

1 介绍

这篇应用笔记阐述了如何使用 i.MX RT1010 单片机的 FlexIO 模块来模拟 SSI 总线接口。SSI 是一种广泛应用于传感器和控制器之间的串行总线接口。当前 i.MX RT1010 不直接支持 SSI 接口，但是一个很好的解决方案是使用其 FlexIO 来模拟 SSI，以实现与专用 SSI 设备之间的通讯。

i.MX RT1010 处理器基于 ARM Cortex-M7 平台，具有很高的 CPU 性能和最佳的实时响应，且拥有丰富的外设资源。本篇应用笔记通过一个简单的例程来演示 FlexIO 模块模拟 SSI 总线接口的过程，并得到了验证。

2 SSI 简介

SSI 全称 Synchronous Serial Interface，即同步串行接口。在本篇应用笔记中，i.MX RT1010 的 FlexIO 外设要模拟的 SSI 接口是一种能够与 Texas Instruments（德州仪器，TI）同步串行接口的器件进行同步串行通信的主机或从机接口，它的单次传输通信格式如图 1 所示。

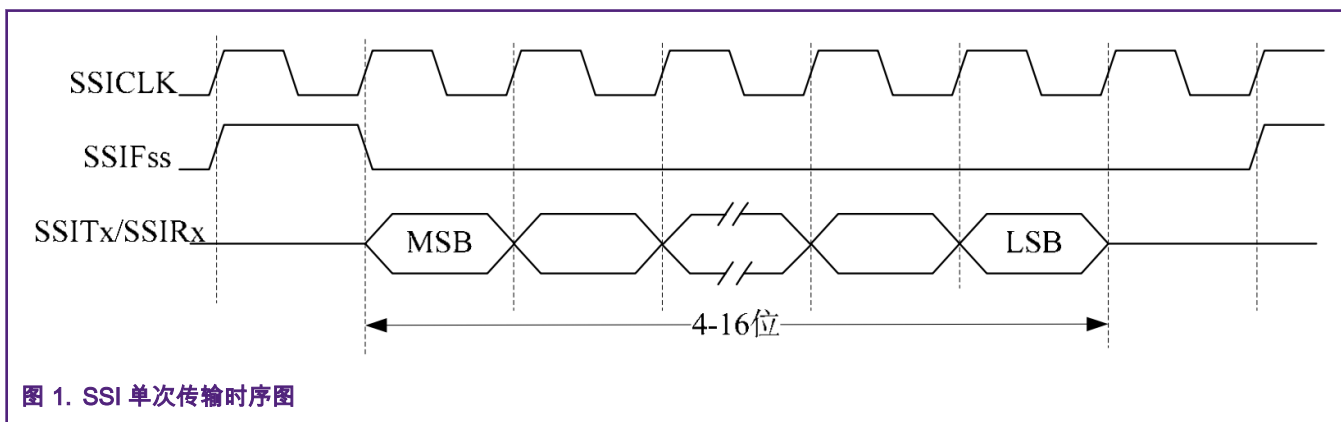


图 1. SSI 单次传输时序图

3 硬件平台

为了模拟同步串行接口（SSI）的 master（例如控制器）和 slave（例如传感器）之间的通信，在此演示中使用了两块 i.MXRT1010 EVK 板子。一块板用于模拟 SSI master 设备，另一块板用于模拟 SSI slave 设备。另外，为了能够成功演示该 demo，需要对 i.MX RT1010 EVK 板子进行如下操作和改动。

SSI master 板子和 SSI slave 板子之间按如下关系连接：

表 1. SSI 主板和备板连接

引脚名称	Master 板子	连接	引脚名称	Slave 板子
SSI_RX	J26-4	<----->	SSI_TX	J26-6
SSI_CLK	J26-8	<----->	SSI_CLK	J26-8

Table continues on the next page...

