Technical Notes

TN_161

FT4222H Errata Technical Note

Version 1.7

Issue Date: 21-11-2024

The intention of this errata technical note is to give a detailed description of known functional or electrical issues with the FTDI FT4222H series device.

The current revision of the FT4222H series is **Revision D, released April 2018.**

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 FT4222H Revision

FT4222H part numbers are listed in Table 1-1. The letter at the end of the date code identifies the device revision.

The current revision of the FT4222H series is **Revision D, released April 2018.**

Part Number	Package
FT4222HQ	32 pin VQFN

Table 1-1 FT4222H Part Numbers

These errata technical note covers the revisions of FT4222H listed in Table 1-2.

Revision	Notes
A	First device revision. Launched Sep 2014
B	Second device revision. Launched Sep 2015
C	Third device revision. Launched Oct 2016
D	Forth device revision. Launched Apr 2018

Table 1-2 FT4222H Series Revisions

2 Errata History Table – Functional Errata

Functional Errata	Short description	Errata occurs in device revision
FT4222H	Android issues	A
FT4222H	CPU usage and latency timer issue	A
FT4222H	I²C combined message issue	A
FT4222H	Default pin status	A
FT4222H	Additional suspend settings support	A
FT4222H	Custom PID settings are ignored	B
FT4222H	Slow response after the host restarts	B
FT4222H	SPI master in single mode loses data and no response	B
FT4222H	SPI master throughput is not good enough	A, B
FT4222H	Data path is not fully reset when a reset on I2C is executed	A, B, C
FT4222H	Not Response STALL to Get BOS Descriptor defined in USB3.0	A, B, C
FT4222H	Flash operating mode support	A, B, C
FT4222H	The bulk maximum packet size for CNFMODE1/CNFMODE2 is not 512 bytes	A, B, C
FT4222H	Detectable frequency on GPIOs is not sufficient for most applications	A, B, C
FT4222H	SPI Slave may lose data when transfer frequency is set to 20MHz	A, B, C
FT4222H	I2C data is corrupt when FT4222_I2CMaster_GetStatus is being called	A, B, C, D
FT4222H	Bus Error condition in USB Device Controller	A, B, C, D
FT4222H	MISO send data at wrong timing in SPI Slave Mode	A, B, C, D
FT4222H	I2C master will continue to transmit data even if it receives a NAK after sending the address byte	A, B, C, D

Table 2-1 Functional Errata

2.1 Errata History Table – Electrical and Timing Specification Deviations

Deviations	Short description	Errata occurs in device revision
-	No known issues	-

Table 2-2 Electrical and Timing Errata

3 Functional Errata of FT4222H

3.1 Revision A

3.1.1 Android issues

Introduction:

FT4222H supports Android devices. With J2XX, it is possible to develop an app utilizing the FT4222H.

Issue:

The following issues may happen when the FT4222H connects to an Android device.

1. The FT4222H works as an SPI master, it may reset during transferring data.
2. The FT4222H works as I^2C slave, the last byte may be lost when the receiving buffer is full.

Workaround:

There are no known workarounds available. This issue is corrected at Revision B.

Package specific:

The effected packages are listed in Table 3-1.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-1 Affected Packages

3.1.2 CPU usage and latency timer issue

Introduction:

In USB, data is received from the device to the PC by a polling method. The driver will request a certain amount of data from the USB scheduler. The latency timer is provided to allow efficient polling and flushing short data packets.

Issue:

The FT4222H does not support the latency timer feature and causes the USB scheduler to be busy and uses too much CPU resource.

Workaround:

There are no known workarounds available. This issue is corrected at Revision B.

Package specific:

The effected packages are listed in Table 3-2.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-2 Affected Packages

3.1.3 I²C combined message issue

Introduction:

A master issue at least two reads and/or writes to one or more slaves. In a combined message, each read or write begins with a START and the slave address. After the first START, the subsequent starts are referred to as repeated START bits; repeated START bits are not preceded by STOP bits, which indicate to the slave the next transfer is part of the same message.

Start	7-bit slave address	Write	ACK	8-bit data	ACK	**SR**	7-bit slave address	Read	ACK	8-bit data	ACK	8-bit data	ACK	STOP

Issue:

Some I²C devices need to communicate with a combined message format. However, the FT4222H does not support this feature.

Workaround:

There are no known workarounds available. The feature of I²C combined messages will be supported at Revision B.

