Application Note

AN_374

FT90x UART to SPI Bridge

Version 1.0

Issue Date: 2015-10-06

This note describes the implementation and usage of the FT90x UART to SPI Master bridge.

Future Technology Devices International Limited (FTDI)
Unit 1, 2 Seaward Place, Glasgow G41 1HH, United Kingdom
Tel.: +44 (0) 141 429 2777 Fax: + 44 (0) 141 429 2758
Web Site: http://ftdichip.com

Table of Contents

1 Introduction 2
1.1 Overview 2
1.2 Licence 2
2 Protocol 3
3 Implementation 4
3.1 Setup..4
3.2 Transfer Logic 5
4 Using the GUI 6
4.1 Selecting a Serial Port 6
4.2 Transferring Data 6
4.3 Troubleshooting 7
5 Contact Information 8
Appendix A – References 9
Document References 9
Acronyms and Abbreviations..9
Appendix B – List of Tables & Figures 10
List of Tables..10
List of Figures 10
Appendix C – Revision History 11

1 Introduction

This Application Note describes and explains the FT90x UART to SPI Bridge. The FT90x UART to SPI Bridge allows the SPI Master device on the FT90x chip to be controlled via commands issued over a UART.

1.1 Overview

This document describes the design and implementation of the FT90x UART to SPI Bridge. The FT90x UART to SPI Bridge allows a user to transmit and receive bytes over the SPI Master.

This document is intended to demonstrate the bridging capabilities of the FT90x family of microcontrollers.

1.2 License

© Copyright 2015, Future Technology Devices International Ltd.

This source code ("the Software") is provided by Future Technology Devices International Limited ("FTDI") subject to the license terms set out http://www.ftdichip.com/FTSourceCodeLicenceTerms.htm ("the License Terms"). You must read the License Terms before downloading or using the Software. By installing or using the Software you agree to the License Terms. If you do not agree to the License Terms then do not download or use the Software.

Without prejudice to the License Terms, here is a summary of some of the key terms of the License Terms (and in the event of any conflict between this summary and the License Terms then the text of the License Terms will prevail).

The Software is provided "as is". There are no warranties (or similar) in relation to the quality of the Software. You use it at your own risk. The Software should not be used in, or for, any medical device, system or appliance. There are exclusions of FTDI liability for certain types of loss such as: special loss or damage; incidental loss or damage; indirect or consequential loss or damage; loss of income; loss of business; loss of profits; loss of revenue; loss of contracts; business interruption; loss of the use of money or anticipated savings; loss of information; loss of opportunity; loss of goodwill or reputation; and/or loss of, damage to or corruption of data.

There is a monetary cap on FTDI's liability.

The Software may have subsequently been amended by another user and then distributed by that other user ("Adapted Software"). If so that user may have additional license terms that apply to those amendments. However, FTDI has no liability in relation to those amendments.

2 Protocol

The FT90x UART to SPI Bridge protocol is broken up into two parts: the length and the payload. The length is sent first as a single byte, this indicates the number of payload bytes which will follow this. An example of a packet is shown in Figure 1: UART to SPI Packet Format

.

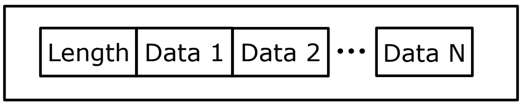

Figure 1: UART to SPI Packet Format

Upon receiving the Length byte, the SPI Master will assert the relevant SS (Slave Select) line low. With every Data byte, this byte will be sent out on SPI and the received byte will be sent back over the UART (as shown in Figure 2: Timing Diagram for a SPI Transfer

).

