APPLICATION NOTE

FocalTech

FT3168

The FT3168 is single-chip capacitive touch panel controllers with built-in enhanced Micro-controller unit(MCU).It provides the benefits of full screen common mode scan technology, fast response time and high level of accuracy.

Preliminary

11.30, 2020

Version:1.0

Table of Contents

Terminology
CTP – Capacitive touch panel
CTPM – Capacitive touch panel module

1. CTPM Interface to Host

Figure 1-1 shows how CTPM communicates with host device. I^2C interface supported by FT3168 that is two-wire serial bus consisting of data line SDA and clock line SCL, used for serial data transferring between host and slave device.

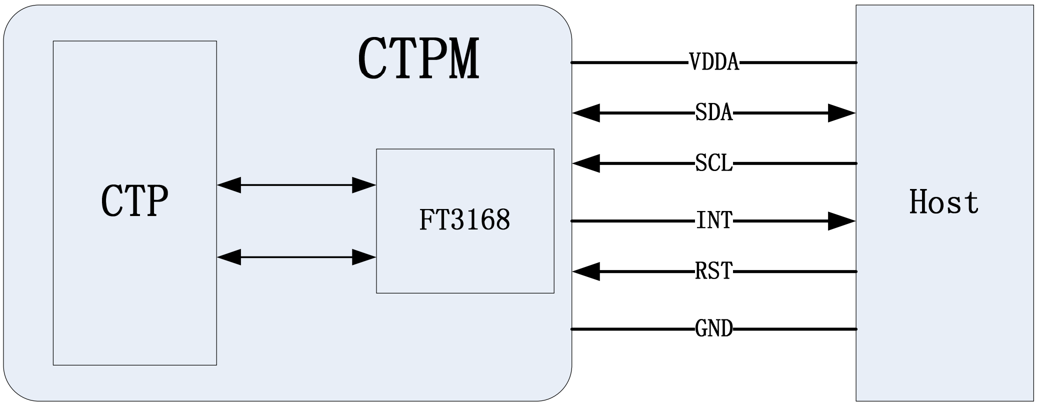


Figure1-1CTPMandHostconnection

The INT port and RST port from the control interface. The INT port is controlled by FT3168, it will send out an interrupt request signal to the host when there is a valid touch on CTPM .Host can send the reset signal to CTPM via RST port to reset the FT3168 if needed. The Power Supply voltage of CTPM ranges from 2.8V to3.6V. For details, please refer to Table1-1 Description for CTPM and Host interface.

Port Name	Description
VDDA	CTPM power supply, ranges from 2.8V to 3.6V.
SDA	I^2C data input and output.
SCL	I^2C clock input.
INT	The interrupt request signal from CTPM to Host.
RST	There set signal from host to CTPM active low, and the low pulse width should be more than 1ms.
GND	Power ground.

1.1 Power ON/ Reset Sequence

The GPIO such as INT and I2C are advised to be low before powering on. Reset should be pulled down to be low before powering on. INT signal will be sent to the host after initializing all parameters and then start to report points to the host. If power down, the voltage of supply must be below 0.3V and Trst is more than 5ms.

