Chipsmall
深圳市津利帝科技有限公司
Geelead Technology Limited
Chipsmall
采购部

STMPE321

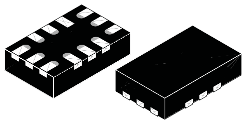
3-channel capacitive touchkey controller

Features

- Up to 3 GPIOs
- Up to 3 capacitive touchkey inputs
- Operating voltage 1.65 - 1.95 V
- Interrupt output pin
- I^2C interface (1.8 V operation, 3.3 V tolerant)
- 8 kV HBM ESD protection
- 40 fF resolution, 128-step capacitance measurement
- Advanced data filtering (AFS)
- Environment tracking calibration (ETC)
- Individually adjustable touch variance (TVR) settings for all channels
- Adjustable environmental variance (EVR) for optimal calibration
- Capacitive key sensing capability in 27 µA sleep mode

Applications

- Mobile phones and smartphones
- Portable media players
- Game consoles



QFN12

(2.2 x 1.5 mm)

Description

The STMPE321 is a 3-channel capacitive touchkey controller. Capacitance measurement is implemented in fully optimized hardware.

All 3 I/Os can be configured via an I^2C bus to function as either capacitive touchkey, or as GPIOs (general purpose I/O).

Table 1. Device summary

Order code	Package	Packing
STMPE321QTR	QFN12 (2.2 x 1.5 mm)	Tape and reel

Contents

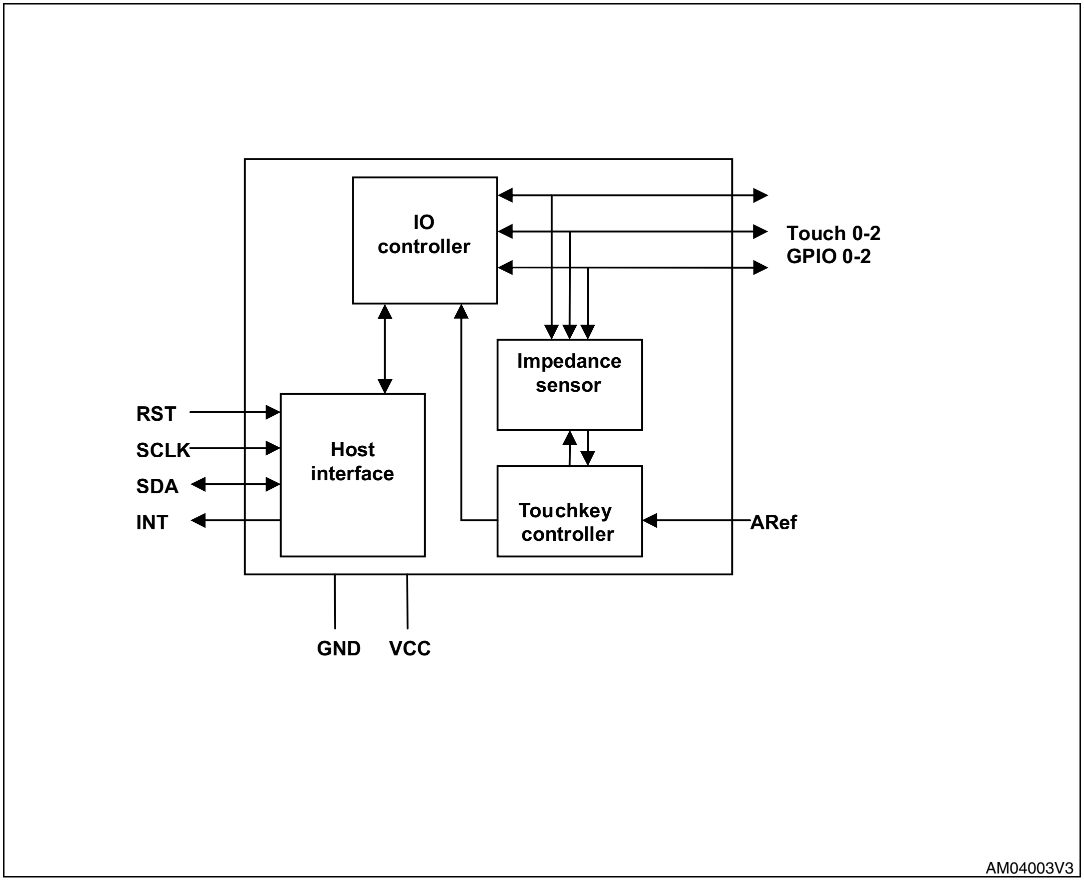
1 STMPE321 functional overview

The STMPE321 consists of the following blocks:

- GPIO controller
- Impedance sensor
- Touchkey controller
- I^{2}C interface

1.1 STMPE321 block diagram

Figure 1. Functional block diagram



1.2 Pin assignment and function

Figure 2. QFN12 pin assignment (top view)

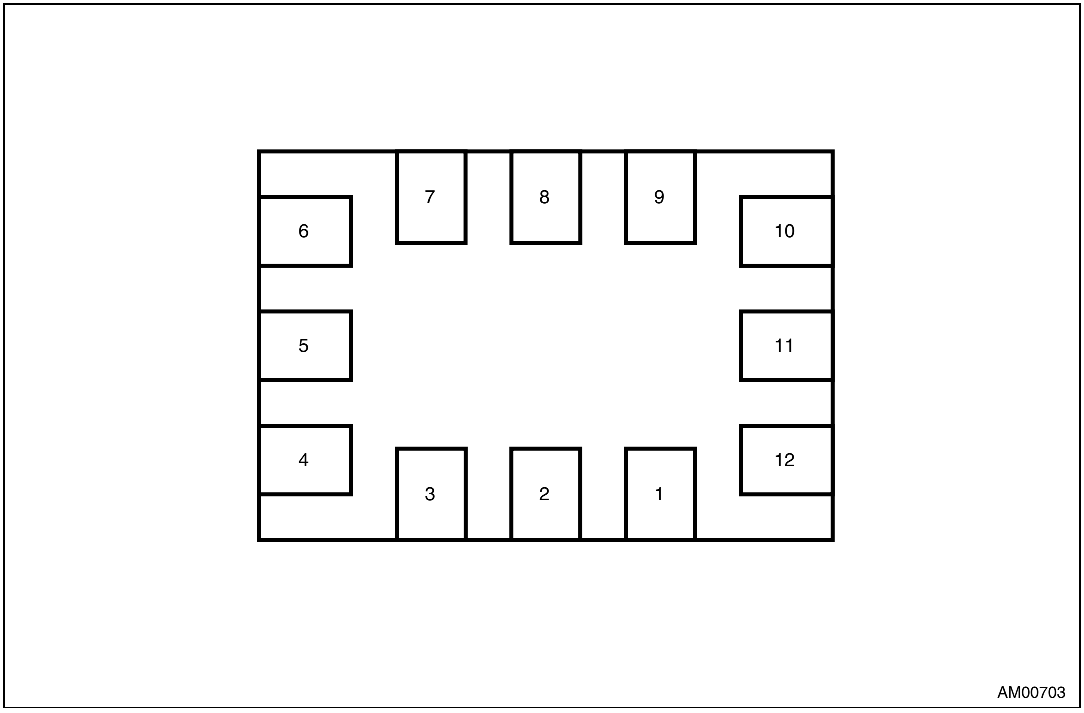


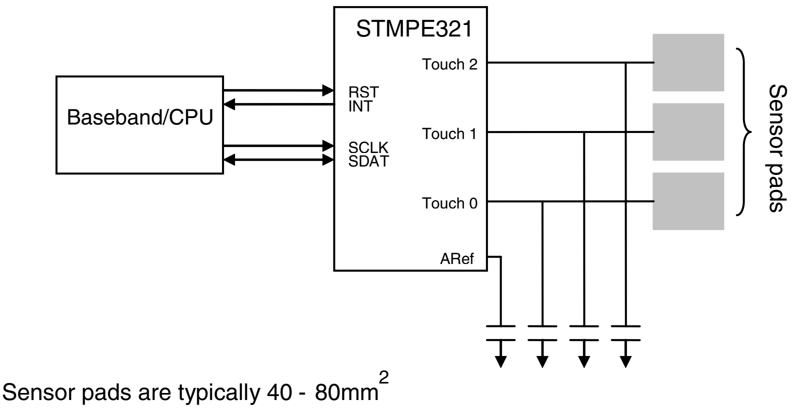
Table 2. Pin assignment and function

Pin number	Pin name	Description
1	GPIO_2 / touch 2	GPIO 2
2	GPIO_1 / touch 1	GPIO 1
3	GPIO_0 / touch 0	GPIO 0
4	NC	-
5	SDA	I^2C data
6	SCL	I^2C clock
7	GND	GND
8	VCC	Supply voltage
9	ARef	Reference capacitor for touch sensor
10	NC	-
11	INT	INT output (open drain)
12	RST	RESET (active low) This pin is internally pulled up to V_{CC}