Package specific:

The effected packages are listed in Table 3-3.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-3 Affected Packages

3.1.4 Default pin status

Introduction:

By default, the FT4222H will be initialized as an SPI master after power on. When the FT4222H is ready, i.e., finishes USB enumeration, the status of the pins of the Revision A device is as shown in Table 3-4:

Pin num	Pin name	Mode 0	Mode 1	Mode 2	Mode 3
8	SCK	SCK (OUT, low)	SCK (OUT, low)	SCK (OUT, low)	SCK (OUT, low)
9	MISO	MISO (IN)	MISO (IN)	MISO (IN)	MISO (IN)
10	MOSI	MOSI (OUT, high)	MOSI (OUT, high)	MOSI (OUT, high)	MOSI (OUT, high)
11	IO2	IO2 (IN)	IO2 (IN)	IO2 (IN)	IO2 (IN)
12	IO3	IO3 (IN)	IO3 (IN)	IO3 (IN)	IO3 (IN)
13	GPIO0	GPIO0 (OUT, low)	SS1O (OUT, low)	SS1O (OUT, low)	GPIO0 (OUT, low)
14	GPIO1	GPIO1 (OUT, low)	SS2O (OUT, low)	SS2O (OUT, low)	GPIO1 (OUT, low)
15	GPIO2	suspend out (OUT, low)	suspend out (OUT, low)	SS3O (OUT, low)	suspend out (OUT, low)
16	GPIO3	remote wakeup (IN)	remote wakeup (IN)	remote wakeup (IN)	remote wakeup (IN)
17	SS0O	SS0O (OUT, low)	SS0O (OUT, low)	SS0O (OUT, low)	SS0O (OUT, low)

Pin num	Pin name	Mode 0	Mode 1	Mode 2	Mode 3
32	SS	SS (IN)	SS (IN)	SS (IN)	SS (IN)

Table 3-4 Revision A FT4222H ready

In the Revision B, the pin status will be changed as per Table 3-5 below:

Pin num	Pin name	Mode 0	Mode 1	Mode 2	Mode 3
8	SCK	SCK (OUT, low)	SCK (OUT, low)	SCK (OUT, low)	SCK (OUT, low)
9	MISO	MISO (IN)	MISO (IN)	MISO (IN)	MISO (IN)
10	MOSI	MOSI (OUT, high)	MOSI (OUT, high)	MOSI (OUT, high)	MOSI (OUT, high)
11	IO2	IO2 (IN)	IO2 (IN)	IO2 (IN)	IO2 (IN)
12	IO3	IO3 (IN)	IO3 (IN)	IO3 (IN)	IO3 (IN)
13	GPIO0	GPIO0 (IN)	SS1O (OUT, high)	SS1O (OUT, high)	GPIO0 (IN)
14	GPIO1	GPIO1 (IN)	SS2O (OUT, high)	SS2O (OUT, high)	GPIO1 (IN)
15	GPIO2	suspend out (OUT, low)	suspend out (OUT, low)	SS3O (OUT, high)	suspend out (OUT, low)
16	GPIO3	remote wakeup (IN)	remote wakeup (IN)	remote wakeup (IN)	remote wakeup (IN)
17	SS0O	SS0O (OUT, high)	SS0O (OUT, high)	SS0O (OUT, high)	SS0O (OUT, high)
32	SS	SS (IN)	SS (IN)	SS (IN)	SS (IN)

Table 3-5 Revision B FT4222H ready

Package specific:

The effected packages are listed in Table 3-6.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-6 Affected Packages

3.1.5 Additional Suspend Settings Supported

Introduction:

The FT4222H provides flexible settings for suspend behavior via FT_PROG. The Revision B of the FT4222H device provides additional options for customers to configure the pin status during suspend.

- SUSPEND_OUT_POL
  - **Suspend output is High active. (default)**
  - Suspend output is Low active.
- SPI_SUSPEND_MODE
  - **Disable SPI IP and make SPI pins input (tri-state). (default)**
  - Keep SPI pin status when the FT4222H suspends.
  - Enable SPI pin control. Refer to SPI_SUSPEND for detail settings.
- SPI_SUSPEND (enable by SPI_SUSPEND_MODE)
  - miso_suspend
    - push low when suspend
    - push high when suspend
  - mosi_suspend
    - push low when suspend
    - push high when suspend

- io2_io3_suspend
  - push low when suspend
  - push high when suspend
- ss0O_suspend
  - No change (default)
  - push low when suspend
  - push high when suspend

- GPIO_SUSPEND
  - gpio0_suspend
    - No change (default)
    - input (tri-state)
    - push low when suspend
    - push high when suspend
  - gpio1_suspend
    - No change (default)
    - input (tri-state)
    - push low when suspend
    - push high when suspend
  - gpio2_suspend
    - No change (default)
    - input (tri-state)
    - push low when suspend
    - push high when suspend
  - gpio3_suspend
    - No change (default)
    - input (tri-state)
    - push low when suspend
    - push high when suspend

The default pin status of the Revision A device during suspend is shown in Table 3-7.