目录

1 介绍.....	1
2 SSI 简介.....	1
3 硬件平台.....	1
4 SSI 模拟.....	3
4.1 FlexIO 简介.....	3
4.2 SSI master 配置.....	4
4.3 SSI slave 配置.....	5
4.4 Demo 演示.....	7
5 总结.....	8
6 参考资料.....	8

表 1. SSI 主板和备板连接 (续)

引脚名称	Master 板子	连接	引脚名称	Slave 板子
SSI_Fss	J56-10	<----->	SSI_Fss	J56-10
SSI_TX	J26-6	<----->	SSI_RX	J26-4
GND	J1-10	<----->	GND	J1-10

在 i.MX RT1010 上，FLEXIO 共有 27 个引脚。在本应用中，有 4 根 FlexIO 引脚分别用于模拟 SSI_Fss 引脚，SSI_RX 引脚，SSI_TX 引脚和 SSI_CLK。表 2 给出了这 4 根 FlexIO 引脚及板子对应的位置。

表 2. FlexIO 引脚使用

FlexIO 引脚	引脚位置
flexio1.FLEXIO21	J26-4
flexio1.FLEXIO22	J26-6
flexio1.FLEXIO26	J26-8
flexio1.FLEXIO00	J56-10

实际的硬件平台如 图 2 所示，为了能够使本应用笔记中例程能够成功演示，还需要对板子进行以下改动：

- 去掉电阻 R792，并在 RT1010 EVK 板 R800 处焊接 0Ω 电阻。
- 将 ISP 拨码开关 SW8 更改为 0b0010。
- 将 J1-1 和 J1-2 引脚用短路帽连接。
- 将 USB 插到板上的 J41 进行供电。



图 2. 硬件连接

4 SSI 模拟

这一节主要介绍如何使用 FlexIO 模块来模拟 图 1 所示的 SSI 单次传输，以及详细描述 SSI 主机模式和从机模式的配置。

4.1 FlexIO 简介

FlexIO 是高度可配置的模块，支持各种协议，包括但不限于 UART，I²C，SPI，I²S 这几种，并提供多种功能，例如：

- 高度灵活的 16 位定时器，支持各种内部或外部触发，复位，使能和禁用条件。
- 可编程的与总线时钟无关的波特率，在 stop 模式下支持异步操作。
- 支持中断，DMA 或轮询的发送/接收操作。
- 用于并行接口支持的 1、2、4、8、16 或 32 位的移位宽度。
- 具有发送，接收，数据匹配模式的 32 位移位寄存器阵列，双缓冲移位器可实现连续的数据传输。

FlexIO 是一个非常灵活的模块。对于固定的时序模拟，其配置方法不是唯一的。通过不限数量的定时器和移位器，以及不同组合的多种配置方法可以实现同样的效果。本应用中分别介绍了一种实现 SSI master 和 SSI slave 的配置方法。

4.2 SSI master 配置

总共使用两个 Timer，两个 Shifter 用于模拟 SSI master 设备。Timer0 用于产生 SSI_CLK 信号，Timer1 用于产生 SSI_Fss 信号。Shifter0 连接到 SSI_TX 引脚，并在 SSI_Clk 的每个上升沿发送数据，Shifter2 连接到 SSI_RX 引脚，并在 SSI_Clk 的每个下降沿接收数据。图 3 显示了 FlexIO 模拟 SSI master 接口的内部连接关系。