1.3 STMPE321 typical application

The STMPE321 is capable of supporting capacitive sensors of up to 3 channels.

Figure 3. Typical application diagram



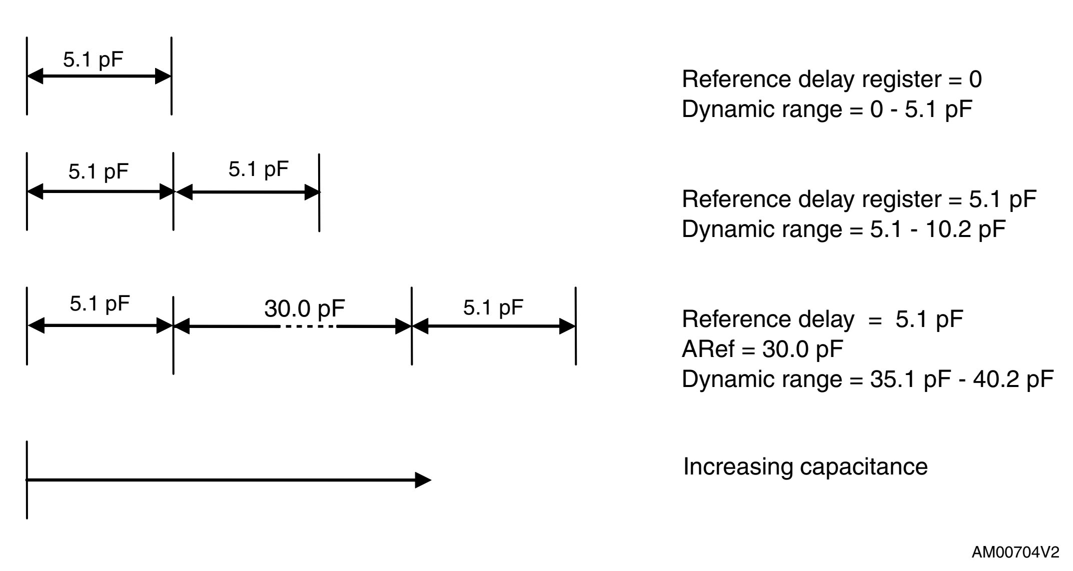
2 Capacitance compensation

The STMPE321 is capable to measuring up to 5.1pF in capacitance difference between the reference point (Zref) and the individual channels. In the case where PCB connection between the sensor pads and the device is too long, the "REFERENCE DELAY" register is able to shift the reference by up to 5.1pF, allowing the TOUCH channels to measure added capacitance 5.1pF with offset of 5.1pF, as shown in following diagram.

In the case where this is still not enough to compensate for the capacitance on sensor lines (due to very long sensor trace), an external capacitor of up to 30 pF could be connected at the A_Ref pin.

This would further shift up the dynamic range of the capacitance measurement.

Figure 4. Capacitance compensation


5.1 pF
Reference delay register = 0
Dynamic range = 0 - 5.1 pF
5.1 pF
5.1 pF
Reference delay register = 5.1 pF
Dynamic range = 5.1 - 10.2 pF
5.1 pF
30.0 pF
5.1 pF
Reference delay = 5.1 pF
ARef = 30.0 pF
Dynamic range = 35.1 pF - 40.2 pF
Increasing capacitance
AM00704V2

The sensed capacitance is accessible to the host through the IMPEDANCE registers.

2.1 Calibration algorithm

The STMPE321 maintains 2 parameters for each touch channel: TVR and CALIBRATED IMPEDANCE. CALIBRATED IMPEDANCE is an internal reference which, if the currently measured IMPEDANCE exceeds the CALIBRATED IMPEDANCE by a magnitude of TVR, is considered a “TOUCH”.