Pin num	Pin name	Mode 0	Mode 1	Mode 2	Mode 3
8	SCK	SCK (OUT, low)	SCK (OUT, low)	SCK (OUT, low)	SCK (OUT, low)
9	MISO	MISO (OUT, low)	MISO (OUT, low)	MISO (OUT, low)	MISO (OUT, low)
10	MOSI	MOSI (OUT, low)	MOSI (OUT, low)	MOSI (OUT, low)	MOSI (OUT, low)
11	IO2	IO2 (OUT, low)	IO2 (OUT, low)	IO2 (OUT, low)	IO2 (OUT, low)
12	IO3	IO3 (OUT, low)	IO3 (OUT, low)	IO3 (OUT, low)	IO3 (OUT, low)
13	GPIO0	GPIO0 (OUT, low)	SS1O (OUT, no change)	SS1O (OUT, no change)	GPIO0 (OUT, low)
14	GPIO1	GPIO1 (OUT, low)	SS2O (OUT, no change)	SS2O (OUT, no change)	GPIO1 (OUT, low)
15	GPIO2	suspend out (OUT, high)	suspend out (OUT, high)	SS3O (OUT, no change)	suspend out (OUT, high)
16	GPIO3	remote wakeup (IN)	remote wakeup (IN)	remote wakeup (IN)	remote wakeup (IN)
17	SS0O	SS0O (OUT, no change)	SS0O (OUT, no change)	SS0O (OUT, no change)	SS0O (OUT, no change)
32	SS	SS (IN)	SS (IN)	SS (IN)	SS (IN)

Table 3-7 Revision A FT4222H suspend

In the Revision B device, the default suspend setting is changed as per Table 3-8 below:

Pin num	Pin name	Mode 0	Mode 1	Mode 2	Mode 3
8	SCK	SCK (tri-state)	SCK (tri-state)	SCK (tri-state)	SCK (tri-state)
9	MISO	MISO (IN)	MISO (IN)	MISO (IN)	MISO (IN)
10	MOSI	MOSI (IN)	MOSI (IN)	MOSI (IN)	MOSI (IN)
11	IO2	IO2 (IN)	IO2 (IN)	IO2 (IN)	IO2 (IN)
12	IO3	IO3 (IN)	IO3 (IN)	IO3 (IN)	IO3 (IN)
13	GPIO0	GPIO0 (no change)	SS1O (OUT, no change)	SS1O (OUT, no change)	GPIO0 (no change)
14	GPIO1	GPIO1 (no change)	SS2O (OUT, no change)	SS2O (OUT, no change)	GPIO1 (no change)
15	GPIO2	suspend out (OUT, high)	suspend out (OUT, high)	SS3O (OUT, no change)	suspend out (OUT, high)
16	GPIO3	remote wakeup (IN)	remote wakeup (IN)	remote wakeup (IN)	remote wakeup (IN)
17	SS0O	SS0O (OUT, no change)	SS0O (OUT, no change)	SS0O (OUT, no change)	SS0O (OUT, no change)
32	SS	SS (IN)	SS (IN)	SS (IN)	SS (IN)

Table 3-8 Revision B FT4222H suspend

Package specific:

The effected packages are listed in Table 3-9.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-9 Affected Packages

3.2 Revision B

3.2.1 Custom PID Settings are ignored

Introduction:

It is not possible to change the PID on the FT4222H from our default value of 601C to a custom value. Note, there are no problems changing the VID.

Issue:

Any changes made to the PID using the OTP are ignored and the value returns to its default state.

Workaround:

There are no known workarounds available. This issue is corrected at Revision C.

Package specific:

The effected packages are listed in Table 3-10.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-10 Affected Packages

3.2.2 Slow Response after the Host Restarts

Issue:

After the host restarts, the FT4222H may have slow response or output unexpected bytes from its USB interface.

Workaround:

There are no known workarounds available. This issue is corrected at Revision C.

Package specific:

The effected packages are listed in Table 3-11.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-11 Affected Packages

3.2.3 SPI master in single mode loses data and no response

Issue:

The SPI master in single mode may lose the last byte and then no response. This issue may be observed easily in the following configurations:

- 48M/128, 48M/256, 48M/512
- 24M/64, 24M/128, 24M/256, 24M/512

When this issue happens, the support lib function FT4222_SPIMaster_SingleReadWrite may not return, or return FT_FAILED_TO_WRITE_DEVICE.

