ICNT86Series Multi-touch Capacitive Screen Solutions

Documentation version: V1.0

Release date: 2015/03/16

content

I. Overview

ICNT86series is built-inMCUThe capacitance detection chip of the mutual capacitance touch screen supports multi-touch function, combined with mutual capacitance or self capacitance touch screen, it can form a capacitive touch screen module (Capacitive Touch Panel Module, CTPM), the user interface is friendly and easy to input. This document describes how to apply in the systemICNT86series chips andCTPM, and introduceHostHow to do itCTPMfor multi-touch.

1.1Function introduction

ICNT86The series of chips are single-chip solutions, which mainly include the following functions:

A.true multi-touch

support real5Touch, can support10touch

b.supportDITO, SITO,single-layer multi-point,OGS,CompleteITO

c.Anti-interference ability

Built-in high-performance anti-jamming module, effectively suppressRF,LCDinterference; support automatic frequency hopping

D.Support passive pen, glove touch, standby gesture and other functions

E.point rate

Recommended reporting rate80Hz, High reach200Hz

F.Power consumption

It is related to the touch screen body, reporting rate, scanning frequency, number of contacts and other factors, see the table1.

G.power supply

voltage 2.8V ~ 3.3V

I/OVoltage 1.8V/VCC(software configurable)

H.Application field

Mobile communication terminal, portable multimedia player

Home appliances, home entertainment equipment

i. Mobile Internet Device

Portable/Tablet

GPS/Digital camera/game console/navigation system/information kiosk, etc.

J. interface standard

I_2CInterface: standard interface, high speed up to400kpscompatibleSlave

K.supportI_2CWake up with standby gesture

L.User Programmable Scannable Sequence, LargerADCDynamic Range

M.Good adaptability to environment and touch changes

N.Automatic calibration

Table 1 Current parameters in different modes

Parameter	Description	ICNT8658			ICNT86L8			ICNT8688			ICNT8698			
		Min	Typ	Min	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Unit
IOPR	normal operation mode current		6.7			9.9			15.1			15.9		mA
IMON	Monitor mode current		2.8			3.5			4.9			5.2		mA
ISUS	Suspend mode current		167			167			167			167		µA
ISLP	Sleep mode current		91			91			91			91		µA
IGES	Gesture mode current		700			700			700			700		µA

Notice:

1. suspend modeandsleep modeThe current parameters are all measured at room temperature,VCCThe voltage is3.3V;

2.Configured via software (firmware), can besuspend modeEntersleep mode. The default issuspend mode.

1.2product model

Table 2 ICNT86 flat panel product selection table

Model Name	Channels		Package			Touch Panel Size
	TX	RX	Type	Pin	Size	
ICNT8672	28TX*19RX 29TX*18RX 30TX*17RX		QFN	68	8*8	7″ ~ 9″
ICNT8682	32TX*24RX 33TX*23RX 34TX*22RX 35TX*21RX 36TX*20RX		QFN	68	8*8	7" ~ 10.1"
ICNT8692	42TX*30RX		QFN	88	10*10	8″ ~ 12.1″

Table 3 ICNT86 mobile phone series product selection table

Model Name	Channels		Package			Touch Panel Size
	TX	RX	Type	Pin	Size	
ICNT8636	21TX*13RX 22TX*12RX 23TX*11RX		QFN	48	6*6	4.0" to 5.0"
ICNT8656	23TX*17RX 24TX*16RX 25TX*15RX 26TX*14RX 27TX*13RX		QFN	52	6*6	4.5″～6.0″

Table 4 ICNT86 on-cell series product selection table

Model Name	Channels		Package			Touch Panel Size
	TX	RX	Type	Pin	Size	
ICNT8658	23TX*17RX 24TX*16RX 25TX*15RX 26TX*14RX 27TX*13RX		QFN	52	6*6	4.5"~6.0" (Phone) 7.0" ~ 8.0" (Tablet)
ICNT86L8	15TX*30RX 16TX*29RX 17TX*28RX 18TX*27RX 19TX*26RX		QFN	56	6*6	4.5" ~ 6.5" (Phone) 7.0" ~ 9.0" (Tablet)
ICNT8688	32TX*24RX 33TX*23RX 34TX*22RX 35TX*21RX 36TX*20RX		QFN	68	8*8	7.0" ~ 10.1" (Tablet)
ICNT8698	42TX*30RX		QFN	88	10*10	8.0" ~ 12.1" (Tablet)