If the IMPEDANCE is higher than the CALIBRATED IMPEDANCE, but the magnitude does not exceed CALIBRATED IMPEDANCE by TVR, it is not considered a TOUCH. In this case, 2 scenarios are possible:

1. Environmental changes have caused the IMPEDANCE to increase

2. Finger is near the sensing pad, but not near enough

In case 1, the change in IMPEDANCE is expected to be small, as environmental changes are normally gradual. A value "EVR" is maintained to specify the maximum IMPEDANCE change that is still considered an environmental change.

Table 3. Calibration action under different scenarios

Scenario	Touch sensing and calibration action
IMP>CALIBRATED IMP + TVR	TOUCH, no calibration
IMP<CALIBRATED IMP + TVR IMP>CALIBRATED IMP + EVR	NO TOUCH, no calibration
IMP<CALIBRATED IMP + TVR IMP<CALIBRATED IMP + EVR IMP>CALIBRATED IMP	NO TOUCH, new CALIBRATED IMP = previous CALIBRATED IMP + change in IMP
IMP>CALIBRATED IMP	CALIBRATED IMP + change in IMP
IMP<CALIBRATED IMP	NO TOUCH, new CALIBRATED IMP = new IMP

‘IMP’ and ‘CALIBRATED IMP’ used in this table is not the direct register read-out.

IMP = 127 - impedance register readout

CALIBRATED IMP = 127 - calibrated impedance register readout.

The ETC WAIT register states a period of time for which all TOUCH inputs must remain "NO TOUCH" for the next calibration to be carried out.

The CAL INTERVAL states the period of time between successive calibrations when there are prolonged NO TOUCH conditions.

2.1.1 Noise filtering

When the STMPE321 is operating in the vicinity of highly emissive circuits (DC-DC converters, PWM controllers/drives etc.), the sensor inputs can be affected by high-frequency noise. In this situation, the time-integrating function can be used to distinguish between a real touch, or an emission-related false touch.

The INTEGRATION TIME and STRENGTH THRES registers are used to configure the time-integrating function of the STMPE321.

2.1.2 Data filtering

The output from the calibration unit provides an instantaneous TOUCH or NO TOUCH status. This output is directed to the filtering stage where the TOUCH is integrated across a programmable period of time. The output of the integration stage is a “STRENGTH" (in the STRENGTH register) that indicates the number of times a TOUCH is detected across the integration period.

The STRENGTH is then compared to the value in STRENGTH THRESHOLD register. If STRENGTH exceeds the STRENGTH THRESHOLD, it is considered a final, filtered TOUCH status.

In the data filtering stage, 3 modes of operation are supported:

Mode 1: Only the touch channel with highest STRENGTH is taken

Mode 2: All touch channels with STRENGTH > STRENGTH THRESHOLD are taken

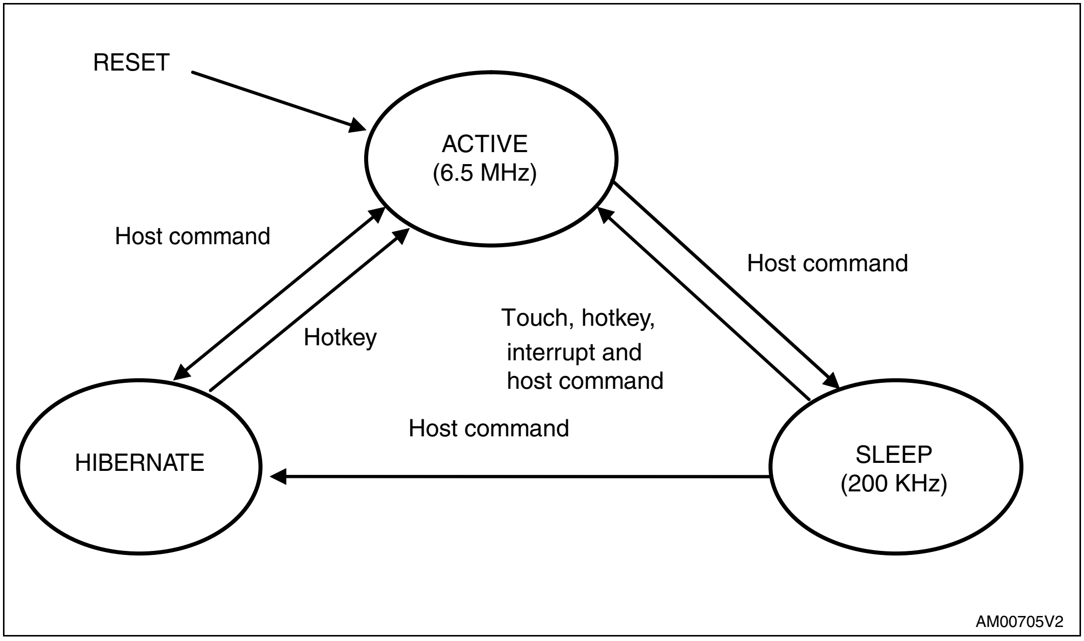
Mode 3: The 2 touch channels with the highest STRENGTH are taken.