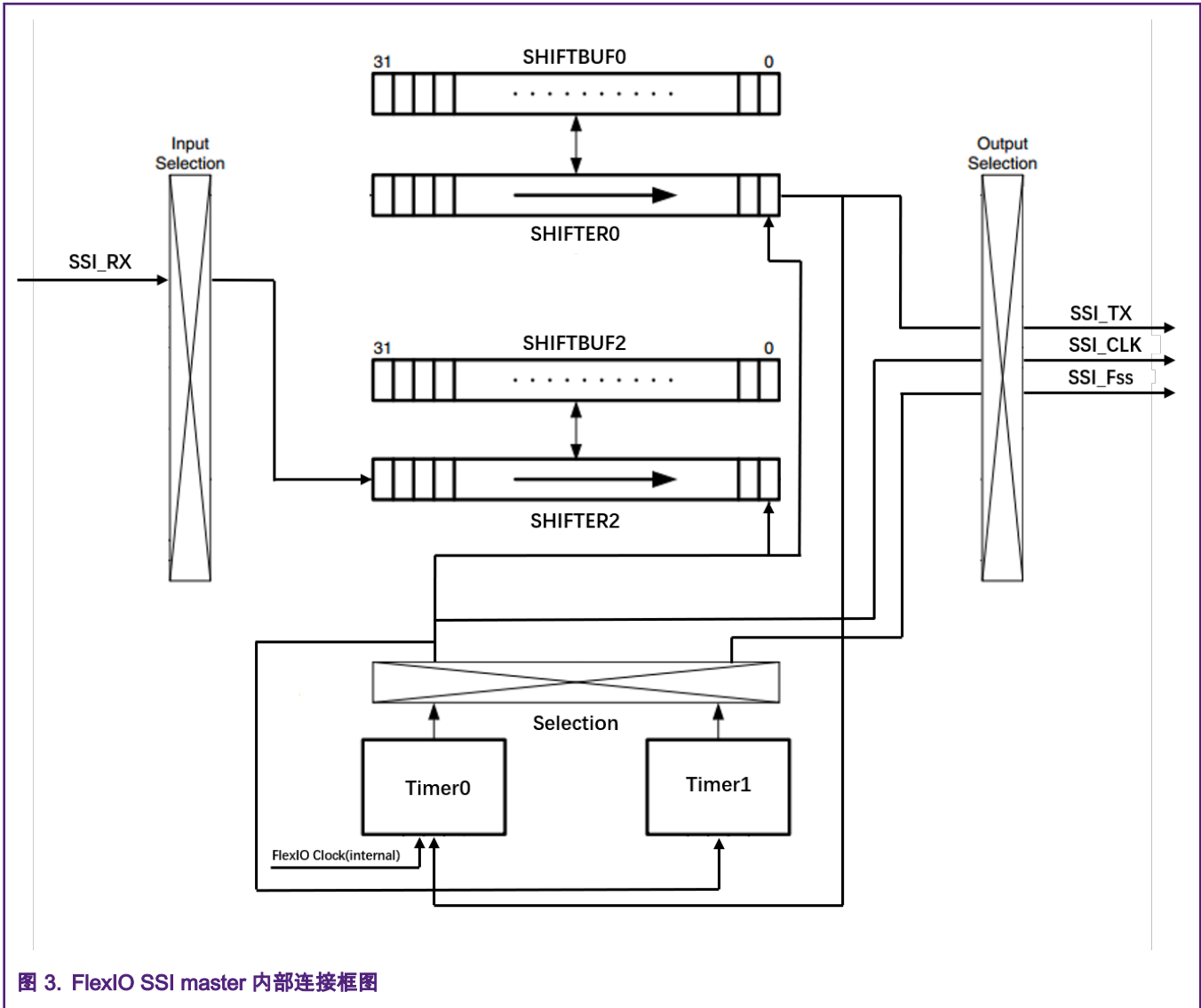


图 3. FlexIO SSI master 内部连接框图

Timer0 配置为双 8 位计数器，在触发事件高电平时被使能，在比较事件到来时禁用。Timer0 的触发源连接到内部 Shifter0，Timer0 的计数递减源配置为 FlexIO 时钟，Shift 时钟是由 Timer0 的每一次计时输出决定的，Timer0 的起始位也需要被使能。另外，Timer0 需要被配置为能够被 Shifter0 的状态标志位触发。在此应用中，由于传输频率为 200kHz，因此 Timer0 的 Compare 寄存器的值经计算得到需配置为 0xF1D。