2. Functional structure and working mode

2.1functional module

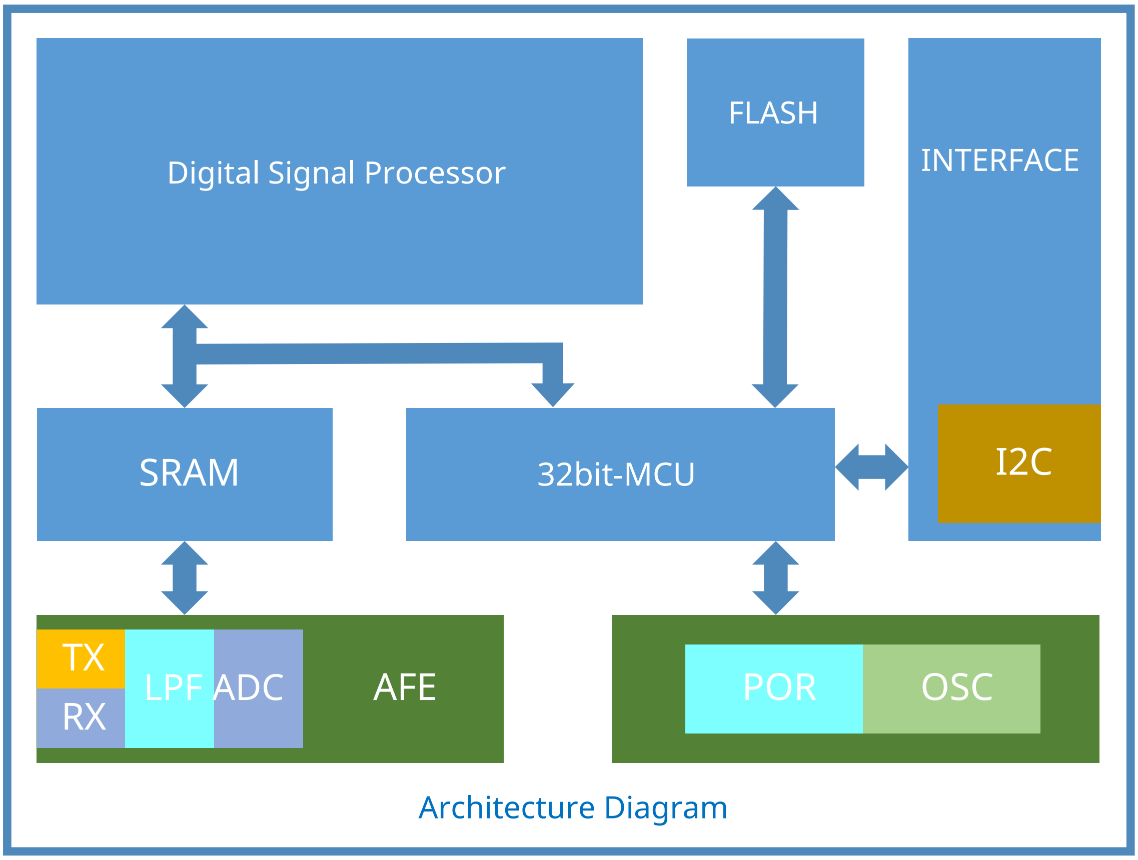

Figure 1 Internal block diagram of ICNT86 series

ICNT86series chip built-in32bit MCUandDSP, which is used to process touch information data, implement complex software algorithms, and coordinate the work of each global module.

SRAMUsed to store touch and algorithm processing information, etc.

AFEcontainADCand other modules to form an analog information channel, which can effectively suppressRF,LCDand other noise introduced to ensure a high signal-to-noise ratio. The module is configurable through a flexibleTX,RXThe port is connected to the screen, and the captured touch information is converted into digital signals and stored inSRAM,forMCUandDSPoperation.

The interface controller module contains400kbps I_2Cinterface. In addition, the module provides management control of ports such as interrupt interface, wake-up interface, and external reset interface.

ICNT86xxbuilt-inFLASH, which is used to store system firmware and key system parameters.

2.2Operating mode

1. Normal Mode:

That is, the normal working mode. In this mode, the chip passesTX,RXContinuously scan the touch information on the screen, and each functional module that scans and processes the touch information is in a normal application state. The default scan reporting rate is80Hz, HostThe reporting rate can be increased or decreased through the port as required.

2. Monitor Mode:

ICWhen no touch action is detected, it can be automatically entered after a certain period of timeMonitormode, the time can be configured by software, the default is1s. In this mode, the chip scans the touch information on the screen at a very low frequency, and the algorithm is simplified to only detect whether there is a touch on the screen to reduce power consumption. When there is a touch on the screen, the chip automatically switches from Monitormode switch toActiveMode, normally detects and processes touch information;

3. Hibernate Mode:

This mode can be entered by command from the master. In this mode, the chip runs with extremely low power consumption, all functional modules are in sleep or power-down state, and only detect and respond to wake-up signals and external reset signals.

4. Gesture Mode: Make a specific gesture when the touch device is in standby mode. After the touch screen recognizes the correct gesture, an interrupt is triggered, and the master controller is notified to let the master controller read the corresponding gesture code, and the master controller can realize the functions required by the customer. The following figure briefly describes the transition between the four working modes:

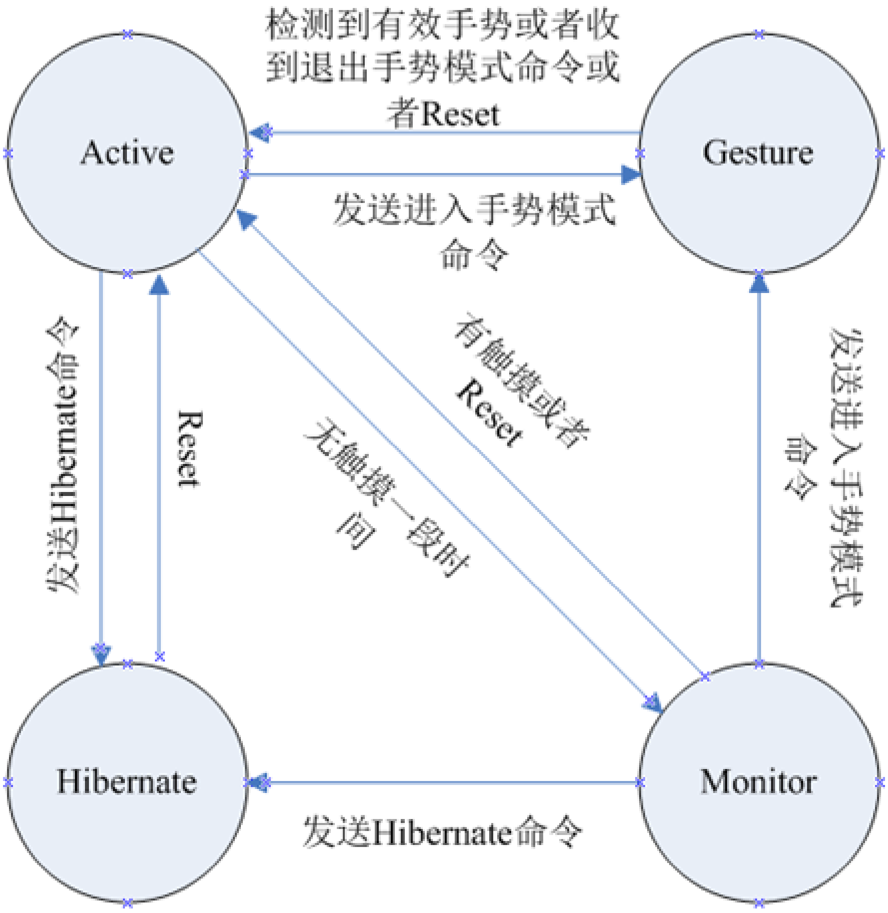

Figure 2 Working mode conversion

2.3Automatic frequency hopping

ICNT86It has good hardware anti-interference characteristics. The noise of each frequency band can be monitored in real time. When it is found that the noise of the current operating frequency exceeds the threshold, it will automatically switch the operating frequency to another clean frequency to avoid interference.

2.3.1Frequency Hopping Mode and Frequency Hopping Parameters

Frequency Hopping Mode:DEFAULT_FREQ_HOP_MODE.

Currently supported frequency hopping modes are:

FREQ_HOP_MODE_OFF: frequency hopping off

FREQ_HOP_MODE_ON: Frequency hopping on, every frequency hopping, updatebase, the recommended mode

FREQ_HOP_MODE_SEEMLESS: Seamless frequency hopping mode, used in the case of screen impedance and small capacitive reactance.

The relevant parameters are:

Table 5 Frequency hopping parameter table

Parameter	Description
DEFAULT_NOISE_BIT_FREQ	used for noise detectionBitFreq, without modification. Currently using the same value as a normal scan.
DEFAULT_FREQ_HOP_START_CYCLE_N UM	start of frequency hopping rangeCycleNum, From this parameter, the starting frequency of frequency hopping can be deduced: DEFAULT_FREQ_HOP_START_CYCLE_NUM* DEFAULT_BIT_FREQ
DEFAULT_FREQ_HOP_STOP_CYCLE_NUM	Termination of frequency hopping rangeCycleNum, From this parameter, the frequency hopping termination frequency can be deduced: DEFAULT_FREQ_HOP_START_CYCLE_NUM * DEFAULT_BIT_FREQ
DEFAULT_FREQ_HOP_NORMOLIZE	Normalized parameter, which is automatically calculated by the tool provided by the host computer Figure it out.
DEFAULT_FREQ_HOP_CYCLE_NUM	Corresponding to the center frequency of the bandCycleNum, This parameter is automatically calculated by the tool provided by the host computer.
DEFAULT_FREQ_HOP_NOISE_TH	Frequency hopping threshold, the system can be observed through the host computer tool Noise level, set reasonable according to the noise level of the actual system threshold.
DEFAULT_FREQ_HOP_CONFIRM_TIMES	Noise repeated confirmation times, the more confirmation times, the The closer the noise situation is to the actual noise situation, the lag of the frequency hopping time approx.

2.3.2Frequency Hopping Configuration Method

1)Modify the frequency hopping parameters toFREQ_HOP_MODE_OFF, select the corresponding DEFAULT_BIT_FREQ, DEFAULT_F1_CYCLE_NUM(abbreviated asDF1CN) to debug the best touch performance.

2)Modify the frequency hopping parameters toFREQ_HOP_MODE_ON, fill in the following two parameters:

DEFAULT_FREQ_HOP_START_CYCLE_NUM(abbreviated asDFHCN_STA),
DEFAULT_FREQ_HOP_STOP_CYCLE_NUM(abbreviated asDFHSCN_STO) .

Before operation, it must be ensured that the touch performance has reached the best state.

DF1CNThe following conditions:DFHCN_STA<DF1CN<DFHSCN_STO.

3)Start the host computer, switch toFAEmodel.

4)Calculate frequency hopping parameters.

three,CTPMapplication

CTP ModuleIt is a mutual capacitive multi-point sensing touch screen module. Include2Parts: Mutual capacitive touch screenCTP with touch screen sensor chip (ICNT86xx) .

3.1 CTPMandHOSTConnection

3.1.1 CTPMCommunication with the host

whenCTPMWhen communicating with the host, there are mainly the following

four forms: A, the host passesI_2CandCPTMdata communication.

B,CTPMWhen a valid touch occurs, the host is notified by outputting an interrupt signal, as shown in the figure3, the interrupt signal passes throughINT Port output, interrupt signal effective level programmable configuration;

C, the host makes theCTPMsleep (detailed in register information), viaINT\WAKE,GPIO4 feet orIICwakeCTPM, the way of the wake-up signal is determined by the firmware;

D, the host resets the port byRSTMakeCTPMReset, active low.