These modes are selected using the FEATURE SELECTOR register. The final, filtered data is accessible through the Touch Byte register.

2.2 Power management

The STMPE321 operates in 3 states, as described below:

Figure 5. STMPE321 operating states



On RESET, the STMPE321 enters the ACTIVE state immediately.

Upon a fixed period of inactivity, the device enters a SLEEP state. Any touch activity occurring during a SLEEP state causes the device to return to an ACTIVE state.

In SLEEP mode:

-Calibration continues if the F2A bit is set in the CONTROL register

-Calibration stops if the F2A bit is NOT set in the CONTROL register

If no touch activity is expected, the host may set the device to a HIBERNATE state to save power.

If any key is touched and held, the I^{2}C command to enter SLEEP or HIBERNATE is put on hold until the key is released.

3 I^{2}C interface

The following features are supported by the I^{2}C interface:

- I^{2}C slave device
- Compliance with Philips I^{2}C specification version 2.1
- Standard (up to 100 kbps) and fast (up to 400 kbps) modes.
- 7-bit and 10-bit device addressing modes
- General call
- Start/Restart/Stop
- I^{2}C address is 0x58 (0xB0/0xB1 for write/read, including the LSB)

Start condition

A Start condition is identified by a falling edge of SDATA while SCLK is stable at high state. A Start condition must precede any data/command transfer. The device continuously monitors for a Start condition and does not respond to any transaction unless one is encountered.

Stop condition

A Stop condition is identified by a rising edge of SDATA while SCLK is stable at high state. A Stop condition terminates communication between the slave device and bus master. A read command that is followed by NoAck can be followed by a Stop condition to force the slave device into idle mode. When the slave device is in idle mode, it is ready to receive the next I^{2}C transaction. A Stop condition at the end of a write command stops the write operation to the registers.

Acknowledge bit (ACK)

The acknowledge bit is used to indicate a successful byte transfer. The bus transmitter releases the SDATA after sending eight bits of data. During the ninth bit, the receiver pulls the SDATA low to acknowledge the receipt of the eight bits of data. The receiver may leave the SDATA in high state if it does not acknowledge the receipt of the data.

Data Input

The device samples the data input on SDATA on the rising edge of the SCLK. The SDATA signal must be stable during the rising edge of SCLK and the SDATA signal must change only when SCLK is driven low.