Timer1 被配置为 16 位计数器模式，由 Timer0 触发，当 Timer0 被使能时也使能，在比较事件到来时禁用。Timer1 的计数递减源设置为触发信号的两个边沿。

Shifter0 配置为 Transmit 模式，并在移位器时钟的上升沿移位，使能移位器起始位并将其设置为逻辑低电平。

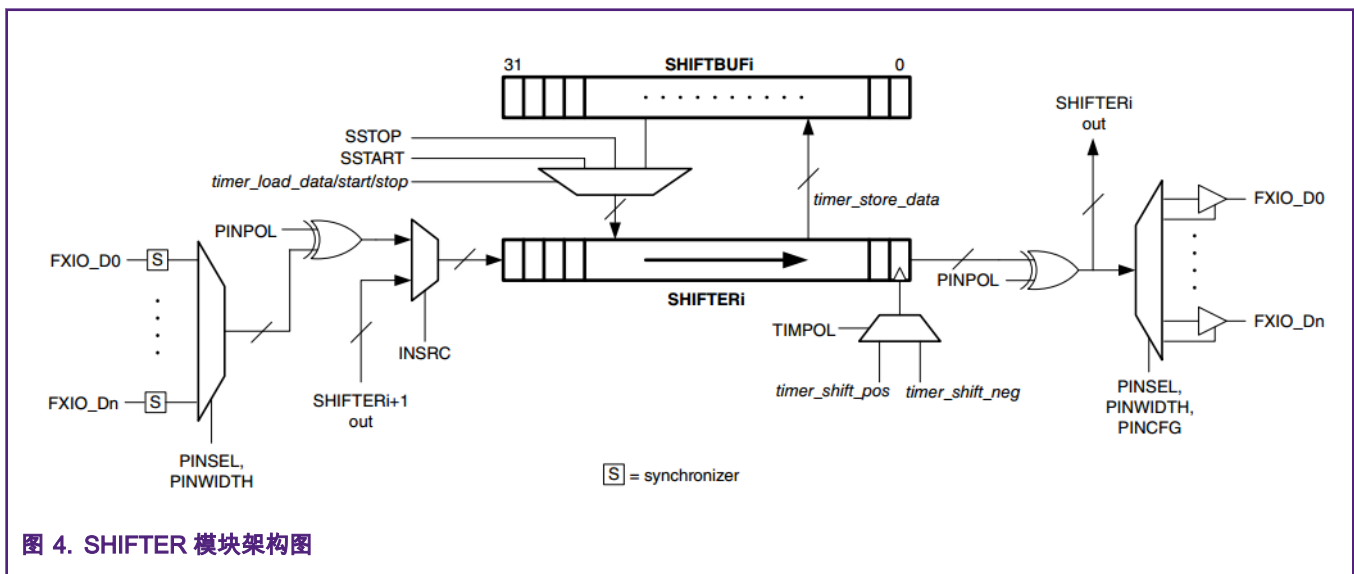
Shifter2 配置为 Receive 模式，并在移位器时钟的下降沿移位，使能移位器起始位并将其设置为逻辑低电平。

下面给出了 SSI master 模式下详细的寄存器配置：

- FIEXIO01.SHIFTCTL[0] = 0x00031502
- FIEXIO01.SHIFTCTL[2] = 0x00801601

- FIEXIO01.SHIFTCFG[2] = 0x00000002
- FIEXIO01.SHIFTCFG[2] = 0x00000002
- FIEXIO01.TIMCTL[0] = 0x01C31A01
- FIEXIO01.TIMCTL[1] = 0x03430003
- FIEXIO01.TIMCFG[0] = 0x00002222
- FIEXIO01.TIMCFG[1] = 0x00102100
- FIEXIO01.TIMCMP[0] = 0x00000F1D
- FIEXIO01.TIMCMP[1] = 0x00000002