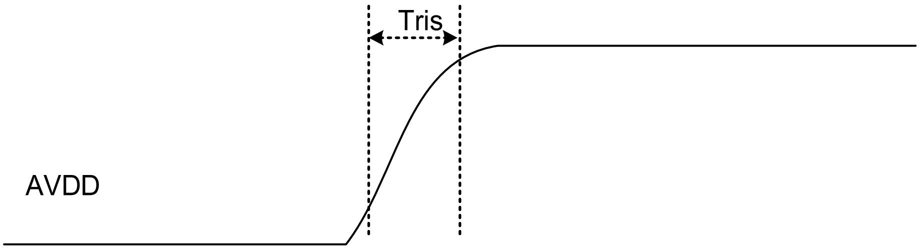


Figure 3-1 Power on time

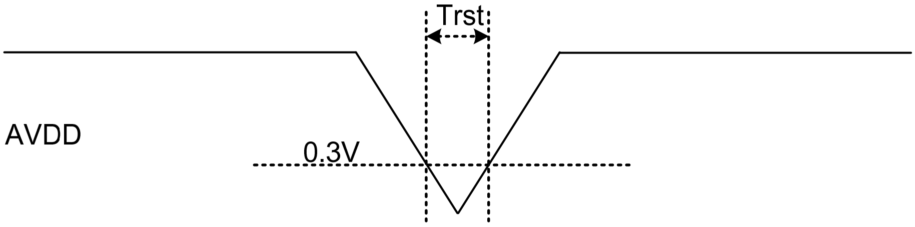


Figure 3-2 Power down requirement

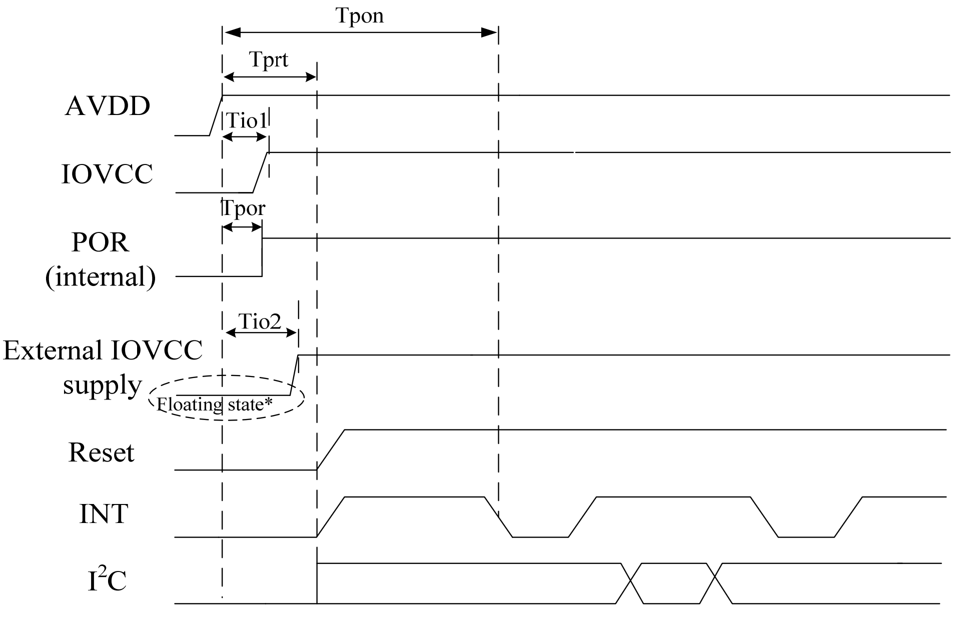


Figure 3-3 Power on Sequence (for external IOVCC supply exist)

Notes*: external IOVCC supply should be in a floating state when it is not on to avoid unexpected leakage.

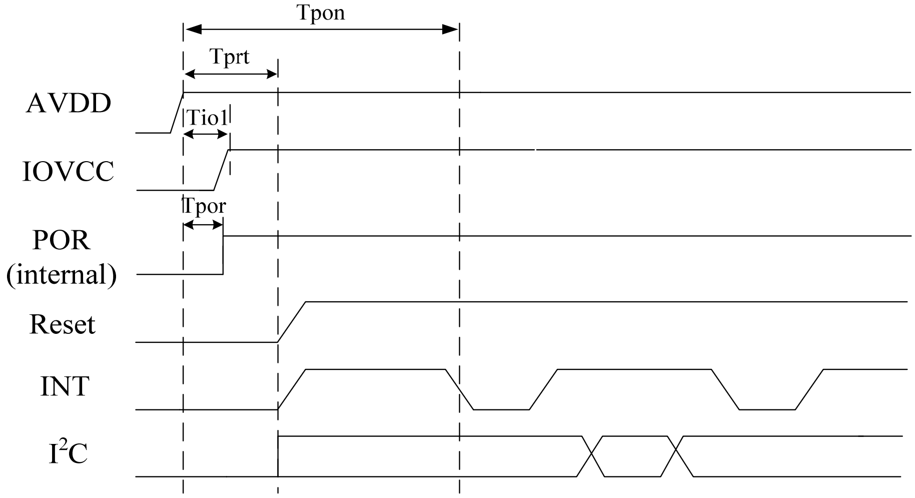


Figure 3-4 Power on Sequence (for external IOVCC supply not exist)

Reset time must be enough to guarantee reliable reset, the time of starting to report point after resetting approach to the time of starting to report point after powering on.