Memory addressing

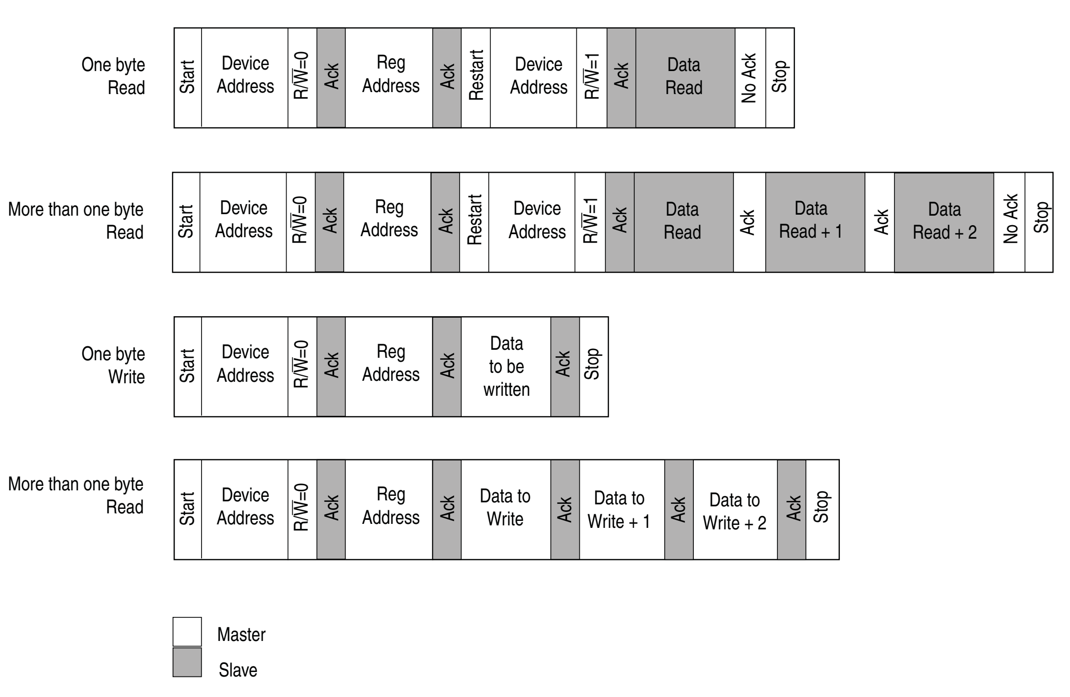
For the bus master to communicate to the slave device, the bus master must initiate a Start condition, followed by the slave device address. Accompanying the slave device address, there is a Read/$\overline{\text{WRITE}}$ bit (R/$\overline{\text{W}}$). The bit is set to 1 for a read operation, and 0 for a write operation.

If a match occurs on the slave device address, the corresponding device gives an acknowledgement on the SDA during the 9th bit time. If there is no match, it deselects itself from the bus by not responding to the transaction.

Table 4. Operation modes

Mode	Byte	Programming sequence
Read	≥1	Start, Device address, R/$\overline{W}$ = 0, Register address to be read
		Restart, Device address, R/$\overline{W}$ = 1, Data Read, STOP
		If no Stop is issued, the Data Read can be continuously performed. If the register address falls within the range that allows an address auto-increment, then the register address auto-increments internally after every byte of data being read. For those register addresses that fall within a non-incremental address range, the address is kept static throughout the entire write operations. Refer to the memory map table for the address ranges that are auto and non-increment. An example of such a non-increment address is FIFO.
Write	≥1	Start, Device address, R/$\overline{W}$ = 0, Register address to be written, Data Write, Stop
		If no Stop is issued, the Data Write can be continuously performed. If the register address falls within the range that allows address auto-increment, then the register address auto-increments internally after every byte of data being written in. For those register addresses that fall within a non-incremental address range, the address is kept static throughout the entire write operations. Refer to the memory map table for the address ranges that are auto and non-increment.

Figure 6. Read and write modes (random and sequential)



4 Register map and function description

This section lists and describes the registers in the STMPE321 device starting with a register map, and then provides detailed descriptions of the register types.

