Application Note

AN_208

FT311D_Demo_APK_User_Guide

Version 1.0

Issue Date: 2012-07-20

FTDI's FT311D device is targeted specifically at providing a data bridge from an Android USB device port to alternative interfaces such as GPIO, UART, PWM, I^2C or SPI. To use the device requires JAVA applications developed to run on Android platforms that support Open Accessory Mode (3.1 onwards). This user guide describes how to use the sample applications FTDI has provided.

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

FTDI's FT311D device is targeted specifically at providing a data bridge from an Android USB device port to alternative interfaces such as GPIO,UART, PWM, I^2C or SPI. To use the device requires JAVA applications developed to run on Android platforms. This user guide describes how to use the sample applications FTDI has provided.

1.1 Overview

Android is a commonly used operating system, mostly applied in portable devices such as mobile phones and tablet computing devices. Sometimes it is still desirable to connect these devices to external peripherals ("accessories" in Android terminology). One possible solution is to use USB. To help make this bridge, the FT311D IC from FTDI will enumerate the USB device port of an Android device that supports Android Open Accessory Mode – Android version 3.1 onwards. This application note shows how to run the demonstration applications that FTDI has provided. Although source code is provided, it is not discussed in this application note and is provided as is. Functionality is neither guaranteed nor supported.

All files may be downloaded from the FTDI website: www.FTDIchip.com/Android.htm

2 Loading Examples onto the Android

An "executable" file for android has the extension.apk. To load the file onto the Android operating system it is recommended to follow the steps below.

Step 1 is to allow non-Market applications to install. This is done from the settings window. Select security -> unknown sources, to allow the application to install.

Step 2 is to copy the .apk file to the Android device. This can be done by connecting the device directly to your desktop/laptop and simply copying the file over with File Explorer (assumes a windows PC). An Android app such as "File Expert" may then be used to browse to the file on the Android device.

Step 3 is to click (tap the touch screen) to open the apk file which should launch the application installer to allow the application to be installed. By default the application will appear in the applications window of your device. Dragging it to the desktop is optional but not essential.

3 Examples

Examples exist to verify each of the FT311D interfaces:

- GPIODemoActivity.apk
- PWMDemoActivity.apk
- UARTDemoActivity.apk
- I2CDemoActivity.apk
- SPIMasterDemoActivity.apk
- SPISlaveDemoActivity.apk

3.1 GPIO Demo

The GPIO example application is named GPIODemoActivity.apk.

3.1.1 GPIO Test Hardware

The diagram below shows how the FT311D may be connected up to be used with this demo.