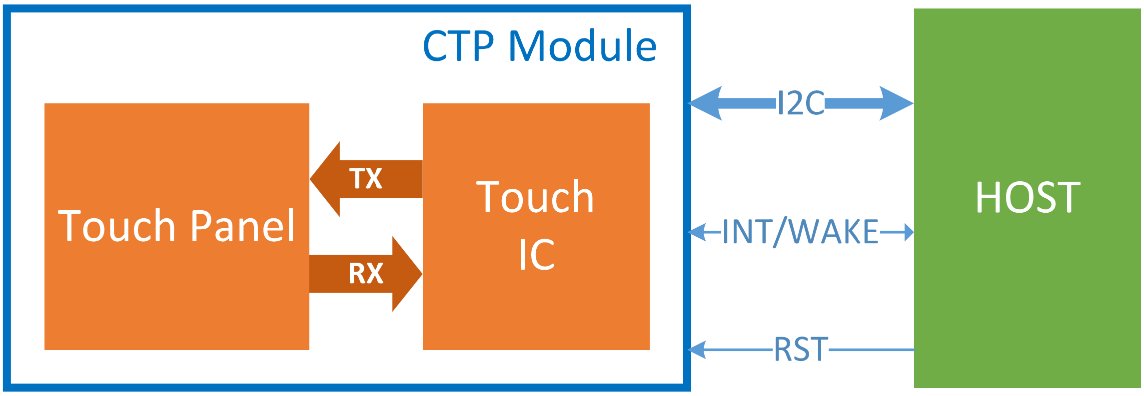

Figure 3 Schematic diagram of the connection between CTPM and HOST

3.2 I2Cinterface

ICNT86Series offers standardI2Ccommunication interface, bySCLandSDACommunicate with the main control chip. The system always acts as a slave device, and the high communication speed is400kbps,support7bitdevice address, which is0x48 (0b1001000). When the master chip addressesCPTMAt the same time, the read and write control bits need to be sent. The read and write control bits are after the slave address field, "0" Indicates a write operation, "1" Represents a read operation, such as:0x90Indicates the address of the write operation,0x91Indicates the address of the read operation.

I2CCommunication is always initiated by the host, and valid start signals are:SCLkeep as "1"hour,SDAHappened by "1"arrive"0" jump. Both the address information and the data stream are transmitted after the start signal. When receiving address information that matches itself,CPTMin the9clock cycles, theSDAChange it to the output port, and place the "0", as a response signal. If you receive address information that does not match you,CPTMwill remain idle.

SDAdata on the mouth9clock cycle serial transmission9Bit data:8Bit significant data+1acknowledgment signal sent by the receiverACKor non-response signalNACK. data transfer inSCLfor"1"valid when. When communication is complete, a stop signal is sent by the host: whenSCLfor"1"hour,SDAStatus by "0"Towards"1"jump.

picture4listed aboveI2CThe format and timing of communication, in which the description and numerical constraints of each timing parameter are shown in the table6.

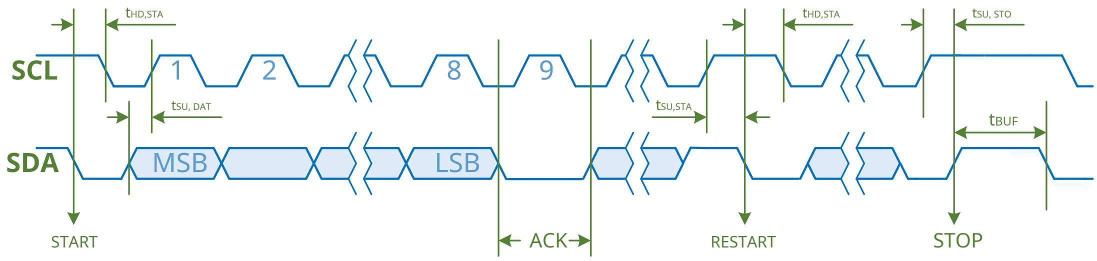

Figure 4 I2C Communication Format and Timing

Table 6 I2C Timing Parameters

illustrate	symbol	small value	Large value	Numerical unit
SCLClock frequency	FSCL		400	kHz
start signal (START/RESTART)hold time	THD_STA	0.6		μs
SCLclock low time	TLOW	1.3		μs
SCLclock high time	THIGH	0.6		μs
Repeated start signal (RESTART)Build time	TSU_STA	0.6		μs
data creation time	TSU_DAT	0.1		μs
data retention time	THD_DAT	0.6	0.9	μs
SCLandSDARise Time	TR		0.3	μs

SCLandSDAfall time	TF		0.3	μs
end signal (STOP)Build time	TSU_STO	0.6		μs
transmission interval	TBUF	1.3		μs

A write operation is completed by a transfer, includingI2CDevice address, write flag, register address (16bits) with the data and the correspondingACKbit, as shown in the figure5shown. A read operation is completed by two transfers, the first transfer of the write register address, includingI2CDevice address, write flags, register addresses and the correspondingACKbit; second transfer reads data, includingI2CDevice address, read flags, data and correspondingACKbit, as shown in the figure6shown. Among them, only read1bytes of data when the first transfer ofSTOPsignal can be omitted;HOSTShould be sent after the next byte of data has been received NACK,Make SLAVERelease the bus.

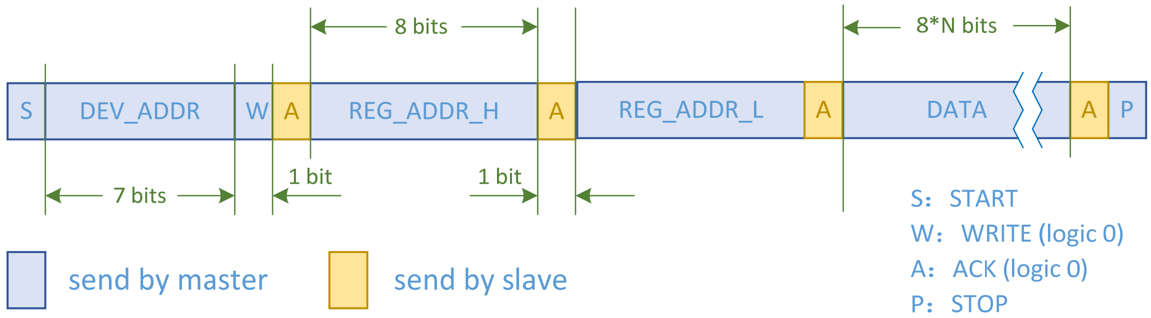

Figure 5 HOST write operation

Figure 6 HOST read operation

3.4Main register information

According to different working modes,CTPMProvide different register information, the host can read the touch point coordinates orCTPMMake settings. The register area is divided into header information area, coordinate information area, configuration register area,Debugarea, which stores different information, such as table7shown.