This issue can be observed with the Revision A also.

Workaround:

There are no known workarounds available. This issue is corrected at revision C.

Package specific:

The effected packages are listed in Table 3-12.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-12 Affected Packages

3.2.4 SPI master throughput is not good enough

Issue:

The FT4222H SPI master throughput is not good enough. The max throughput of SPI quad write is only 28.1Mbps.

Workaround:

N/A. The FT4222H SPI master throughput is improved at revision C.
The max throughput of SPI quad write is improved from 28.1MBps to 53.8Mbps at 80MHz operation clock and 1/2 clock ratio.

SCK Freq. (Hz)		SCK = Operating Clock * the following ratio							
Operating Clock	Max Throughput can be expected	1/2	1/4	1/8	1/16	1/32	1/64	1/128	1/256
80MHz	28.1Mbps*	40M*	20M*	10M	5M	2.5M	1.25M	625K	312.5K
60MHz	20.5Mbps*	30M*	15M	7.5M	3.75M	1.875M	937.5K	468.75K	234.375K
48MHz	16.3Mbps*	24M*	12M	6M	3M	1.5M	750K	375K	187.5K
24MHz	8.0Mbps*	12M*	6M	3M	1.5M	750K	375K	187.5K	93.75K

The rev.B SCK Operating Frequency in SPI Master Mode

SCK Freq. (Hz)		SCK = Operating Clock * the following ratio							
Operating Clock	Max Throughput can be expected	1/2	1/4	1/8	1/16	1/32	1/64	1/128	1/256
80MHz	53.8Mbps*	40M*	20M*	10M	5M	2.5M	1.25M	625K	312.5K
60MHz	39.7Mbps*	30M*	15M	7.5M	3.75M	1.875M	937.5K	468.75K	234.375K
48MHz	31.5Mbps*	24M*	12M	6M	3M	1.5M	750K	375K	187.5K
24MHz	15.8Mbps*	12M*	6M	3M	1.5M	750K	375K	187.5K	93.75K

The rev.C SCK Operating Frequency in SPI Master Mode

*The max. throughput can be expected under the condition of quad mode transfers with a high operating frequency on SCK. It also depends on the USB bus transfer condition. For example, the max throughput that can be expected is up to 28.1Mbps when the operating clock is equal to 80MHz, SCK is set as 20MHz or 40MHz, the data bus is operating in quad mode and the USB bus is operating at hi-speed USB rates with sufficient bandwidth.

Package specific:

The effected packages are listed in Table 3-13.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-13 Affected Packages

3.3 Revision C

3.3.1 Data path is not fully reset when a reset on I²C is executed

Issue:

When the I²C bus encounters errors or works abnormally, users can use the reset APIs to reset the I²C function. When a reset command is received, only the I²C controller is reset. The transferring data may still be left in the related USB pipe. The USB pipe associated to I²C functions should also be reset as the initial status for the next transfer.

Workaround:

There are no known workarounds available. This issue is corrected at Revision D.

Package specific:

The effected packages are listed in Table 3-14.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-14 Affected Packages

3.3.2 No Response STALL to Get BOS Descriptor defined in USB3.0

Issue:

BOS (Binary device Object Store) descriptor is a newly defined descriptor in the USB3.0 specification. Since the FT4222H is a USB2.0 compliant USB device, getting a BOS descriptor command is not supported. A STALL should be returned, but the FT4222H returns NAKs.

Workaround:

There are no known workarounds available. This issue is corrected at Revision D.

Package specific:

The affected packages are listed in Table 3-15.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-15 Affected Packages

3.3.3 Flash Operating Mode Support

Issue:

When accessing Toshiba flash with SPI Master Quad mode, FT4222H will hang without giving any responses.

FT4222H operates as master with supporting Dual or Quad SPI with three phases as illustrated in **Figure 3.1**.