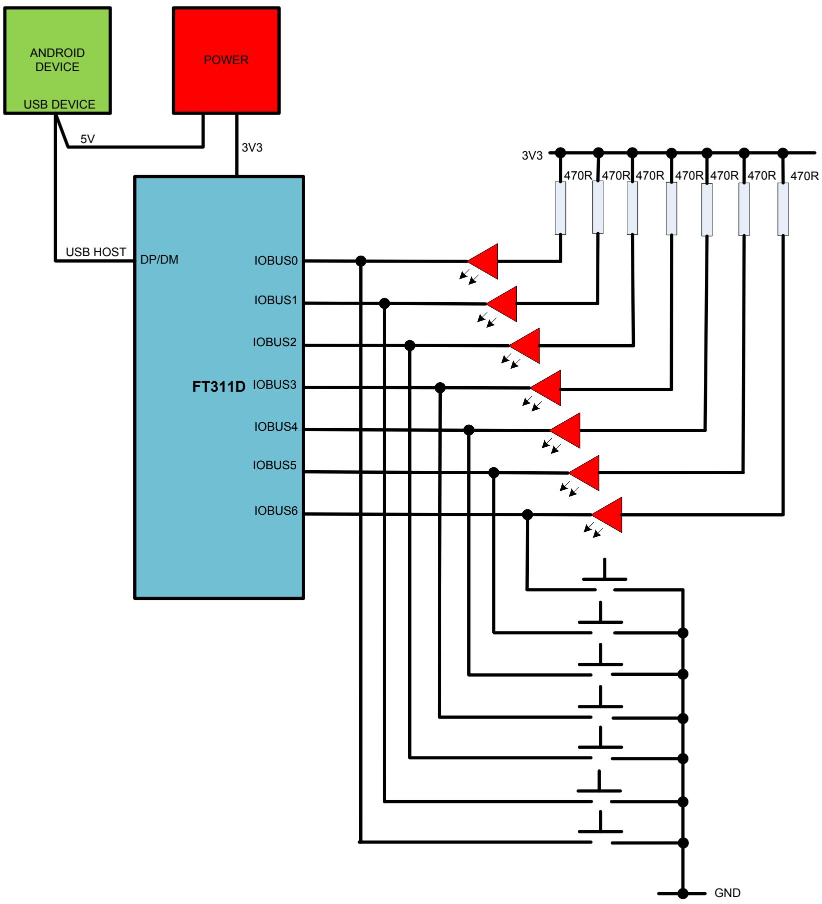

Figure 3.1 GPIO Demo Block Diagram

If the GPIO line (IOBUS0-6) is an output then the LED will be lit or unlit depending on whether the value written out is logic 1 or 0.

If the GPIO line is an input then by default it is logic 1. Pushing the switch will create a logic 0 input.

3.1.2 GPIO Application

The GUI is shown in Figure 3.2 It consists of touch key buttons to accept inputs to send to the device and display boxes to report back the status of the pins.

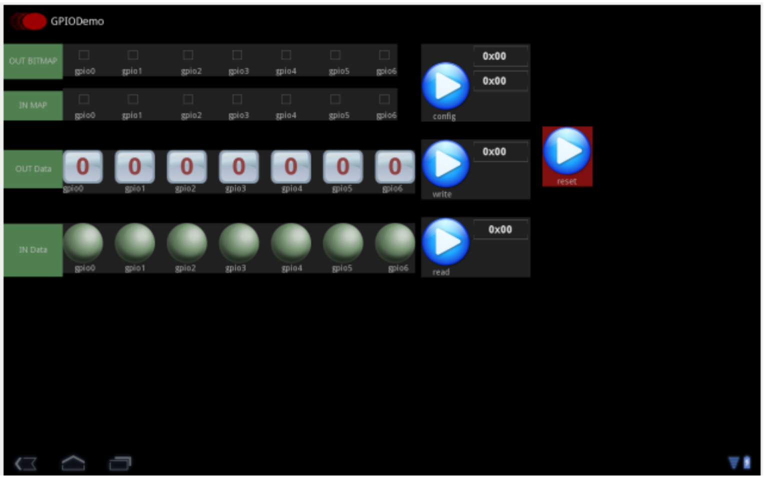

Figure 3.2 GPIO Demo Application Screen

To use the application the pins must first be defined as inputs or outputs.

To do this, tick the relevant box for the OUT BITMAP or the IN MAP. Note it is not valid to select the same pin as an input and an output at the same time.

Use the CONFIG button to send the setup to the chip.

If the pin is an output and OUT Data pin is set to 0 then the WRITE button will set the corresponding pin to logic 0. Conversely if the OUT Data pin is set to 1, the pin state is changed to logic 1 when the WRITE button is pressed. The values remain set on the output pins until the WRITE button sends new data. Pins defined as inputs are not changed by the WRITE button.

If the pin is an input then the IN Data will report back the value. If the pin is logic 0 then the green ball (LED) will change to bright green when the READ button is pressed. If the pin is logic 1 then the green ball will change to dull green. Pins designated as outputs should be ignored. The display remains fixed until the READ button updates the screen.

Note the hexadecimal value equivalent of the GPIO lines is displayed next to the command buttons.

RESET will return all pins to inputs.

3.2 UART Demo

The UART example application is named UARTDemoActivity.apk.