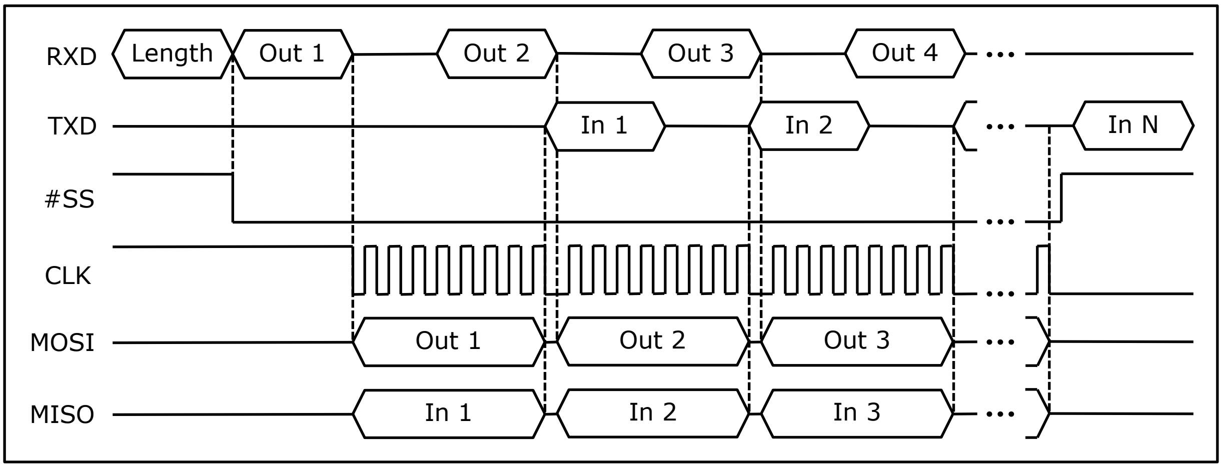

Figure 2: Timing Diagram for a SPI Transfer

This format was chosen in order to facilitate multi-byte transfers required by modern SPI devices.

3 Implementation

All the source files for the FT90x firmware are located in the src directory.

3.1 Setup

The setup_uart function, location in main.c, will set up UART0 and apply a given divider as well as set the UART for 8 data bits, no parity, and 1 stop bit.

```
void setup_uart(uint16_t divider)
{
    sys_enable(sys_device_uart0);

    gpio_function(48, pad_uart0_txd); /* UART0 TXD */
    gpio_function(49, pad_uart0_rxd); /* UART0 RXD */

    uart_open(UART0,                    /* Device */
              1,                        /* Prescaler = 1 */
              divider,
              uart_data_bits_8,         /* No. Data Bits */
              uart_parity_none,         /* Parity */
              uart_stop_bits_1);        /* No. Stop Bits */

}
```

Table 1: Code listing for setup_uart()

The setup_spi function, located in main.c, sets up the SPI Master device for a given channel, mode and divider.

```
void setup_spi(uint8_t channel, spi_clock_mode_t mode, uint16_t divider)
{
    /* Enable the SPI Device */
    sys_enable(sys_device_spi_master);

    gpio_function(27, pad_spim_sck);
    gpio_function(28, pad_spim_ss0);
    gpio_function(29, pad_spim_mosi);
    gpio_function(30, pad_spim_miso);
    gpio_function(31, pad_spim_io2);
    gpio_function(32, pad_spim_io3);
    gpio_function(33, pad_spim_ss1);
    gpio_function(34, pad_spim_ss2);
    gpio_function(35, pad_spim_ss3);

    spi_channel = channel;

    if (-1 == spi_init(SPIM, spi_dir_master, mode, divider))
        uart_puts(UART0, "spi_init ERROR\r\n");
}
```

Table 2: Code listing for setup_spi()

The function setup(), location in main.c, will call setup_uart() and setup_spi() in order to set up the UART for 19200 baud, and set up the SPI Master for channel SPI_SSx (defined at the top of main.c), SPI Mode 0 and a divider of 16 (fclk = 100MHz / 16 = 16.667 MHz).

```
void setup()
{
    /* Enable the UART Device... */
    setup_uart(UART_DIVIDER_19200_BAUD);

    /* Enable the SPI Device */
    setup_spi(SPI_SSx, spi_mode_0, 16);
}
```

Table 3: Code listing for setup()

3.2 Transfer Logic

The function loop, located in main.c, is the main logic for transferring SPI data.

```
void loop()
{
    static uint8_t bytes = 0;
    static uint8_t c;

    uart_read(UART0, &c);

    if (bytes == 0)
    {
        bytes = c;
        spi_open(SPIM, spi_channel);
    }
    else
    {
        spi_write(SPIM, c);
        spi_read(SPIM, &c);

        bytes--;
        if (bytes == 0)
            spi_close(SPIM, spi_channel);

        uart_write(UART0, c);
    }

}
```

Table 4: Code listing for loop()

This will implement the behavior shown in Figure 2: Timing Diagram for a SPI Transfer

4 Using the GUI

A Java-based GUI application is provided with this Application Note as an example of how the UART to SPI Bridge can be used.