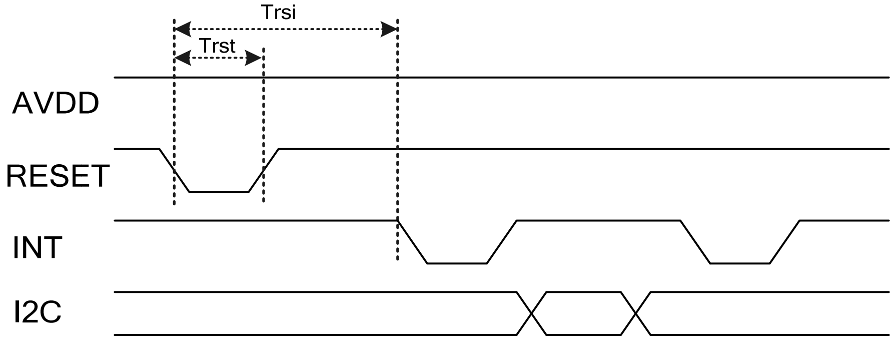


Figure 3-5 Reset Sequence

Table 3-5 Power on/Reset/Wake Sequence Parameters

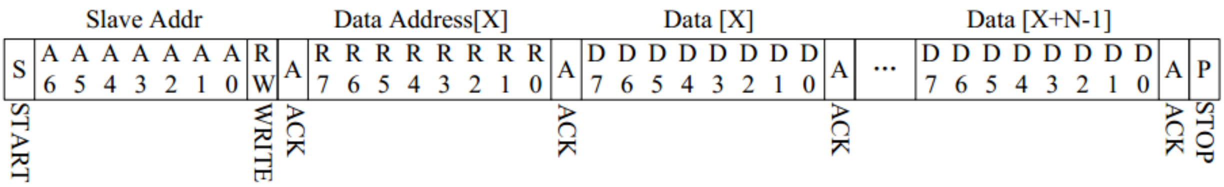
Parameter	Description	Min	Max	Units
Tris	Rise time from 0.1VDD to 0.9VDD	-	3	ms
Tpor	Time of internal POR ready after AVDD power on	0.3	-	ms
Tio1	Time of IOVCC rising after AVDD power on	0.5	-	ms
Tio2	Time of external IOVCC supply after AVDD power on	1.0	-	ms
Tpon	Time of starting to report point after powering on	50	-	ms
Tprt*	Time of RESETB being low after power on	0.5	5	ms
Trsi	Time of starting to report point after resetting	50	-	ms
Trst*	Reset time	0.5	5	ms

Notes*: reset time should not to be too long, because when reset is low, all the control logic would be fixed to be the default value, and the system state is fixed to be a default state.

1.2 I2C Read/Write Interface description

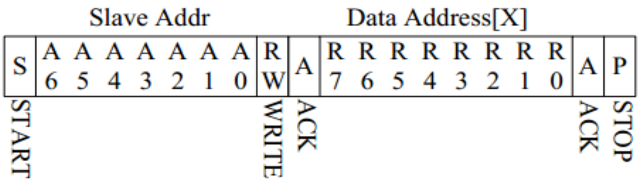
It is important to note that the SDA and SCL must connect with a pull-high resistor respectively before you read/write I^{2}C data.

1.2.1 Host write data to slave

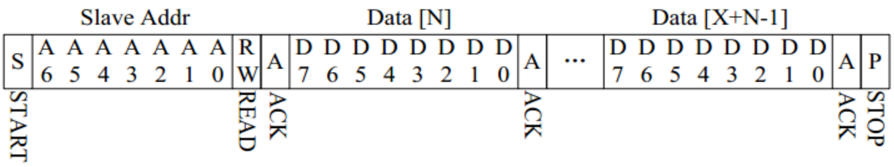


1.2.2 Host read data from slave

Step1: write data address



Step2: read data



1.3 Interrupt signal from CTPM to Host

As for standard CTPM, host needs to use both interrupt signal and I2C interface to get the touch data. CTPM will output an interrupt request signal to the host when there is a valid touch. Then host can get the touch data via I2C interface. If there is no valid touch detected, the INT will output high level, and the host does not need to read the touch data. There are two kinds of method to use interrupt: interrupt trigger and interrupt polling.