3.2.1 UART Test Hardware

The diagram below shows how the FT311D may be connected up to be used with this demo.

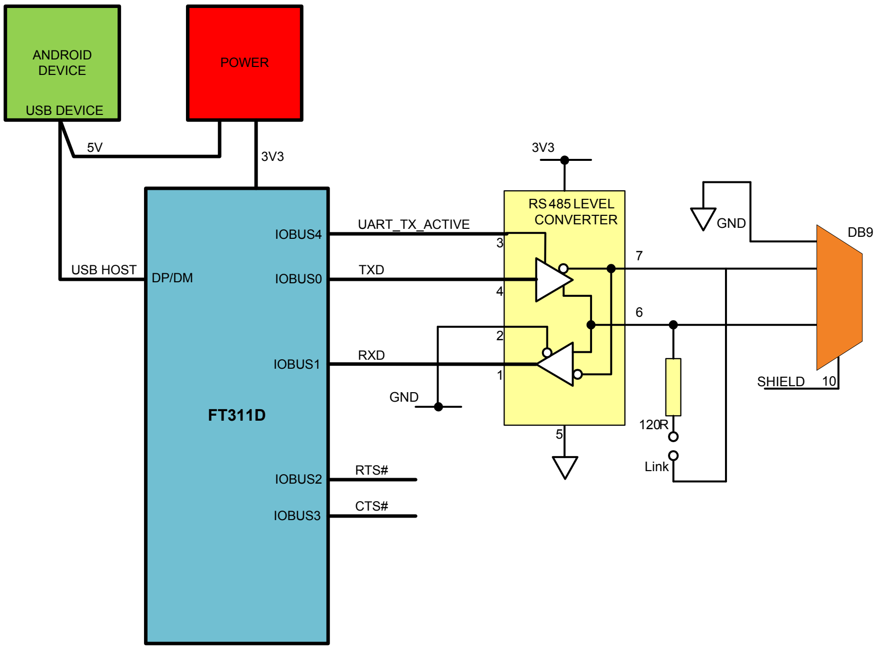

Figure 3.3 UART Demo Block Diagram

The block diagram has shown the FT311D connected to a RS485 transceiver. It is possible that the transceiver could be replaced with an RS232 or RS422 transceiver. Equally it is possible that the device may be connected at 3V3 logic levels directly to an MCU or an FPGA for UART data transfers. The block diagram shows the transmitter is only active when the FT311D is transmitting and the receive channel is always on. RTS/CTS flow control is not wired up in RS485. An external RS485 device would be required to inject data into the FT311D with this setup. USB_RS485 cables from FTDI would be an example of a suitable device for this.

3.2.2 UART Application

The GUI is shown in Figure 3.4 It consists of data boxes to accept inputs to send to the device and display boxes to report back the status of the pins.

Figure 3.4 UART Demo Application Screen

The first row of the application allows the FT311D UART interface to be configured.

The configuration settings allow the baud rate to be set at standard values between 300 and 921600 baud.

Stop bits may be set for 1 or 2.

Data bits may be set for 7 or 8

Parity may be set for ODD, EVEN, Mark, Space, None.

Flow allows for no flow control or RTS/CTS flow control.

Note you can only do this once. To change settings a second time requires the app to be restarted.

The READ BYTES box displays the data received on the FT311D RXD pin. It is currently displayed as an ascii value.

The WRITE BYTES box allows a user to type in ascii values for sending from the Android device over USB and out the FT311D TXD pin. The data is sent when the WRITE button is pressed.

Note the WRITE BYTES box only accepts 64 characters to write.

3.3 PWM Demo

The PWM example application is named PWMDemoActivity.apk.

3.3.1 PWM Test Hardware

The diagram below shows how the FT311D may be connected up to be used with this demo.