This application is located within ui/exe with the accompanying source located at ui/src.

In order to launch the application, double-click on the app.jar file located within ui/exe.

4.1 Selecting a Serial Port

When opening the GUI application, the first window that appears will ask you to choose which serial port to use.

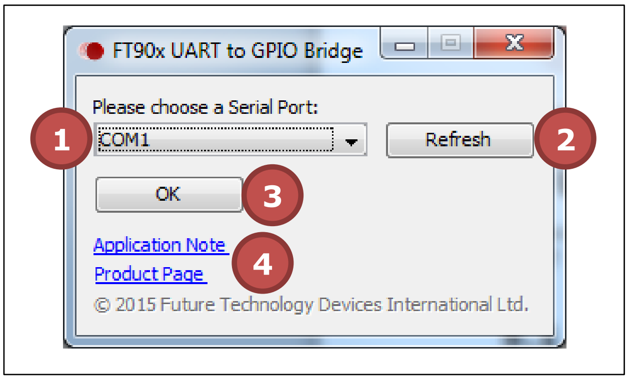

Figure 3: Serial Port Selection Window Example

Figure 3: Serial Port Selection Window Example

shows an example of a Serial Port Selection Window. Within it are these main controls:

1. **The Serial Port combo box.** Use this to select which serial port to use.
2. **The Refresh button.** If the serial port you wish to use is not shown, then it is currently in use by another application. After freeing up the serial port you wish to use, click this button to cause the GUI application to rescan for available serial ports.
3. **The OK button.** Click this to confirm your selection and progress to the next screen.
4. **Links.** Some links to the FT90x product page and a link to the application note corresponding to this application.

4.2 Transferring Data

After selecting a serial port, the GUI will show the SPI Transfer Window (shown in Figure 4: SPI Transfer Window Example

).

Within this window are these main controls:

1. **Transmit Format.** This control selects which format the input in the Transmit Text Box (2) should be parsed as.
2. **Transmit Text Box.** This is the data to be transmitted. Each line represents a separate transfer.
3. **Transmit Button.** Clicking this button will start the transfer.
4. **Clear Button.** Clicking this button will clear the Transmit Text Box (2).
5. **Load Button.** Clicking this button will allow the user to load in a text file into the Transmit Text Box (2).
6. **Received Format.** This control selects which format the input in the Received Text Box (7) should be displayed as. Changing this option will automatically update the Text Box without having to start another transfer.

7. **Received Text Box.** This is the data received over SPI. Each line represents a separate transfer.
8. **Status Bar.** This will display any errors or a copyright notice if no errors have occurred in this transfer.