当数据从 SHIFTER 加载到 SHIFTBUF 寄存器中或数据从 SHIFTBUF 寄存器加载到 SHIFTER 中时，如果已经将 SHIFTER 状态标志位 (SHIFTSDEN SSDE) 置 1，就可以产生一个 DMA 请求。整个 SHIFTER 的微体系结构如 图 4 所示，它充分展示了 SHIFTER 中各个模块之间的关系以及 IO 引脚输入输出的关系。



4.3 SSI slave 配置

SSI slave 的配置与 SSI master 的配置类似，SSI slave 的模拟也是需要用到两个 Shifter 和两个 Timer。Timer0 用于检测 SSI_Fss 信号，Timer1 用于检测 SSI_CLK 信号。Shifter0 连接到 SSI_TX 引脚，Shifter2 连接到 SSI_RX 引脚。图 5 显示了 FlexIO 模拟 SSI slave 接口的内部连接关系。

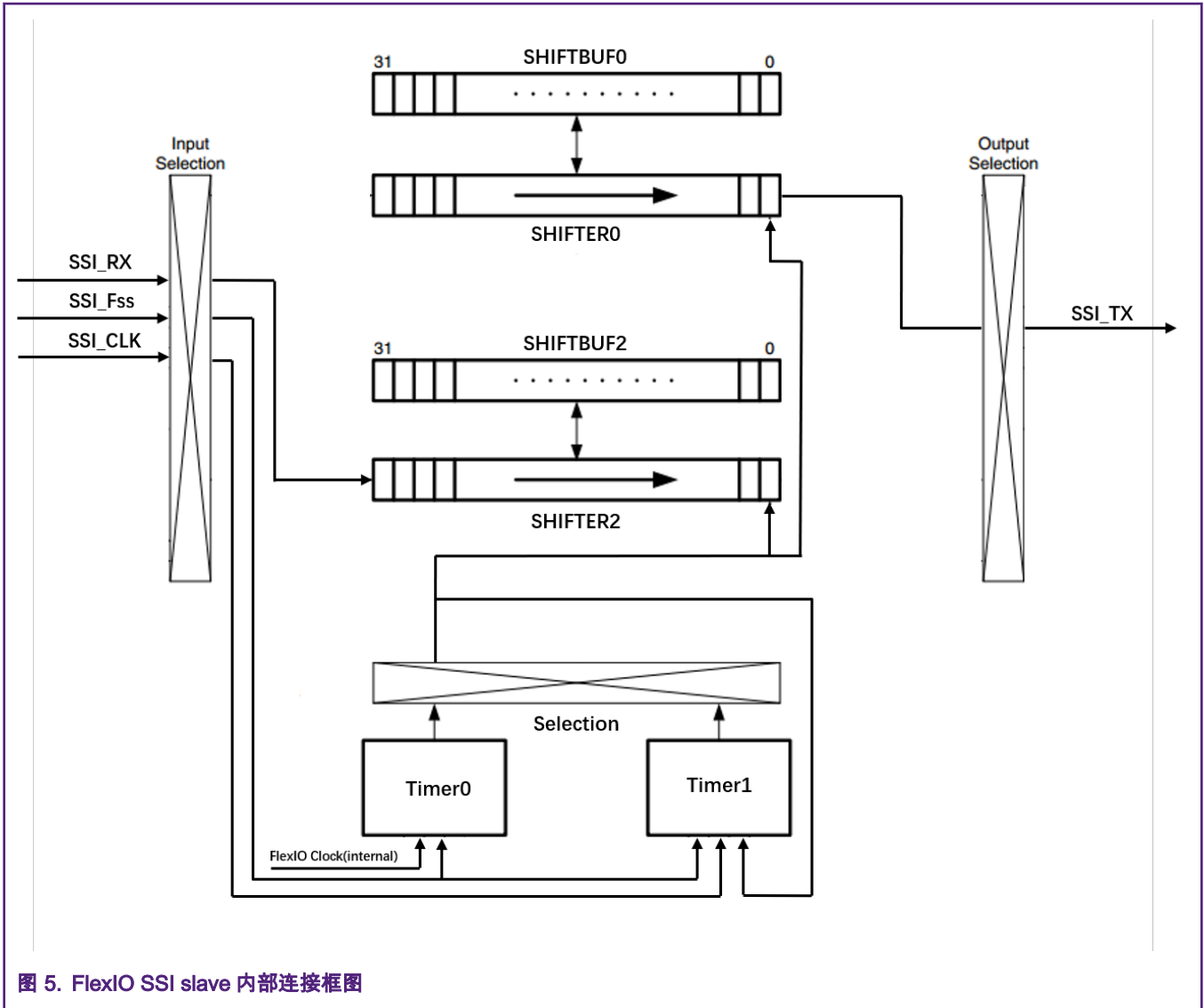


图 5. FlexIO SSI slave 内部连接框图

Timer0 配置为双 8 位计数器，在触发事件高电平时被使能，在比较事件到来时禁用。Timer0 的触发源是 SSI_Fss 输入引脚。Timer0 的计数递减源配置为 FlexIO 时钟，Shift 时钟是由 Timer0 的每一次计时输出决定的，Timer0 的起始位也需要被使能。同样，由于传输频率为 200kHz，因此 Timer0 的 Compare 寄存器的值经计算得到为 0xF1D。

Timer1 被配置为 16 位计数器模式，将 SSI_Fss 引脚的输入作为触发事件，并在触发信号的上升沿使能，在 Timer0 禁用时也禁用。Timer1 的计数递减源设置为 SSI_CLK 输入引脚的两个边沿。