Table 7 Division of register functional areas

Ribbon name	address range	illustrate
Common Header	0x0000-0x001F	Different Mode have the same Header
Points info	0x1000-0x1FFF	Points info Area
Config Area	0x8000-0x8FFF	Configuration info

surface8Lists the main register addresses and their descriptions in the header information area.

Table 8 Main registers of header information area

register name	address	read and write	Register data and its description
WorkMode	0x0000	R/W	0:Normal mode; 1:FactoryMode;2:ConfigMode Different WorkMode, Different Memory Map
SystemBusyFlag	0x0001	RO	0:idle(can receive another command), 1:busy(doing last command)
Cmd	0x0004	R/W	0x01: CMD_MCU_RESET 0x02: CMD_ENTER_HIBERNATE 0x03: CMD_ENTER_WRITE_PARA_TO_FLASH_MODE 0x04: CMD_WRITE_PARA_TO_FLASH 0x10: CMD_RECOVERY_TX_VOL 0x11: CMD_DEC_TX_VOL_1 0x12: CMD_DEC_TX_VOL_2 0x13: CMD_DEC_TX_VOL_3 0x14: CMD_DEC_TX_VOL_4 0x15: CMD_DEC_TX_VOL_5 0x16: CMD_DEC_TX_VOL_6 0x20: CMD_READ_RAWDATA_AND_CFG_NORMAL 0x21: CMD_END_READ_RAWDATA_AND_CFG_NORMAL 0x30: CMD_UPDATE_BASELINE 0x40: CMD_ENTER_GESTURE_MONITOR 0x41: CMD_QUIT_GESTURE_MONITOR 0x42: CMD_SET_GESTURE_ENABLE_FLAG 0x55: CMD_CHARGER_PLUG_IN 0x66: CMD_CHARGER_PLUG_OUT 0x70: CMD_WRITE_RAWDATA_TO_FLASH 0x80: CMD_AUTO_CLIB_FREQ,

			0x90: CMD_HIGH_SENSE_ENBALE 0x91: CMD_HIGH_SENSE_DISABLE 0xa0: CMD_PROXIMITY_ENABLE 0xa1: CMD_PROXIMITY_DISABLE
PowerMode	0x0005	R/W	0: Active mode,1:Monitor mode,2:Hibernate mode
u8ChargerPlugIn	0x0007	R/W	Charger status
LibVersion	0x0009	RO	lib version
IcVersionMain	0x000A	RO	0x:86 (ICNT86xx)
IcVersionSub	0x000B	RO	0x00(Test Package), 0x01(40QFN), 0x02(48QFN), 0x03(68QFN), 04(xxx)
FirmWareMainVersion	0x000C	RO	- - Firmware main version
FirmWareSubVersion	0x000D	RO	- - Firmware sub version
CumstomerId	0x000E	RO	- - Customer Id
ProductId	0x000F	RO	- - Product Id

surface9Lists the main registers of the coordinate information area and their descriptions.

Table 9 Main registers of coordinate information area

register name	**address**	**read and write**	**Register data and its description**
GestureId	0x1000	RO	Gesture and Virtual Key 0bxxxxxxx1:KEY0 0bxxxxxx1x:KEY1 0bxxxxx1xx:KEY2 0bxxxx 1xxx:KEY3 0bx001xxxx:Gesture_ID01 0bx010xxxx:Gesture_ID02 0bx011xxxx:Gesture_ID03 0bx100xxxx:Gesture_ID04 0bx101xxxx:Gesture_ID05 0bx110xITY:Gesture_ID06 0bx111xxxx:Gesture_ID07 0b1xxxxxxx
NumPointer	0x1001	RO	Number of Pointers(0~10)
Pointer[0].ID	0x1002	RO	0~9
Pointer[0].XL	0x1003	RO	XpositionofPointer[0]
Pointer[0].XH	0x1004	RO	
Pointer[0].YL	0x1005	RO	YpositionofPointer[1]
Pointer[0].YH	0x1006	RO	
Pointer[0].Pressure	0x1007	RO	Pressure Level onPointer[0], 0~255
Pointer[0].EventId	0x1008	RO	0:None 1:Down 2:Move 3:Stay 4:Up

Pointer[1].*	0x1009~0x100F	RO	Information of Pointer[1]
Pointer[2].*	0x1010~0x1016	RO	Information of Pointer[2]
…	…	RO	…
Pointer[9].*	0x1041~0x1047	RO	Information of Pointer[9]

surface10Lists the main registers and their descriptions in the configuration register area.

Table 10 The main configuration registers of the register area

register name	**address**	**read and write**	**Register data and its description**
u16ResX	0x8000		Resolution of Col
u16ResY	0x8001		Resolution of Row
u8RowNum	0x8004		Row total number (Tp + vk)
u8ColNum	0x8005		Column total number (Tp + vk)
u8TXOrder[PHYSICAL_MAX_NUM_ROW]	0x8006		TX Order, start from zero
u8RXOrder[PHYSICAL_MAX_NUM_COL]	0x8030		TX Order, start from zero
u8NumVKey	0x804E		Virtual Key setting
u8VKeyMode	0x804F		Virtual Key Mode
u8TpVkOrder[PHYSICAL_MAX_VK_NUM]	0x8050		Virtual Key Order
u8VKDownThreshold	0x8054		Virtual Key Touch Down Th
u8VKUpThreshold	0x8055		Virtual Key Touch Up Th
u8MaxTouchNum	0x8056		max touch support
u8ThresholdDyncMode	0x8068		Threshold Mode
u8HighSenseThreshold	0x8069		Threshold of High sensitive
u8TouchUpThresold	0x806B		Threshold of Touch up
u8TouchDownThresold	0x806C		Threshold of Touch Down
u8TouchChargerThresold	0x806D		Charger Thresold
u8XySwap	0x80B7		XY Resolution swap
u8IntMode	0x80B8		Interrupt line mode0:Low,1:High
u8IntKeepTime	0x80B9		Interrupt line assert time
u8WakeUpPol	0x80BA		Wake up line Polarity0:Low,1: High
u8GpioVol	0x80BB		Gpio voltage 0:3.3v,1:1.8V
u8ReportRate	0x80BC		Report rate(10~ 200)
u8EnterMonitorCnt	0x80BD		Entering Monitor Mode Max Idle Counter

3.5Power up, power down and reset

picture7Schematic diagram of the power supply rise time.