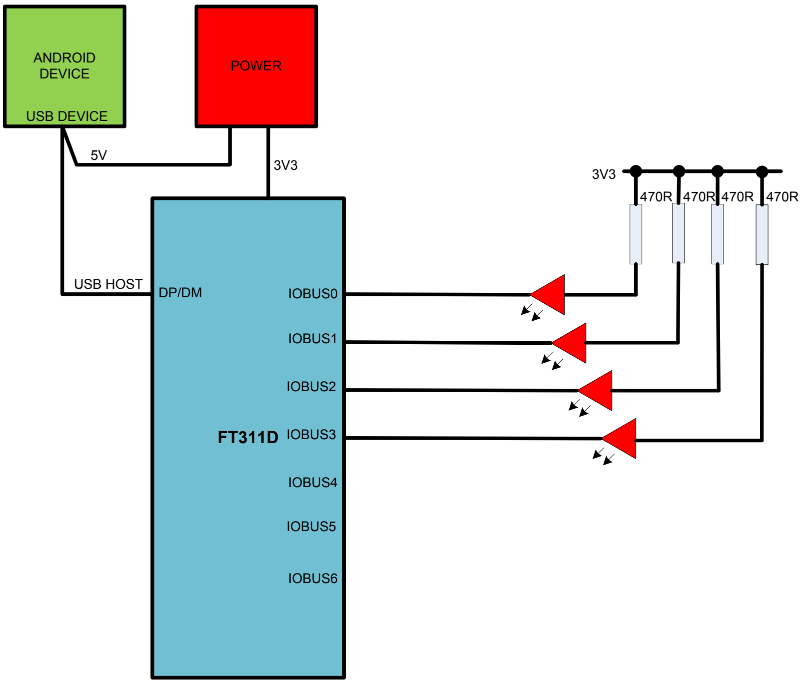

Figure 3.5 PWM Demo Block Diagram

The PWM application will output different PWM pulse trains on the 4 channels (IOBUS0-3). The result can then be observed as an LED flashing at different rates.

3.3.2 PWM Application

The GUI is shown in Figure 3.6. It consists of data boxes to accept configuration data.

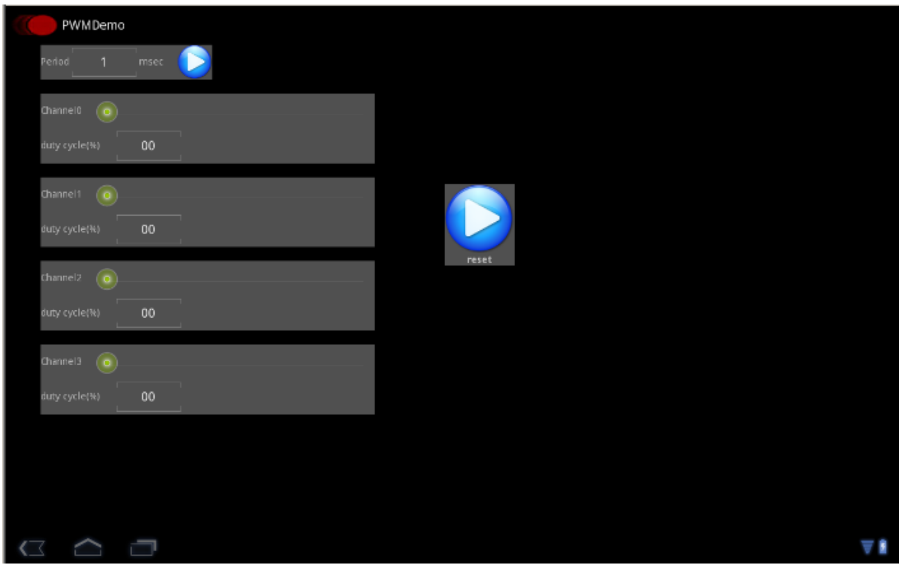

Figure 3.6 PWM Demo Application Screen

The first row of the application allows the user to enter the period for the PWM waveform. This is effectively a setting to define how often the pulse repeats.