Shifter0 配置为 Transmit 模式，在移位器时钟的上升沿进行移位操作，Shifter0 的起始位禁用并使其在第一次移位操作时加载数据。

Shifter2 配置为 Receive 模式，在移位器时钟的下降沿进行移位操作，Shifter2 的起始位禁用并使其在第一次移位操作时加载数据。

下面给出了 SSI slave 模式下详细的寄存器配置：

- FIEXIO01.SHIFTCTL[0] = 0x00031502
- FIEXIO01.SHIFTCTL[2] = 0x00801601
- FIEXIO01.SHIFTCFG[2] = 0x00000001
- FIEXIO01.SHIFTCFG[2] = 0x00000001
- FIEXIO01.TIMCTL[0] = 0x00400001
- FIEXIO01.TIMCTL[1] = 0x00401A03

- FIEXIO01.TIMCFG[0] = 0x00002402
- FIEXIO01.TIMCFG[1] = 0x01201600
- FIEXIO01.TIMCMP[0] = 0x00000F1D
- FIEXIO01.TIMCMP[1] = 0x0000000F

4.4 Demo 演示

下面是以两块 i.MX RT1010 EVK 板子之间的 SSI 通信为例进行的 demo 演示。首先，将各个 SSI 信号引脚连接到示波器，板上电后，抓取两块板子之间的第一次 SSI 通信数据，其波形如图 6 所示。SSI master 板子将数据 0xC5 发送到 SSI slave 板子，并同时接收到数据 0xB5，可以看出数据与图中的波形匹配，达到预期的 SSI 模拟效果。