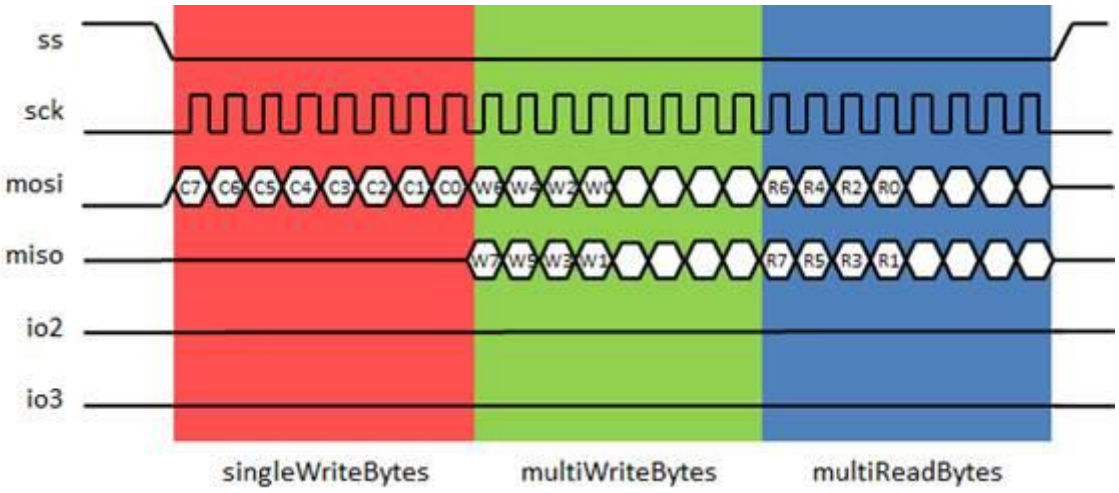


Figure 3.1 Quad SPI with Multi Write Phase

As shown in **Figure 3.1**, 'C', 'W' and 'R', correspond to "Command Phase", "Write Phase" and "Read Phase", where there are information/data in all three phases that are to be exchanged.

Some flash devices operate with single write and multi read protocol but without the multi write phase as illustrated in **Figure 3.2**:

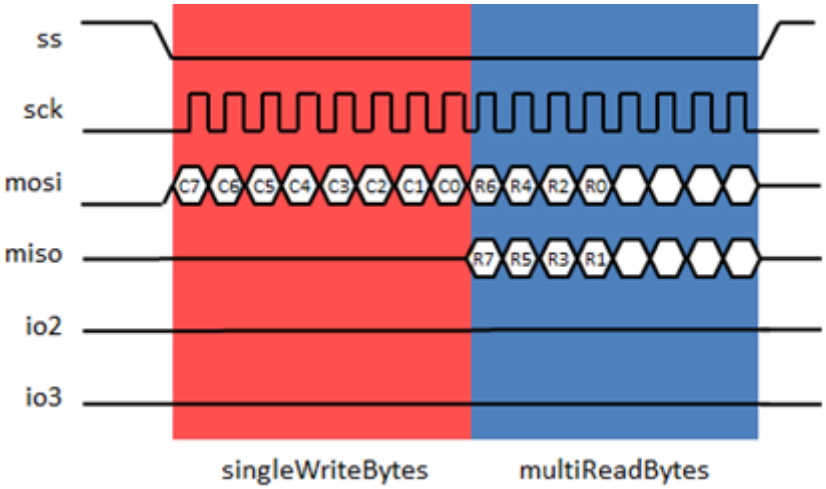


Figure 3.2 Quad SPI without Multi Write Phase

Write Phase has no information/data to transfer. This combination of operating mode is not supported with Revision A, B, and C.

Workaround:

In SPI Master Quad mode, the operation of single write with quad read was not supported. No workaround was provided, this feature is implemented with Revision D.

Package specific:

The affected packages are listed in Table 3-16.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-16 Affected Packages

3.3.4 SPI Slave Data Lost

Issue:

When operating in SPI Slave mode, the FT4222H would occasionally lose data packets.

Workaround:

Verified that the latency Timer configuration was not correct, hence causing no full packet responses, instead all packets responded with short packets. This results in packet drops with D2XX driver.
No workaround was provided, this issue is corrected at Revision D.

Package specific:

The effected packages are listed in Table 3-17.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-17 Affected Packages

3.3.5 The bulk maximum packet size for CNFMODE1/CNFMODE2 is not 512 bytes

Issue:

The USB defines the allowable maximum bulk packet size to be only 512 bytes for high-speed. However, the maximum bulk packet size of CNFMODE1 and CNFMODE2 is 256 bytes.

Workaround:

N/A. This issue is corrected at revision D.
In CNFMODE1 and CNFMODE2, 4 USB interfaces are enabled. In FT4222H Rev D, the buffer operation for IN and OUT pipes is changed to a 512 bytes single buffer instead of 256 bytes dual banks to match the maximum packet size defined for a bulk transfer in a USB high-speed device. In CNFMODE0, the buffer operation for GPIOs is also changed to 512 bytes double buffers.