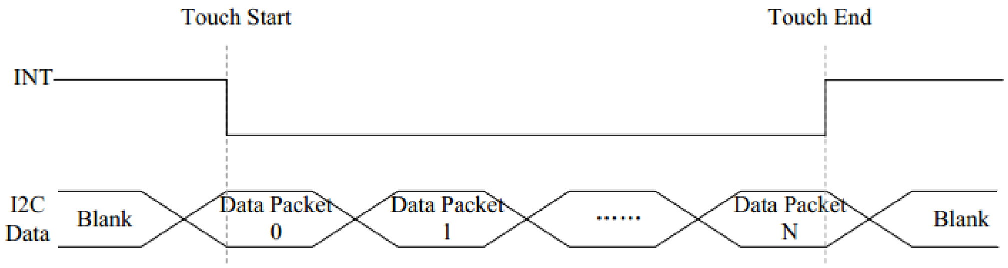


Figure 1.2-1 Interrupt polling mode

As for interrupt polling mode, INT will always be pulled to low level when there is a valid touch point, and be high level when a touch finished.

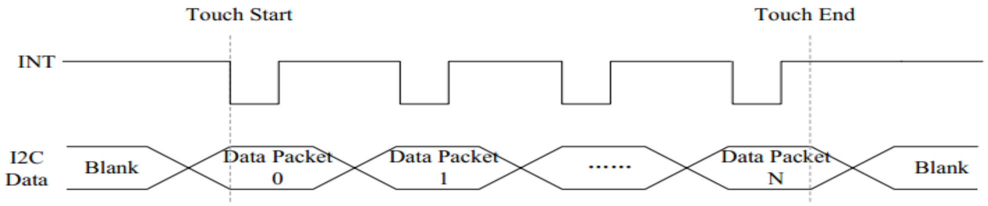


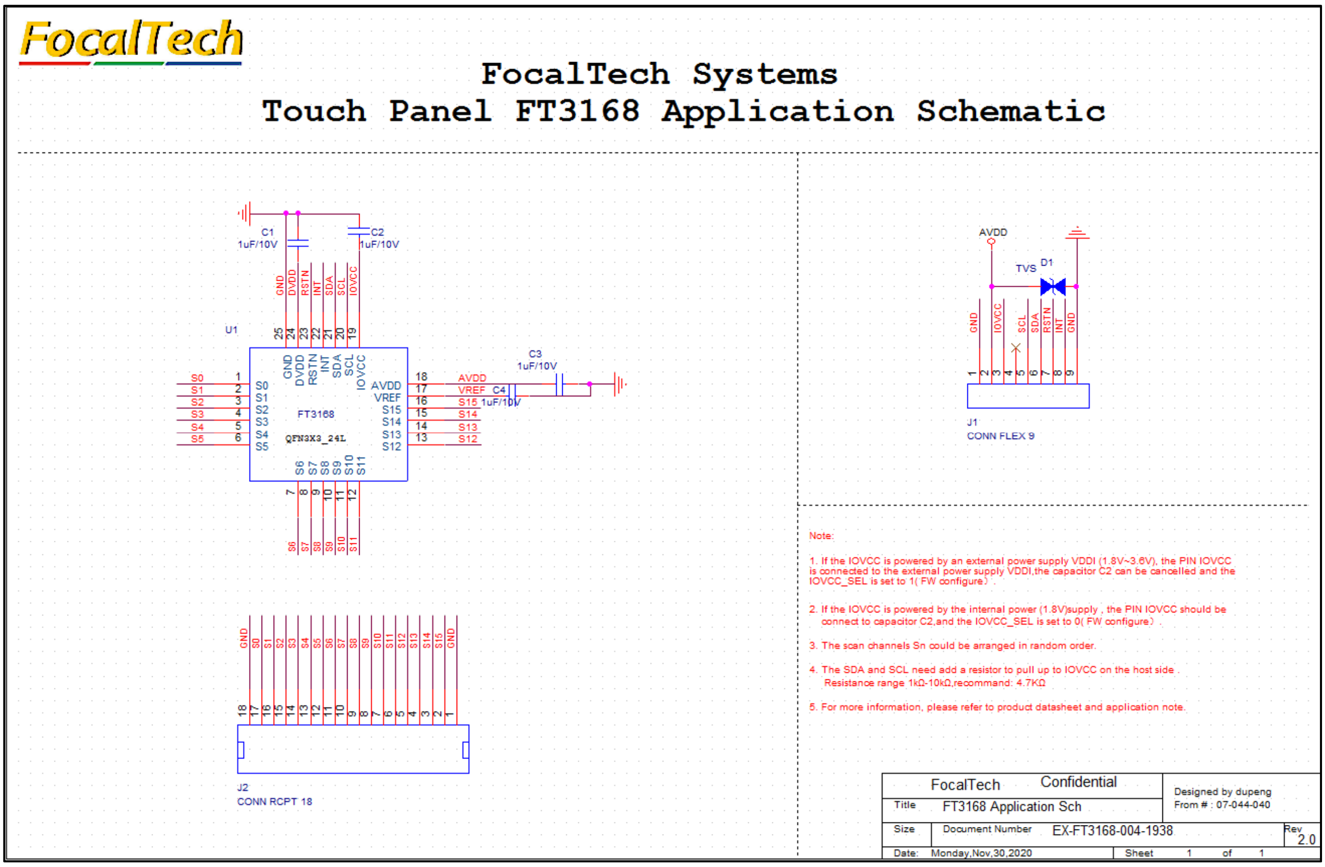
Figure 1.2-2 Interrupt trigger mode

While for interrupt trigger mode, INT signal will be set to low if there is a touch detected. But whenever an update of valid touch data, CTPM will produce a valid pulse on INT port for INT signal, and host can read the touch data periodically according to the frequency of this pulse. In this mode, the pulse frequency is the touch data updating rate.

1.4 Reset signal from Host to CTPM.

Host can send the reset signal via RST port to reset FT3168. The reset signal should not be set to low while in normal working mode. The RST port can also be used to active the CTPM in hibernate mode. Note that the reset pulse width should be more than 1ms.

2. Application Circuit



Note:

1. If the IOVCC is powered by an external power supply VDDI (1.8V~3.6V), the PIN IOVCC is connected to the external power supply VDDI,the capacitor C2 can be cancelled and the IOVCC_SEL is set to 1(FW configure）.

2. If the IOVCC is powered by the internal power (1.8V)supply , the PIN IOVCC should be connect to capacitor C2,and the IOVCC_SEL is set to 0(FW configure）.