The next 4 rows are the setting for the 4 channels. The duty cycle is defining the percentage of the period time when the signal is logic 1 or logic 0. The duty cycle may be set between 0 and 95%

Reset will reset the application to its default state.

3.4 I²C Demo

The I²C example application is named I2CDemoActivity.apk.

3.4.1 I²C Test Hardware

The diagram below shows how the FT311D may be connected up to be used with this demo.

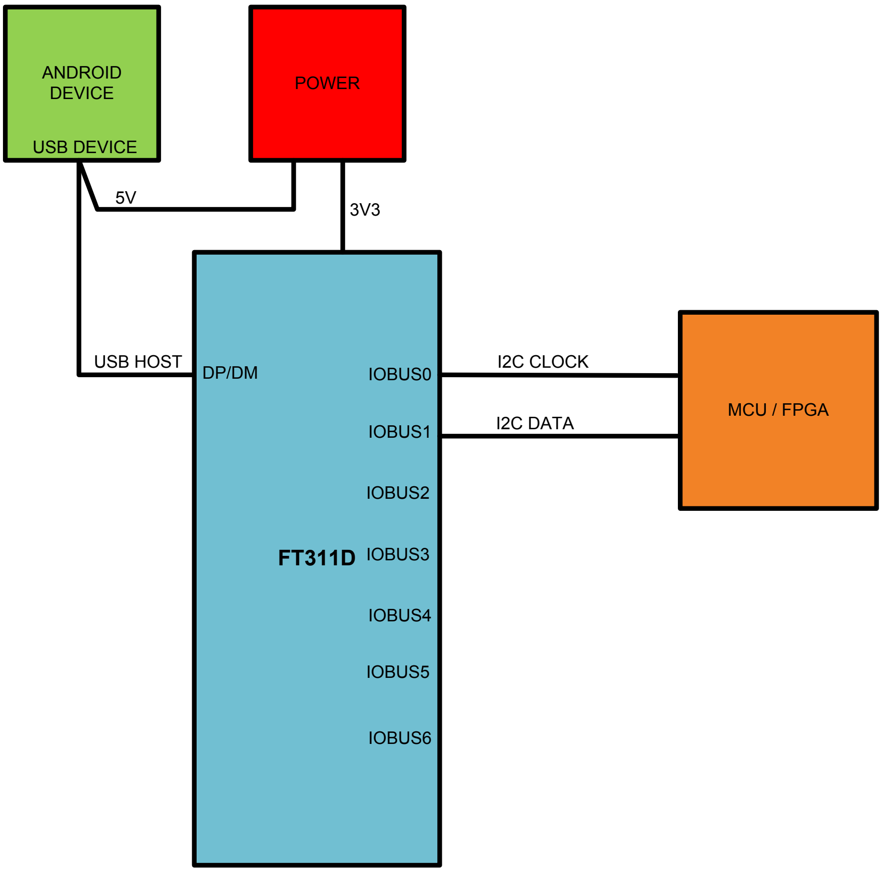

Figure 3.7 I²C Demo Block Diagram

The I2C clock is an output to the I²C slave which maybe an FPGA or an MCU with an I²C port. The I2C data line is bidirectional with the FT311D being the master of the data line.

3.4.2 I²C Application

The GUI is shown in Figure 3.8. It consists of touch key buttons and data boxes to accept inputs to send to the device and display boxes to report back the status of the pins.