Max. Packet Size and No. of Buffer Bank	FT4222H Rev D			
	CNFMODE0	CNFMODE1	CNFMODE2	CNFMODE3
Interface A	Data Stream 512 Bytes Dual Banks	Data Stream 512 Bytes Single Bank	Data Stream 512 Bytes Single Bank	Data Stream 512 Bytes Dual Banks
Interface B	For GPIOs 512 Bytes Dual Banks	Data Stream 512 Bytes Single Bank	Data Stream 512 Bytes Single Bank	Disable
Interface C	Disable	Data Stream 512 Bytes Single Bank	Data Stream 512 Bytes Single Bank	Disable
Interface D	Disable	For GPIOs 512 Bytes Single Bank	Data Stream 512 Bytes Single Bank	Disable

Endpoint Maximum Packet Size and Buffer Configuration in Rev D

Max. Packet Size and No. of Buffer Bank	FT4222H Rev A/B/C			
	CNFMODE0	CNFMODE1	CNFMODE2	CNFMODE3
Interface A	Data Stream 512 Bytes Dual Banks	Data Stream 256 Bytes Dual Bank	Data Stream 256 Bytes Dual Bank	512 Bytes Dual Banks
Interface B	For GPIOs 8 Bytes Dual Banks	Data Stream 256 Bytes Dual Bank	Data Stream 256 Bytes Dual Bank	Disable
Interface C	Disable	Data Stream 256 Bytes Dual Bank	Data Stream 256 Bytes Dual Bank	Disable
Interface D	Disable	For GPIOs 8 Bytes Dual Banks	Data Stream 256 Bytes Dual Bank	Disable

Endpoint Maximum Packet Size and Buffer Configuration in Rev A,B,C

The following table shows the maximum throughput that can be expected within each configuration mode. The calculations assume the FT4222H is operating with a high operating frequency on SCK and the USB bus has enumerated the device as a high speed with sufficient bandwidth. For example, the max throughput that can be expected is up to 52.8Mbps when the operating clock is equal to 80MHz, SCK is set as 40MHz, only 1 data stream interface is enabled not 3 or 4, the data bus is operating in quad mode and the USB bus is operating at hi-speed USB rates with sufficient bandwidth.

Throughput (Unit : Mbps)		FT4222H Rev D			
		CNFMODE0	CNFMODE1	CNFMODE2	CNFMODE3
QuadSPI Master	Single Mode (1 bit)	27.8 (Write/Read)	22.8 (Write/Read)	23.0 (Write/Read)	27.3 (Write/Read)
	Dual Mode (2 bit)	39.7(Write) 42.7(Read)	31.3(Write) 38.1(Read)	31.4(Write) 38.7(Read)	39.6(Write) 42.5(Read)
	Quad Mode (4 bit)	53.3(Write) 48.5(Read)	41.6(Write) 42.1(Read)	41.9(Write) 42.8(Read)	52.8(Write) 49.7(Read)

Maximum throughput for QuadSPI Master Controller in Rev. D

As a consequence of the modification described in Section 1.1, USB compliance for the FT4222H IC was re-tested and a new USB-IF Test-ID distributed in March 2018.

TID for FT4222H IC Rev D: 40001830
TID for FT4222H IC Rev A: 40001599

TID for UMFT4222EV with FT4222HQ-D: 10007740
TID for UMFT4222EV with FT4222HQ-A: 10007262

Package specific:

The effected packages are listed in Table 3-18.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-18 Affected Packages

3.3.6 Detectable frequency on GPIOs is not sufficient for most applications

Issue:

The maximum detectable frequency on GPIOs is 4KHz which should be enhanced.

Workaround:

N/A. This issue is improved at revision D.
Revision D add a higher detection mechanism to increase the detectable frequency on GPIOs. The following table shows the maximum frequency that can be detected on GPIOs. Note that this new higher detection function is disabled by default and can be enabled via APIs defined in LibFT4222.

Max. Freq. can be detected	GPIO			GPIO3 as Interrupt input source		
Operating clock Frequency	With operating SPI pipe	With operating I2C pipe	Only GPIO pipe operates	With operating SPI pipe	With operating I2C pipe	Only GPIO pipe operates
60MHz	< 10KHz	< 2KHz	< 20KHz	< 20KHz	< 4KHz	< 40KHz
80MHz	< 10KHz	< 4KHz	< 30KHz	< 60KHz	< 8KHz	< 70KHz

Detectable Frequency on GPIOs

Package specific:

The effected packages are listed in Table 3-19.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-19 Affected Packages

3.4 Revision D

3.4.1 SPI Slave may lose data when transfer frequency is set to 20MHz

Issue:

When operating in SPI Slave mode and receiving large data (>10Kbytes), data loss may occur if the transfer frequency is 20MHz.

Workaround:

Downgrade the transfer frequency to 5MHz.