3. The scan channels Sn could be arranged in random order.

4. The SDA and SCL need add a resistor to pull up to IOVCC on the host side . Resistance range 1kΩ-10kΩ,recommand: 4.7KΩ

5. For more information, please refer to product datasheet and application note.

FocalTech Confidential
Title FT3168 Application Sch
Designed by dupeng
From # : 07-044-040
Size Document Number EX-FT3168-004-1938 Rev 2.0
Date: Monday,Nov,30,2020 Sheet 1 of 1

Note:

1. If GPIO supply voltage is set to VCC (2.8V~3.6V), IOVCC pin can be connected to VCC.
2. If GPIO supply voltage is 1.8V, IOVCC pin should be floating or connect to external I/O1.8V.
3. The SDA and SCL need add a resistor to pull up to IOVCC on the host side .Resistance range 1kΩ-10kΩ,recommand: 4.7KΩ
4. The INT don’t need add a resistor to pull up to IOVCC on the host side .But if the HOST need to add the resistance, the resistance should be range 4.7kΩ-10kΩ

3. BOM List

Item	名称	Quantity	Reference	Part	PCB Footprint	Note
1	电容	4	C1,C2,C3,C4	1uF/10V	C0402/C0201	X7R
2	IC	1	U1	FT3168	QFN3X3	
3	TVS 管	1	D1	TVS	D0402/D0201	RS060306T RS060302K RS1E008T

4. Power Supply Configuration

	Operating Voltage(V)	Ripple&Noise（mV）
VDDA	2.8~ 3.6	<100
IOVCC	1.8~ 3.6	<100

VCC Current	Type
Normal operation mode	1.5mA
Monitor mode	40uA
Sleep mode	10uA

Brief summary of the FT3618 application features.