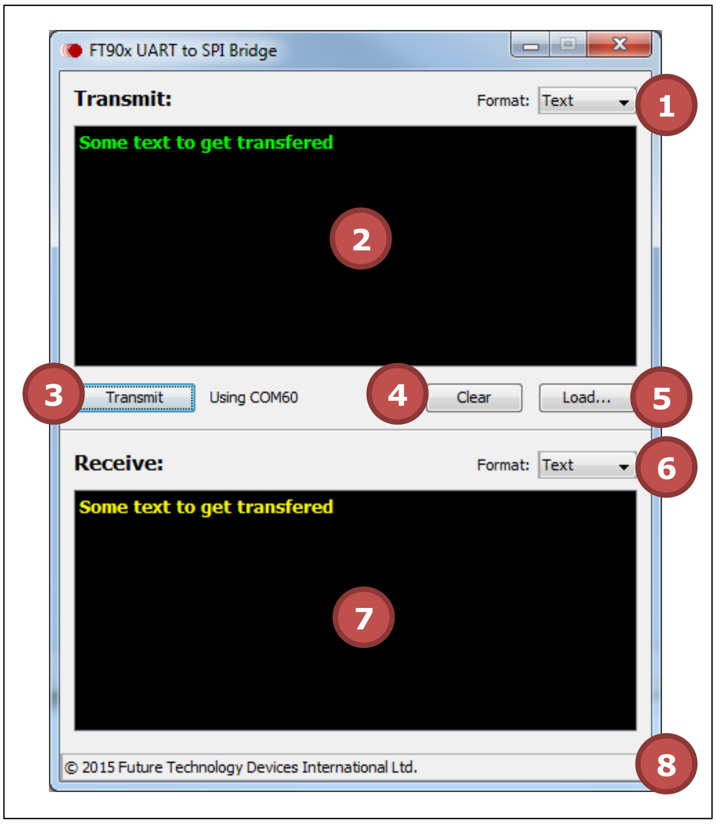

Figure 4: SPI Transfer Window Example

4.3 Troubleshooting

Q: I get *Error: Unknown Number. For input string: "…"* at the bottom of the window when trying to transmit.

A: You have either entered a number in the wrong format, or it is too large. Decimal and Hexadecimal numbers need to be separated by spaces and only numbers from 0 to 255 (FF_h) are usable.

5 Contact Information

Head Office – Glasgow, UK

Future Technology Devices International Limited
Unit 1, 2 Seaward Place, Centurion Business Park
Glasgow G41 1HH
United Kingdom
Tel: +44 (0) 141 429 2777
Fax: +44 (0) 141 429 2758

E-mail (Sales) sales1@ftdichip.com
E-mail (Support) support1@ftdichip.com
E-mail (General Enquiries) admin1@ftdichip.com

Branch Office – Tigard, Oregon, USA

Future Technology Devices International Limited
(USA)
7130 SW Fir Loop
Tigard, OR 97223-8160
USA
Tel: +1 (503) 547 0988
Fax: +1 (503) 547 0987

E-Mail (Sales) us.sales@ftdichip.com
E-Mail (Support) us.support@ftdichip.com
E-Mail (General Enquiries) us.admin@ftdichip.com

Branch Office – Taipei, Taiwan

Future Technology Devices International Limited
(Taiwan)
2F, No. 516, Sec. 1, NeiHu Road
Taipei 114
Taiwan , R.O.C.
Tel: +886 (0) 2 8791 3570
Fax: +886 (0) 2 8791 3576

E-mail (Sales) tw.sales1@ftdichip.com
E-mail (Support) tw.support1@ftdichip.com
E-mail (General Enquiries) tw.admin1@ftdichip.com

Branch Office – Shanghai, China

Future Technology Devices International Limited
(China)
Room 1103, No. 666 West Huaihai Road,
Shanghai, 200052
China
Tel: +86 21 62351596
Fax: +86 21 62351595

E-mail (Sales) cn.sales@ftdichip.com
E-mail (Support) cn.support@ftdichip.com
E-mail (General Enquiries) cn.admin@ftdichip.com

Web Site

http://ftdichip.com

Distributor and Sales Representatives

Please visit the Sales Network page of the FTDI Web site for the contact details of our distributor(s) and sales representative(s) in your country.

Appendix A – References

Document References

FT900/901/902/903 Datasheet

FT905/906/907/908 Datasheet

FT900 User Manual

FT900 code source

Windows utility

Acronyms and Abbreviations

Terms	Description
GUI	Graphical User Interface
MISO	Master In Slave Out
MOSI	Master Out Slave In
RXD	Received Data (UART)
SCK	Serial Clock
SPI	Serial Peripheral Interface
SS	Slave Select
TXD	Transmitted Data (UART)
UART	Universal Asynchronous Receiver Transmitter

Appendix B – List of Tables & Figures

List of Tables

Table 1: Code listing for setup_uart() 4
Table 2: Code listing for setup_spi() 4
Table 3: Code listing for setup() 5
Table 4: Code listing for loop() 5

List of Figures

Figure 1: UART to SPI Packet Format........ 3
Figure 2: Timing Diagram for a SPI Transfer........ 3
Figure 3: Serial Port Selection Window Example........ 6
Figure 4: SPI Transfer Window Example 7

Appendix C – Revision History

Document Title:	AN_374 FT90x UART to SPI Bridge
Document Reference No.:	FT_001170
Clearance No.:	FTDI# 448
Product Page:	http://www.ftdichip.com/FTProducts.htm
Document Feedback:	Send Feedback

Revision	Changes	Date
1.0	Initial Release	2015-10-06