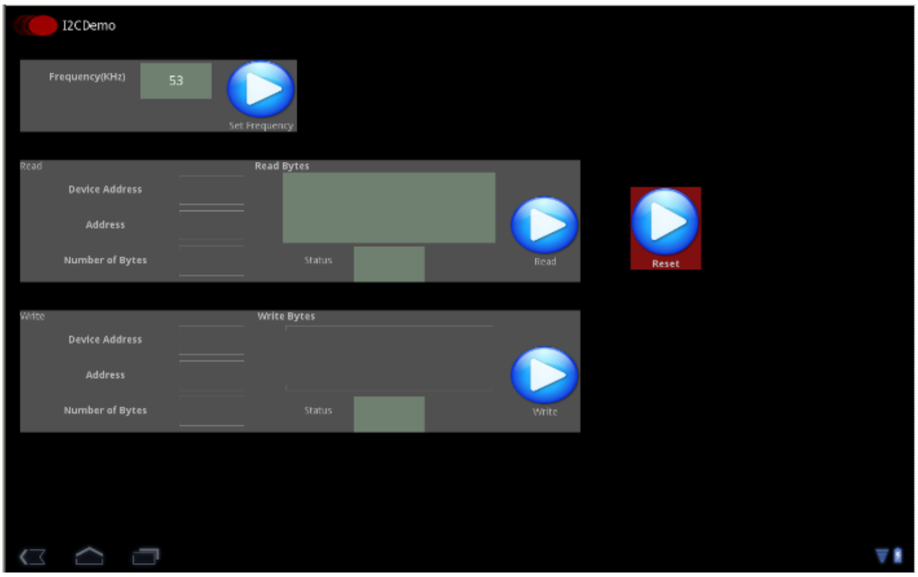

Figure 3.8 I²C Demo Application Screen

The first row sets the frequency of the clock output. The accepted range is 23, 44, 60 or 92kHz

The second row is for reading data from the I²C slave.
The user must input the Device Address that the FT311D is attempting to access.
The Address field is for any specific register in the target device which may be set to 0 if there are no specific registers e.g. a FT200XD USB to I²C bridge device.
The number of bytes to read defines the amount of data to be read.
The values should be entered as decimal values.

Pressing the READ button will update the green box with data from the I²C slave. The data will be displayed in ASCII.

The third row is for writing data to the I²C slave.
The user must input the Device Address that the FT311D is attempting to access.
The Address field is for any specific register in the target device which may be set to 0 if there are no specific registers e.g. a FT200XD USB to I²C bridge device.

The "number of bytes" field reports back the number of bytes written.

Pressing the WRITE button will send the data from into the "Write bytes" box to the I²C slave. The data should be input as ASCII.

3.5 SPI Slave Demo

The SPI Slave example application is named SPISlaveDemoActivity.apk.

3.5.1 SPI Slave Test Hardware

The diagram below shows how the FT311D may be connected up to be used with this demo.