Package specific:

The effected packages are listed in Table 3-20.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-20 Affected Packages

3.4.2 I²C data is corrupt when FT4222_I2CMaster_GetStatus is being called

Issue:

An error would happen when I²C master is writing data and FT4222_I2CMaster_GetStatus is being called at the same time.

Workaround:

Call FT4222_I2CMaster_GetStatus after the end of I²C transmission.

Package specific:

The effected packages are listed in Table 3-21.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-21 Affected Packages

3.4.3 Bus Error condition in USB Device Controller

Introduction:

The USB Device Controller writes past the range of the data buffer when a babble error occurs.
A babble error occurs when USB device receives more data than the maximum packet size.

Issue:

If the data packet comes with the correct CRC16, the USB Device Controller accepts it and responds with ACK. It then writes the data over the address boundary of the data buffer for the endpoint.

If the data packet comes with the incorrect CRC16, the USB Device Controller discards it and times out. However, it still writes the data over the address boundary of the data buffer for the endpoint.

Workaround:

Currently, there is no workaround for this issue.

Package specific:

The effected packages are listed in Table 3-22.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-22 Affected Packages

3.4.4 MISO send data at wrong timing in SPI Slave Mode

Introduction:

The data output of FT4222 in SPI slave mode does not comply with the well-known SPI spec. MISO change the data polarity at wrong timing.

Issue:

The SPI spec defines CPHA and CPOL to configure the clock polarity and phase with respect to the data. CPHA decides when MISO pin changes the IO polarity.

Take CPHA = 0 as example.
The SPI SPEC describe:
1. The first data bit is outputted immediately when CS activates.
2. The MISO changes IO polarity on trailing phase to make sure the data can be got the correct data on leading phase.

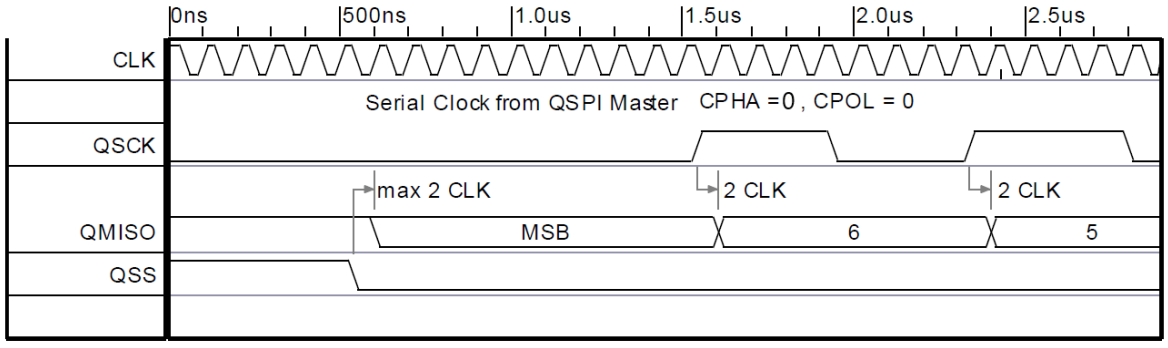
The issue happens on item 2.
As shown in the diagram below, FT4222 changes the IO polarity after 2 CLK (25ns) after leading phase happens. This issue may not cause problem when SPI Master is a hardware decode IP. However, if the SPI Master is a MCU base design with low sampling rate, the master might retrieve the wrong data.

Workaround:

Currently, there is no workaround for this issue. When leading phase happens, the SPI master needs to retrieve data as soon as possible. If the process cannot finish in 25ns, the data will be overwritten by the next data bit.

Package specific:

The effected packages are listed in Table 3-23.



Note: 2CLK = 25ns

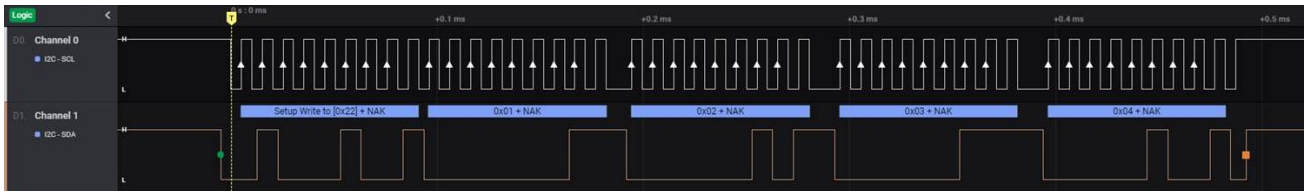
Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-23 Affected Packages

3.4.5 I2C master will continue to transmit data even if it receives a NAK after sending the address byte

Issue:

As shown in the diagram below, the I2C master of FT4222 executes the request from the host that writes four bytes 0x01, 0x02, 0x03, 0x04 to the I2C salve with address 0x22. Even though FT4222 gets NAK following the address byte, FT4222 still sends out all data. In theory, the data transmission should be terminated upon receiving an NAK.