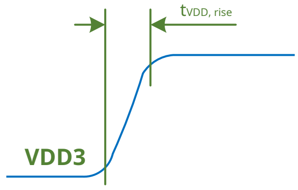

Figure 7 Power supply rise time

picture8It is a schematic diagram of power-on and power-off sequence.

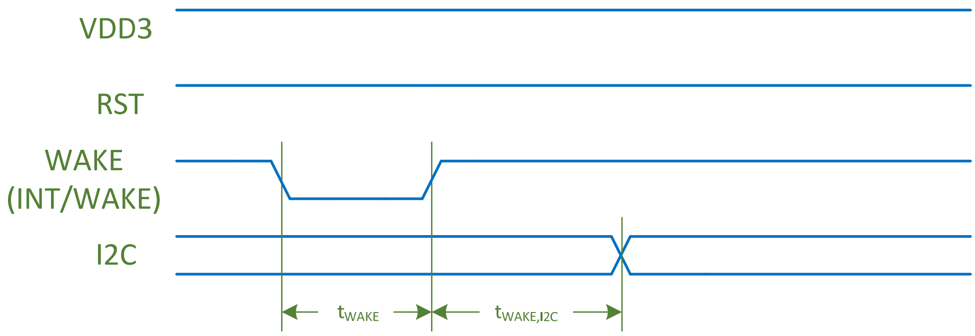

Figure 8 Power-on and power-off sequence

picture9Schematic diagram at reset is listed.

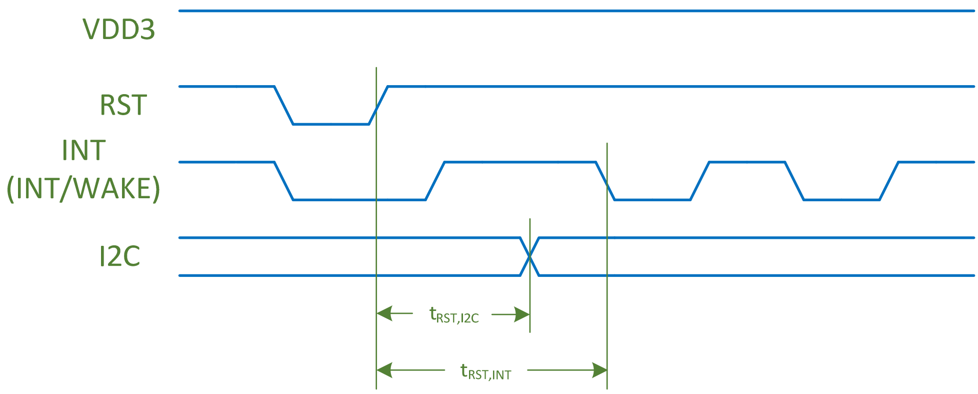

Figure 9 Reset timing

surface11The values of each parameter in the above figure are listed.

Table 11 Power-on and power-off sequence constraints

illustrate	symbol	small value	Large value	unit
power supplyVDD3Rise Time	$t_{VDD, rise}$		10	ms
power supplyVDD3power up toI2CWaiting time to be able to communicate	$t_{VDD, I2C}$	50		ms
power supplyVDD3From power-on to the waiting time of the first reporting point interruption between	$t_{VDD,INT}$	100		ms
power supplyVDD3Valid time for reset before power off	$t_{RST,VDD}$	100		μs
resetRSTarriveI2CWaiting time to be able to communicate	$t_{RST, I2C}$	50		ms
resetRSTWaiting time to the first call interrupt	$t_{RST,INT}$	100		ms

3.6Interrupt, Sleep and Wake

When there is touch information,CTPMneed toHOSTWhen reporting points, the portINTThe interrupt signal will be output, and the effective level of the interrupt signal can be configured by the register (u8IntMode). After an interruption occurs,HOSTWhile reading touch data, interrupt portINTwill return to idle state. ifHOSTIf the data has not been read, the subsequent behavior of the interrupt port is determined by its type.

The interrupt signal is divided into two types: synchronous interrupt and asynchronous interrupt (determined by firmware) . After an interrupt occurs, ifHOST Data has not been read, for synchronization interrupts,CTPMwill remain in the waiting state with no further action; for asynchronous interrupts, after a certain period of time (configured by firmware, the typical value is1ms), the interrupt signal automatically returns to the idle state, and the touch data that triggers the interrupt passes for a period of time (for example, the reporting rate is100Hz, the time is20ms) will be discarded.

whenHOSTto registerCmd(address0x04) to write data0x02hour,CTPMEnter sleep state. need to wake up CTPMhour,HOSTneed to beWAKEThe port is set to an active level (for other wake-up methods, see4.2Festival).WAKEThe valid wake-up level is determined by firmware. picture10with table12The timing diagram and its parameters at wake-up are listed.