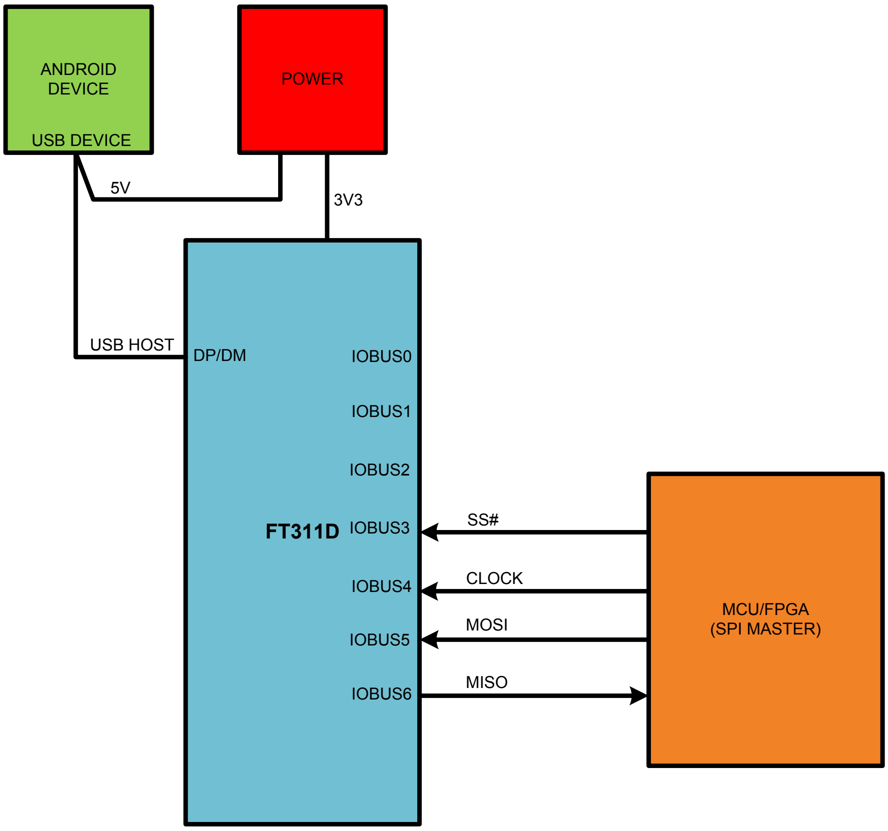

Figure 3.9 SPI Slave Demo Block Diagram

3.5.2 SPI Slave Application

The GUI is shown in Figure 3.10. It consists of touch key buttons and data boxes to accept inputs to send to the device and display boxes to report back the read data.

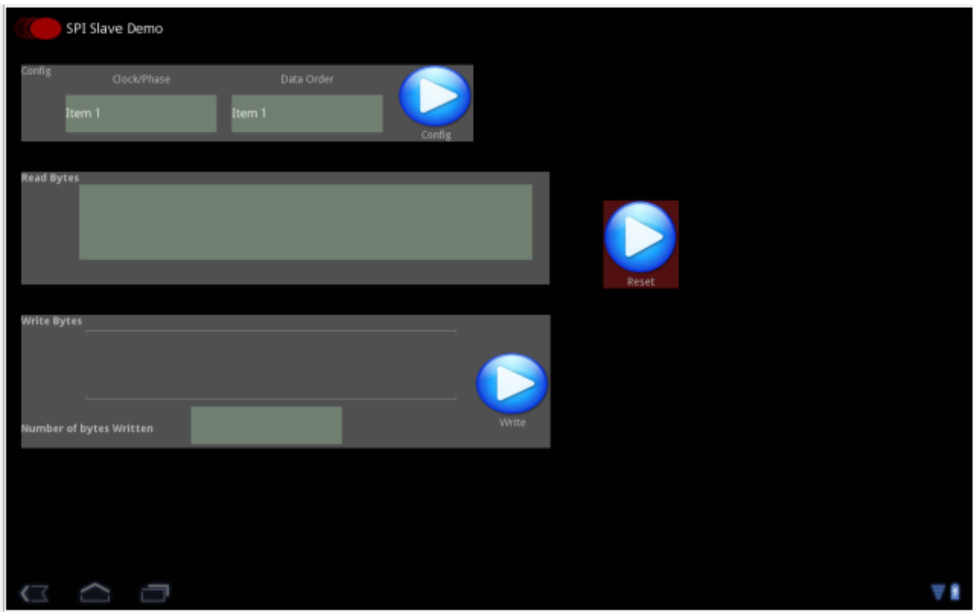

Figure 3.10 SPI Slave Demo Application Screen

The first row sets the SPI mode. The options are Mode 0, Mode 1, Mode 2 and Mode 3 with the data order being either MSB or LSB. The CONFIG button is used to send the selected option to the chip.

The second row is for displaying data sent from the external SPI master and read by the FT311. The data is displayed in ASCII.

The third row is for writing data to the SPI master. The user must input the data on ASCII. Pressing WRITE will send the data. The panel will also report back the number of bytes written in decimal.

3.6 SPI Master Demo

The SPI Master example application is named SPIMasterDemoActivity.apk.

3.6.1 SPI Master Test Hardware

The diagram below shows how the FT311D may be connected up to be used with this demo.

