP B.V. All other product or service names are the property of their respective owners. AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© NXP B.V. 2020.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: April 2, 2020

Document identifier: AN12780