Channel	16 CH
Panel Size	＜1.4inch
Touch points	2
Interface	I^2C
Report rate	100HZ

Note: The above information is only for hardware design. For chip information, please refer to Datasheet.

5. CTPM Register Mapping

This chapter describes the standard CTPM communication registers in address order for working mode.

5.1 Working Mode

The CTPM is fully functional as a touch screen controller in working mode. The access address to read and write is just logical address which is not enforced by hardware. Here is the working mode register map.

Register Map [Working Mode]

ADDR	RW	Name	b7	b6	b5	b4	b3	b2	b1	b0
0x00	**RW**	Mode_Switch		Device Mode[2:0]						
0x01	**RO**	Guesture	Gesture ID [7:0]							
0x02	**RO**	Cur Point	Number of touch points[7:0]							
0x03	**RO**	**TOUCH1_XH**	**1st Event Flag**				**1st Touch X Position[11:8]**			
0x04	**RO**	**TOUCH1_XL**	**1st Touch X Position[7:0]**							
0x05	**RO**	**TOUCH1_YH**	**1st Touch ID[3:0]**				**1st Touch Y Position[11:8]**			
0x06	**RO**	**TOUCH1_YL**	**1st Touch Y Position[7:0]**							
0x07	**RO**	**TOUCH1_WEIGHT**	**1st Touch Weight[7:0]**							
0x08	**RO**	**TOUCH1_MISC**	**1st Touch Area[3:0]**							
0x09	**RO**	**TOUCH2_XH**	**2nd Event Flag**				**2nd Touch X Position[11:8]**			
0x0A	**RO**	**TOUCH2_XL**	**2nd Touch X Position[7:0]**							
0x0B	**RO**	**TOUCH2_YH**	**2nd Touch ID[3:0]**				**2nd Touch Y Position[11:8]**			
0x0C	**RO**	**TOUCH2_YL**	**2nd Touch Y Position[7:0]**							
0x0D	**RO**	**TOUCH2_WEIGHT**	**2nd Touch Weight[7:0]**							
0x0E	**RO**	**TOUCH2_MISC**	**2nd Touch Area[3:0]**							

5.2 DEVICE_MODE

This is the device mode register, which is configured to determine the current mode of the chip.

Address	Bit Address	Register Name	Description
0x00	6:4	[2:0]Device Mode	000b: Working Mode 100b: Test Mode

5.3 GEST_ID

This register describes the gesture of a valid touch.

Address	Bit Address	Register Name	Description

0x01	7:0	Gesture ID[7:0]	Gesture ID 0x10: Move Up 0x14: Move Right 0x18: Move Down 0x1C: Move Left 0x48: Zoom In 0x49: Zoom Out 0x00: No Gesture

5.4 TD_STATUS

This register is the Touch Data status register.

Address	Bit Address	Register Name	Description
0x02	7:0	Number of touch points [7:0]	The detected point number, max. 10

5.5 Pn_XH (n:1-2)

This register describes MSB of the X coordinate of the nth touch point and the corresponding event flag.

Address	Bit Address	Register Name	Description
0x03 0x09	7:6	Event Flag	00b: Press Down 01b: Lift Up 10b: Contact 11b: No event
	5:4	Reserved	
	3:0	Touch X Position [11:8]	MSB of Touch X Position in pixels

5.6 Pn_XL (n:1-2)

This register describes LSB of the X coordinate of the nth touch point.

Address	Bit Address	Register Name	Description
0x04 0x0A	7:0	Touch X Position [7:0]	LSB of the Touch X Position in pixels

5.7 Pn_YH (n:1-2)

This register describes MSB of the Y coordinate of the nth touch point and corresponding touch ID.

Address	Bit Address	Register Name	Description
0x05 0x0B	7:4	Touch ID[3:0]	Touch ID of Touch Point, this value is 0x0F when the ID is invalid
	3:0	Touch Y Position [11:8]	MSB of Touch Y Position in pixels

5.8 Pn_YL (n:1-2)

This register describes LSB of the Y coordinate of the nth touch point.