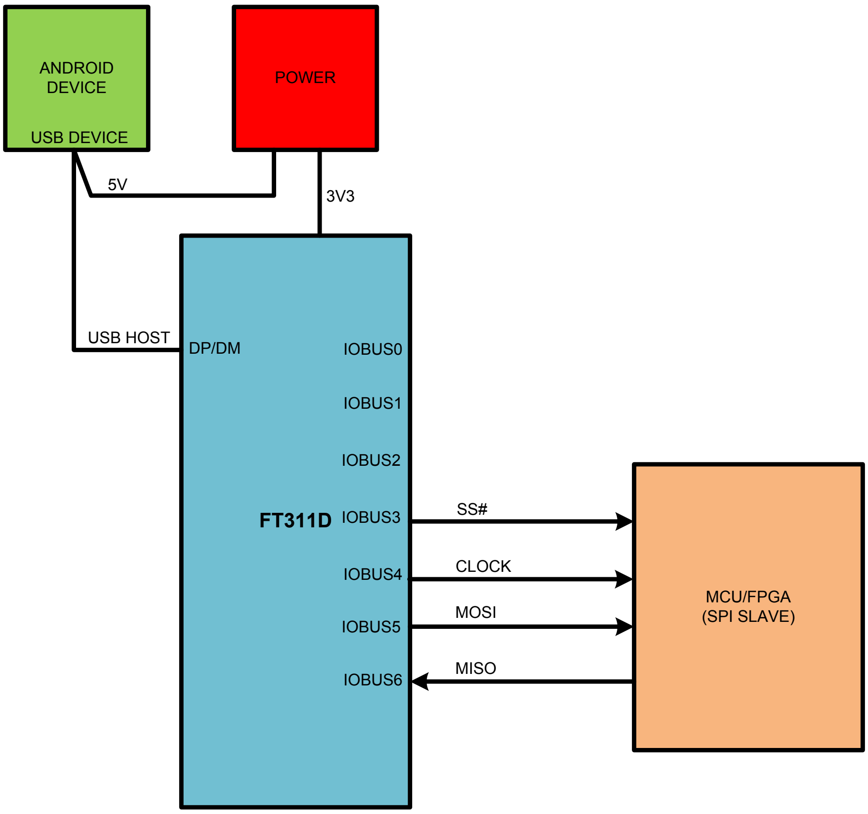

Figure 3.11 SPI Master Demo Block Diagram

3.6.2 SPI Master Application

The GUI is shown in Figure 3.12. It consists of touch key buttons and data boxes to accept inputs to send to the device and display boxes to report back the read data.

Figure 3.12 SPI Master Demo Application Screen

The first row sets the SPI mode. The options are Mode 0, Mode 1, Mode 2 and Mode 3 with the data order being either MSB or LSB.
The clock output frequency must also be set in this window. The valid range is 1 … 24000000
The CONFIG button is used to send the selected option to the chip.

The second row is for displaying data sent from the external SPI slave and read by the FT311D. The user must input the number of bytes to be read in decimal before pressing READ to get the bytes of data. Data is displayed in ASCII.

The third row is for writing data to the SPI slave. The user must input the data as ASCII. Pressing WRITE will send the data. The panel will also report back the number of bytes written in decimal.

4 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 – Hillsboro, Oregon, USA

Future Technology Devices International Limited (USA)
7235 NW Evergreen Parkway, Suite 600
Hillsboro, OR 97123-5803
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

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

Appendix A – References

Document References

FT311D Data Sheet

Acronyms and Abbreviations

Terms	Description
GPIO	General Purpose Input output
I^2C	Inter Integrated Circuit
PWM	Pulse Width Modulation
SPI	Serial peripheral interface
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
USB-IF	USB Implementers Forum

Appendix B – List of Tables & Figures

List of Tables

No table of figures entries found.

List of Figures

Appendix C – Revision History

Document Title:	AN_208 FT311_Demo_APK_User_Guide
Document Reference No.:	FT_000666
Clearance No.:	FTDI# 306
Product Page:	http://www.ftdichip.com/FTProducts.htm
Document Feedback:	Send Feedback

Revision	Changes	Date
1.0	Initial Release	2012-07-20