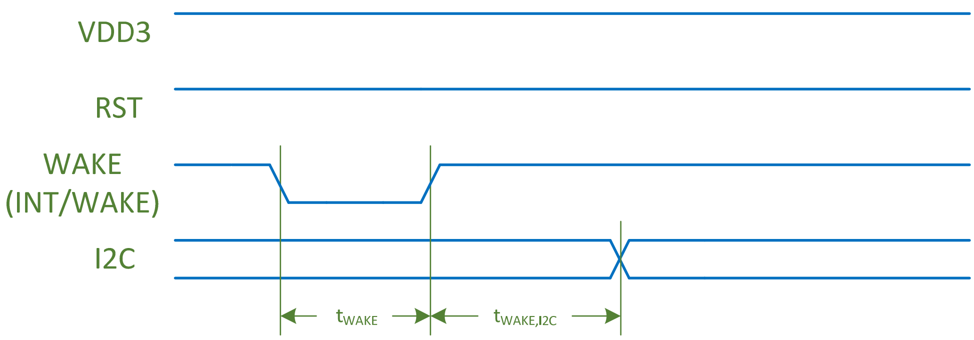

Figure 10 Schematic diagram of low-level wake-up sequence

Table 12 Wake-up Timing Constraints

illustrate	symbol	small value	Large value	unit
wake up signalWAKEEffective time	t_{WAKE}	1		ms
wake up signal toI2CWaiting time to be able to communicate	$t_{WAKE, I2C}$	50		ms

3.7 I2Cwork process

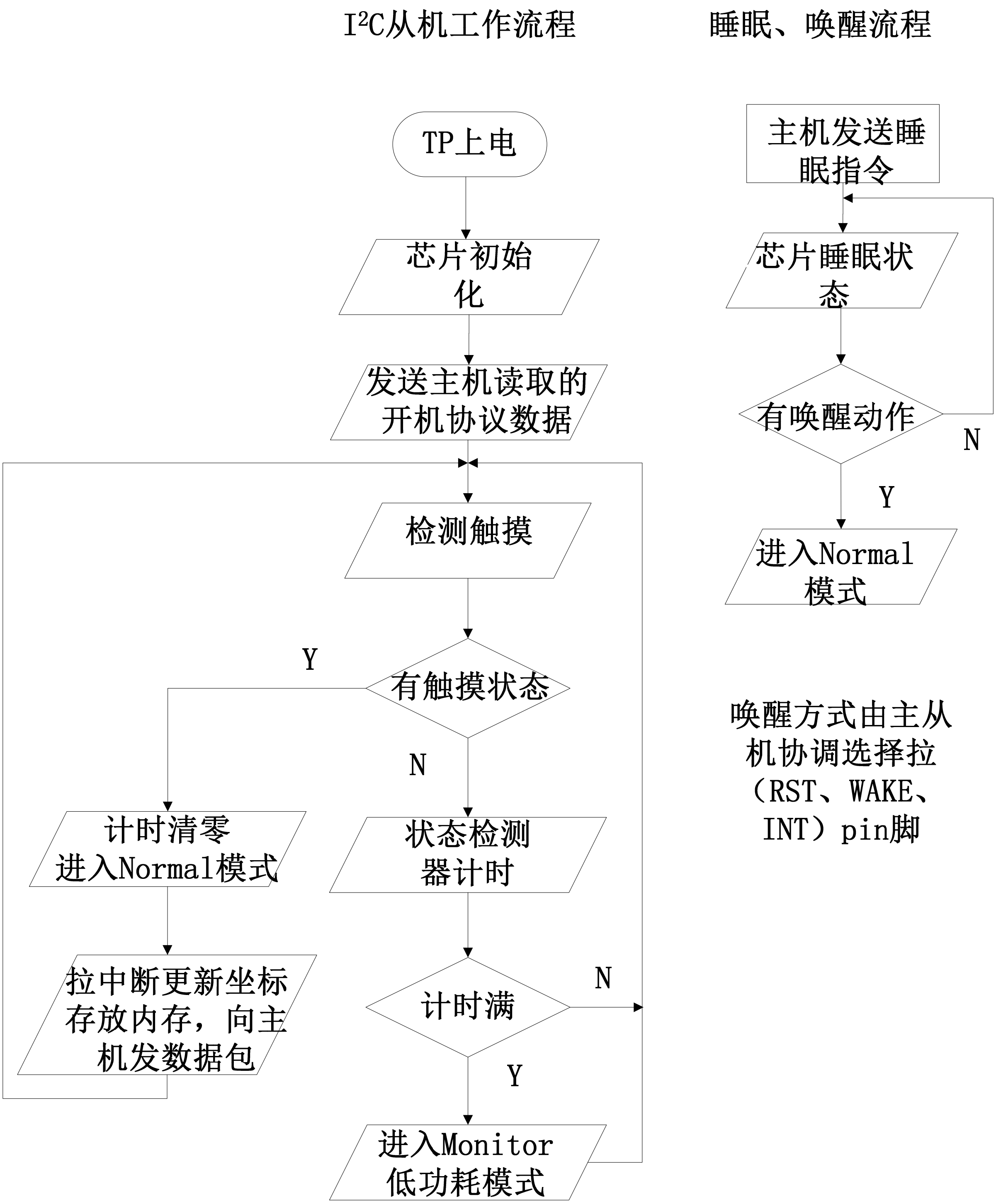

Figure 11 I 2C slave work flow chart

As can be seen from the above figureICNT86The brief workflow of the series chip will scan the screen at the set rate to detect the touch action after completing the initialization. If there is a touch, it will send an interrupt to the host and prepare the touch information to be sent. software configuration) after entering low power consumptionMonitorMode, the software will control to reduce the scanning frequency to detect whether there is a touch action, if so, it will enter the normal mode.