Address	Bit Address	Register Name	Description
0x06 0x0C	7:0	Touch Y Position [7:0]	LSB of the Touch Y Position in pixels

5.9 Pn_WEIGHT (n:1-2)

This register describes weight of the nth touch point.

Address	Bit Address	Register Name	Description
0x07 0x0D	7:0	Touch Weight[7:0]	Touch pressure value

5.10 Pn_MISC (n:1-2)

This register describes the miscellaneous information of the nth touch point.

Address	Bit Address	Register Name	Description
0x08 0x0E	7:4	Touch Area[3:0]	Touch area value
	3:0	Reserved	

6. Communication Between Host and CTPM

6.1 Communication Contents

The data Host received from the CTPM through I2C interface are different depend on the configuration in Device Mode Register of the CTPM. Please refer to Section 2---CTPM Register Mapping.

6.2 I2C Example Code

The code is only for reference, if you want to learn more, please contact our FAE staff.

```
////////////////////////////////////////////////////////////////
// I2C write bytes to device.
// Arguments: ucSlaveAdr - slave address
//                        ucSubAdr - sub address
//                        pBuf - pointer of buffer
//                        ucBufLen - length of buffer
////////////////////////////////////////////////////////////////
void i2cBurstWriteBytes(BYTE ucSlaveAdr, BYTE ucSubAdr, BYTE *pBuf, BYTE ucBufLen)
{
    BYTE ucDummy; // loop dummy
    ucDummy = I2C_ACCESS_DUMMY_TIME;
    while(ucDummy--)
    {
        if (i2c_AccessStart(ucSlaveAdr, I2C_WRITE) == FALSE)
            continue;
        if (i2c_SendByte(ucSubAdr) == I2C_NON_ACKNOWLEDGE) // check non-acknowledge
            continue;
        while(ucBufLen--) // loop of writting data
        {
            i2c_SendByte(*pBuf); // send byte
            pBuf++; // next byte pointer
        } // while
        break;
    } // while
i2c_Stop();
}

////////////////////////////////////////////////////////////////
// I2C read bytes from device.
//
```

```
// Arguments: ucSlaveAdr - slave address
//                                   ucSubAdr - sub address
//                                   pBuf - pointer of buffer
//                                   ucBufLen - length of buffer
/////////////////////////////////////////////////////////////////
void i2cBurstReadBytes(BYTE ucSlaveAdr, BYTE ucSubAdr, BYTE *pBuf, BYTE ucBufLen)
{
    BYTE ucDummy; // loop dummy

    ucDummy = I2C_ACCESS_DUMMY_TIME;
    while(ucDummy--)
    {
        if (i2c_AccessStart(ucSlaveAdr, I2C_WRITE) == FALSE)
            continue;
        if (i2c_SendByte(ucSubAdr) == I2C_NON_ACKNOWLEDGE) // check non-acknowledge
            continue;
        if (i2c_AccessStart(ucSlaveAdr, I2C_READ) == FALSE)
            continue;
        while(ucBufLen--) // loop to burst read
        {
            *pBuf = i2c_ReceiveByte(ucBufLen); // receive byte
            pBuf++; // next byte pointer
        } // while
        break;
    } // while
```

7. DISCLAIMER

The information appearing in this publication is believed to be accurate.

Integrated circuits sold by FocalTech Systems are covered by the warranty and patent indemnification provisions stipulated in the terms of sale only. FocalTech Systems makes no warranty, express, statutory implied or by description regarding the information in this publication or regarding the freedom of the described chip(s) from patent infringement. FURTHERMORE, FocalTech Systems MAKES NO WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PURPOSE. FocalTech Systems reserves the right to halt production or alter the specifications and prices at any time without notice. Accordingly, the reader is cautioned to verify that the datasheets and other information in this publication are current before placing orders. Products described herein are intended for use in normal commercial applications. Applications involving unusual environmental or reliability requirements, e.g. military equipment or medical life support equipment, are specifically not recommended without additional processing by FocalTech Systems for such applications. Please note that application circuits illustrated in this document are for reference purposes only.

8. REVISION HISTORY

Date	Revision #	Description	Page	Auditor
11,30, 2020	1.0	Original.	15	Du Peng