Workaround:

N/A.

Package specific:

The effected packages are listed in Table 3-24.

Package	Applicable (Yes/No)
FT4222HQ	Yes

Table 3-24 Affected Packages

4 FT4222H Series Package Markings

The FT4222H is supplied in a RoHS compliant leadless VQFN-32 package. The package is lead (Pb) free and uses a 'green' compound. The package is fully compliant with European Union directive 2002/95/EC. An example of the markings on the package is shown in Figure 4.1 below.

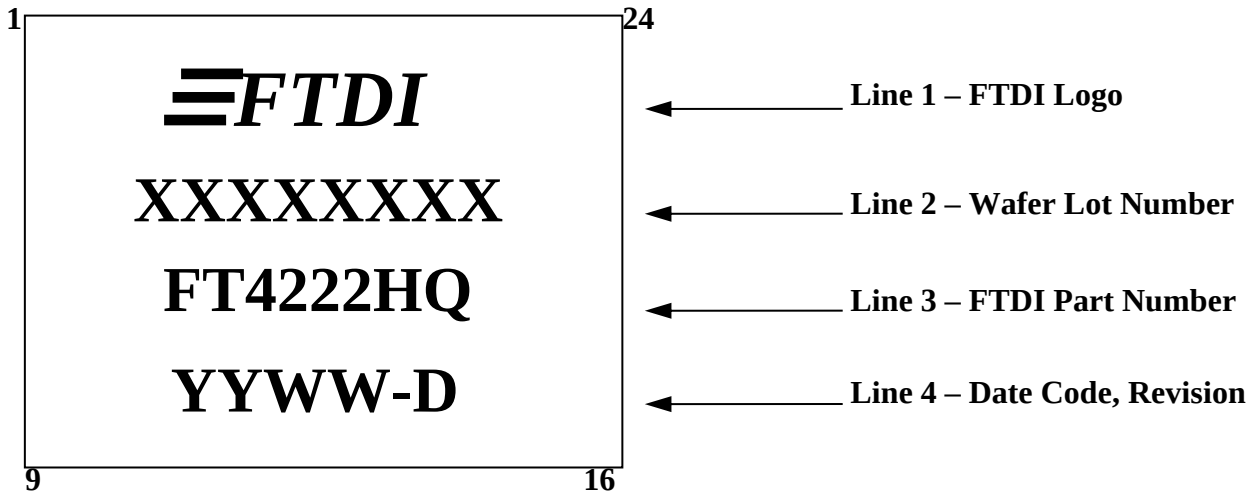


Figure 4.1 VQFN-32 Package Markings

The date code format is **YYWW** where WW = 2-digit week number, YY = 2-digit year number. This is followed by the revision number.

The code **XXXXXXXX** is the manufacturing LOT code.

5 Contact Information

Head Office – Glasgow, UK

Future Technology Devices International Limited (UK)
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 8797 1330
Fax: +886 (0) 2 8751 9737

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

NA

Acronyms and Abbreviations

Terms	Description
CPU	Central Processing Unit
GPIO	General Purpose Input/output
I2C	Inter-Integrated Circuit
MISO	Master In Slave Out
MOSI	Master Out Slave In
PC	Personal Computer
SS	Slave Select
SCK	Serial Clock
SPI	Serial Peripheral Interface
USB	Universal Serial Bus
VQFN	Very Thin Quad Flat Non-Leaded Package

Appendix B – List of Tables and Figures

List of Tables

List of Figures

Appendix C – Revision History

Document Title: TN_161 FT4222H Errata Technical Note
Document Reference No.: FT_001198
Clearance No.: FTDI# 455
Product Page: http://ftdichip.com/products/ft4222h/
Document Feedback: Send Feedback

Revision	Changes	Date
1.0	Initial Release	31-08-2015
1.1	Updated with custom PID issue.	17-05-2016
1.2	Updated with rev C fixes	18-10-2016
1.3	Updated with rev D fixes	28-03-2018
1.4	Updated for IO_ERROR occurs	15-04-2020
1.5	Updated for Section 3.4.1	21-03-2022
1.6	Added section 3.6	12-10-2023
1.7	Added sections 3.2.4, 3.3.5, 3.3.6 as TN_170 FT4222H Rev.C Technical Note and TN_175_FT4222H Rev. D Technical Note have been migrated into this document. Added new sections 3.4.5. Modified Table 2-1 to match naming of all sub-sections.	21-11-2024