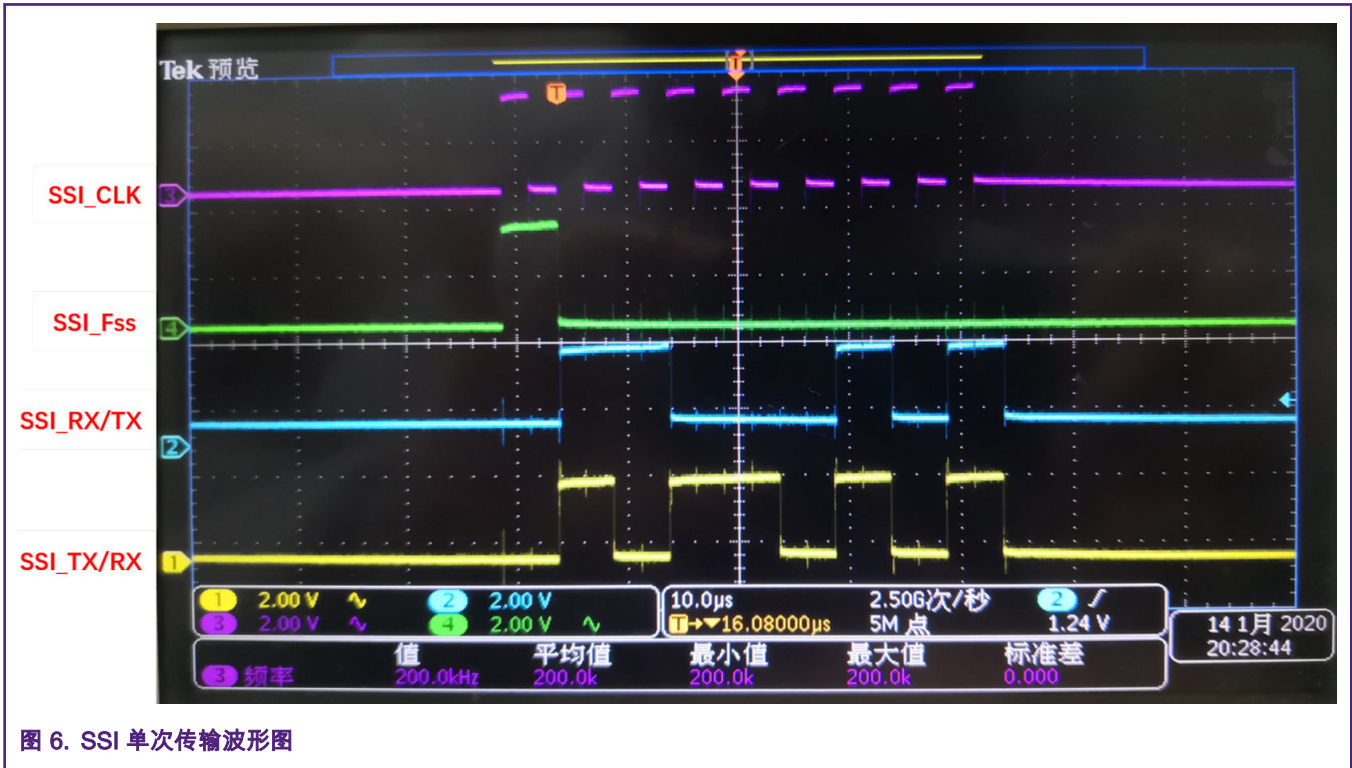


图 6. SSI 单次传输波形图

将例程下载到 i.MX RT1010EVK 板并上电，SSI master 和 SSI slave 之间的通信过程如图 7 所示。

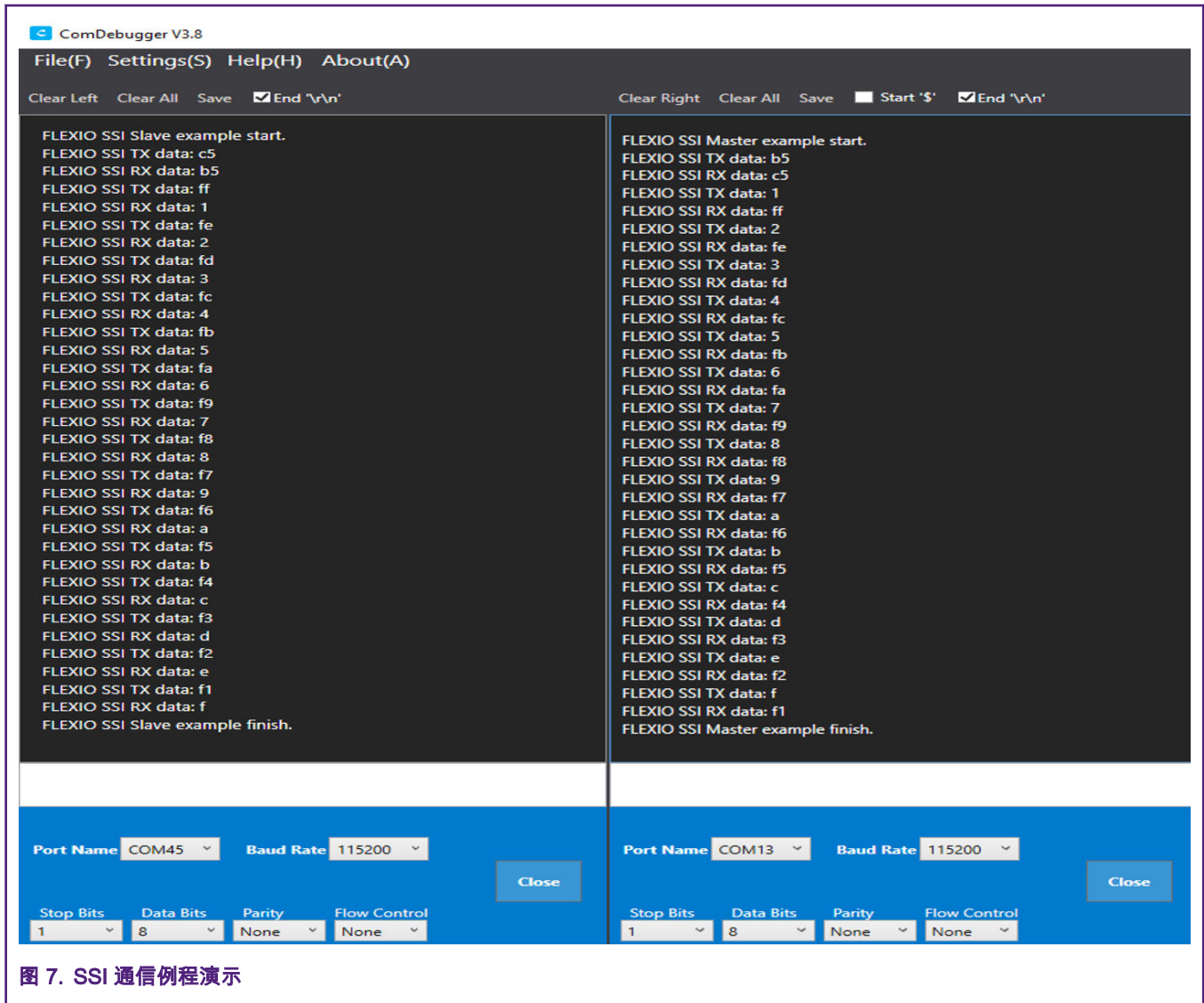


图 7. SSI 通信例程演示

5 总结

本应用笔记介绍了如何通过 i.MX RT1010 提供的 FlexIO 模块来模拟 SSI 总线接口。当 CPU 硬件资源不足时，FlexIO 可以模拟 SSI 接口用于和传感器设备进行通信。使用 FlexIO 模块模拟 SSI 接口时，有以下三个注意事项：

- 由于 FlexIO 同步延迟，串行输入数据的建立时间为 1.5 个 FlexIO 时钟周期，此时 SSI 的最大时钟频率应为 FlexIO 时钟频率的四分之一。
- 由于 FlexIO 同步延迟，串行输出数据的输出有效时间为 2.5 个 FlexIO 时钟周期，此时 SSI 的最大波特率应为 FlexIO 时钟频率的六分之一。
- 除了本文给出的 Timer 和 SHIFTER 的配置外，用户也可以利用其它配置模拟 SSI 接口，这个方法不是唯一的。

6 参考资料

- *i.MX RT1010 Processor Reference Manual (Rev. B, 07/2019)* (document [IMXRT1010RM](#))
- *Emulating SSI Using FlexIO* (document [AN5397](#))

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: May 2020
Document identifier: AN12759