Fourth, the main control terminal platform requirements and programming

existCTPMIn the process of communicating with the master, it always acts as a slave device, and the master will send an addressing command to identify the current touch chip.ICNT86Serial chip defaultI2Cof7The bit address is0x48 (0x1001000), when the host performs read and write operations, it will shift the address one bit to the left and insert the read and write flag at the last position, such as write addressing0x90, read addressing0x91. In order to enable the master and slave parties to communicate in a normal way, ensure thatCTPMworking in normal condition,ICNT86The series chips have certain requirements for the main control terminal platform:

4.1Host Platform Requirements

1) The host has a standardI2Chardware interface andINT,RSTandWAKE(If this method is used to wake upCTMP, available with an output capableGPIOmouth toICNT86ofWAKEpin)

2) supply voltage (VCC)Require:2.6V<=VCC<=3.6V, it is recommended to use2.8V~3.3Vpower supply, guaranteedIC normal operation and meet power consumption requirements, isICThe best working voltage range.

3)IOPort Level Requirements: ForI2Ccommunication line (SDA,SCL) level is guaranteed at1.8VorVCC, the interface is an open-drain structure required4.7k The resistor is pulled up to the power supply. In order to ensure the driving ability and prevent current backflow, the selected pull-up power supply must beIO port level matching. Connection as shown12:

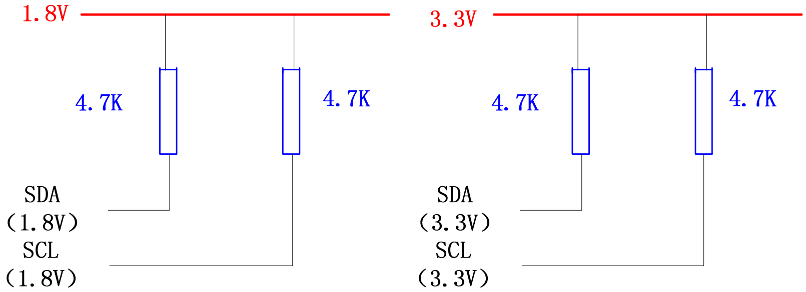

Figure 12 I 2C interface pull-up diagram

4.2Sleep, wake up settings

1) sleep

The host can sendI2Ccommand to letCTPMenter a low-power sleep mode, such as sending (0x04 0x02) sleep command, the host can also control thePMUdisconnectCTPMpower supply, if the power-off method is adopted, it is recommended that the driver connect theCTPM connectedRST,WAKE,INTAlso output low.

2)wake

a) RSTWake: Host availableRSTPull the foot for a certain time (recommended not less than10ms) low level, let the slave reset and enter the normal working mode.

b) GPIOWake: The host can also communicate withCTPMagreed toICNT86xxone ofGPIOmouth like:INT\WAKE or GPIO4Pull a pulse (rising edge, falling edge determined by firmware) to wake up.

c) I2CWake up: After enabling this function,ICNT86xxDuring hibernation if foundI2CWhen there is a device address on the bus that matches the set communication address, it will wake up.

d)Standby gesture to wake up. The specific operations are:

existwakeup pinpick upRCwake-up circuit. In gesture mode,TPautomatic sleep, byRCwake-up circuit, everyN * (DEFAULT_SLEEP_WAKEUP_COUNT) msWake up once, detect whether there is a touch, if there is no touch, continue to sleep, if there is a touch, detect whether there is a gesture; if there is no gesture, continue to sleep, if there is a gesture, exit the gesture mode, and generate an interrupt signal to notify the master, the master receives the message.

After waking up, the screen turns on.

The specific timing and waveform of wake-up can be found in this document3.4and3.5Festival.

CTPMRecalibrates immediately after being awakenedCTPMenvironment in order to adapt to changes in the environment. Master readable slave register verificationCTPMis awakened, Recommended host pairCTPMThe time from wake-up to verification wake-up is not less than50ms.

4.3 I2CWorking mode setting

ICNT86The series chip supports twoI2CThe working mode can be determined by host negotiation:

1.Single Bytemodel

2. Multi-Bytemodel

4.4Interrupt mode setting

ICNT86The series chip reads the real-time touch information by sending an interrupt request to the host. There are two interrupt modes that can be set.

1) edge-triggered

Can negotiate with the host side to trigger a host interrupt using a rising edge or a falling edge (the way most platforms use it)

2) level trigger

Can negotiate with the host side to trigger the host interrupt with high level or low level

4.5Boot and touch information reporting protocol

Mounted in the operating system on the host side withICNT86When driving a series chip, the boot sequence and verification data can be negotiated in the driver, and can be setICNT86The storage method and location of the touch information of the series chips, the host can send the corresponding addressing command to read the corresponding information.

4.6software upgrade

In order to facilitate mass production for users, Chipone North has been improving from the perspective of customersCTMPThere are two main ways to upgrade the program:

1) can be upgraded through the burning tool, including the beta version matched with the host computer software (which can be used forFPC,CTMPModule inspection and upgrade procedures) and mass production burning tools (batch burning of chips), please refer to the mass production tool instructions for details.

2)By upgrading the operating system of the main control terminal, the main control chip is mountedICNT86Available when the series chip is drivenadb(already Platform verification is successful) method to upgrade; or useapkway to upgrade, seeICNT86Serial chip driver porting guide.

V. Important Statement

Device application information and other similar content described in this publication is provided solely to you and may be superseded by updated information. It is your responsibility to ensure that your application meets the technical specifications.ChiponeNo representations or warranties, express or implied, written or oral, statutory or otherwise, are made with respect to this information, including but not limited to representations or warranties of its use, quality, performance, merchantability or fitness for a particular purpose.ChiponeNo responsibility is assumed for this information and the consequences of its use. withoutChiponeWritten approval shall not beChiponeof products used as critical components in life support systems. existChiponeUnder the protection of intellectual property rights, no license may be implicitly or otherwise transferred.

Revision History

Version	Revisions	Date	Modified by
V1.0	1.initial version	2015-03-16	