Table 5. Register summary map table

Address	Register name	Bit	Type	Reset value	Function
0x00	CHIP_ID_0	8	R	0x03	Device identification
0x01	CHIP_ID_1	8	R	0x21	Device identification
0x02	ID_VER	8	R	0x03	Revision number
0x03	SYS_CFG_1	8	R/W	0x00	System configuration 1
0x04	SYS_CFG_2	8	R/W	0xEF	System configuration 2
0x08	INT_CTRL	8	R/W	0x01	Interrupt control register
0x09	INT_EN	8	R/W	0x01	Interrupt enable register
0x0A	INT_STA	8	R	0x09	Interrupt status register
0x0B	GPIO_INT_EN_lsb	8	R/W	0x00	GPIO interrupt enable register
0x0C	GPIO_INT_EN_msb	8	R/W	0x00	GPIO interrupt enable register
0x0D	GPIO_INT_STA_lsb	8	R/W	0x00	GPIO interrupt status register
0x0E	GPIO_INT_STA_msb	8	R/W	0x00	GPIO interrupt status register
0x10	GPIO_MR	8	R/W	0x00	GPIO monitor pin
0x12	GPIO_SET	8	R/W	0x00	GPIO set pin state register
0x14	GPIO_DIR	8	R/W	0x00	GPIO set pin direction register
0x16	GPIO_FUNCT	8	R/W	0x00	GPIO function register
0x18	TOUCH_FIFO	64	R	0x00	Fifo access for touch data buffer
0x20	FEATURE_SEL	8	R/W	0x04	Feature selection
0x21	ETC_WAIT	8	R/W	0x27	Wait time
0x22	CAL_INTERVAL	8	R/W	0x30	Calibration interval
0x23	INTEGRATION_ TIME	8	R/W	0x0F	Integration time
0x25	CTRL	8	R/W	0x00	Control
0x26	INT_MASK	8	R/W	0x08	Interrupt mask
0x27	INT_CLR	8	R/W	0x00	Interrupt clear
0x28	FILTER_PERIOD	8	R/W	0x00	Filter period
0x29	FILTER_THRESHOLD	8	R/W	0x00	Filter threshold
0x2A	REF_DLY	8	R/W	0x00	Reference delay
0x30 - 0x32	TVR [0-2]	8	R/W	0x08	Touch variance setting
0x40	EVR	8	R/W	0x04	Environmental variance

Table 5. Register summary map table (continued)

Address	Register name	Bit	Type	Reset value	Function
0x50-0x52	STRENGTH_THRES [0-2]	8	R/W	0x01	Setting of strength threshold for each channel
0x60 - 0x62	STRENGTH [0-2]	8	R	0x00	Strength
0x70 - 0x72	CAL_IMPEDANCE [0-2]	8	R	0x00	Calibrated impedance
0x80 - 0x82	IMPEDANCE [0-2]	8	R	0x00	Impedance
0x92	INT_PENDING	8	R/W	0x00	Status of GINT interrupt sources

5 System and identification registers

Table 6. System and identification registers map

Address	Register name	Bit	Type	Reset	Function
0x00	CHIP_ID_0	8	R	0x03	Device identification
0x01	CHIP_ID_1	8	R	0x21	Device identification
0x02	ID_VER	8	R	0x03	Revision number
0x03	SYS_CFG_1	8	R/W	0x00	System configuration 1
0x04	SYS_CFG_2	8	R/W	0xEF	System configuration 2

CHIP_ID_x Device identification

Address: 0x00, 0x01

Type: R

Reset: 0x03, 0x21

Description: 16(8+8)-bit device identification

ID_VER Revision number

Address: 0x02

Type: R

Reset: 0x03

Description: 8-bit revision number

SYS_CFG_1 System configuration 1

7	6	5	4	3	2	1	0
RESERVED				SLEEP	WARM_RESET	SOFT_RESET	HIBERNATE

Address: 0x03

Type: R/W

Reset: 0x00

Description: The reset control register enables the reset of the device

[7:4] **RESERVED**

[3] **SLEEP**:
Write ‘1’ to enable sleep mode. Hardware resets this bit to ‘0’ after it successfully enters sleep mode.

[2] **WARM_RESET**:
Write ‘1’ to initiate a warm reset. Register content remains, state machine reset.

[1] **SOFT_RESET**:
Write ‘1’ to initiate a soft reset. All registers content and state machines reset.

[0] **HIBERNATE**: Force the device into hibernation mode.
Write ‘1’ to enter hibernate mode. Hardware resets this bit to ‘0’ after it successfully enters hibernate mode.

SYS_CFG_2 System configuration 2

7	6	5	4	3	2	1	0
SENSOR CLOCK 2	SENSOR CLOCK 1	SENSOR CLOCK 0	RESERVED		GPIO CLOCK DISABLE	FIFO CLOCK DISABLE	TOUCH CLOCK DISABLE

Address: 0x04

Type: R/W

Reset: 0xEF

Description: This register enables the switching off of the clock supply

[7:5] **SENSOR CLOCK**: See description in *Table 7.*

[4] **RESERVED**

[3] **RESERVED**

[2] **GPIO CLOCK DISABLE**:
Write ‘1’ to disable the clock to GPIO unit.

[1] **FIFO CLOCK DISABLE**:
Write ‘1’ to disable the clock to FIFO unit. This must be set to ‘0’ if touch interrupt is required.

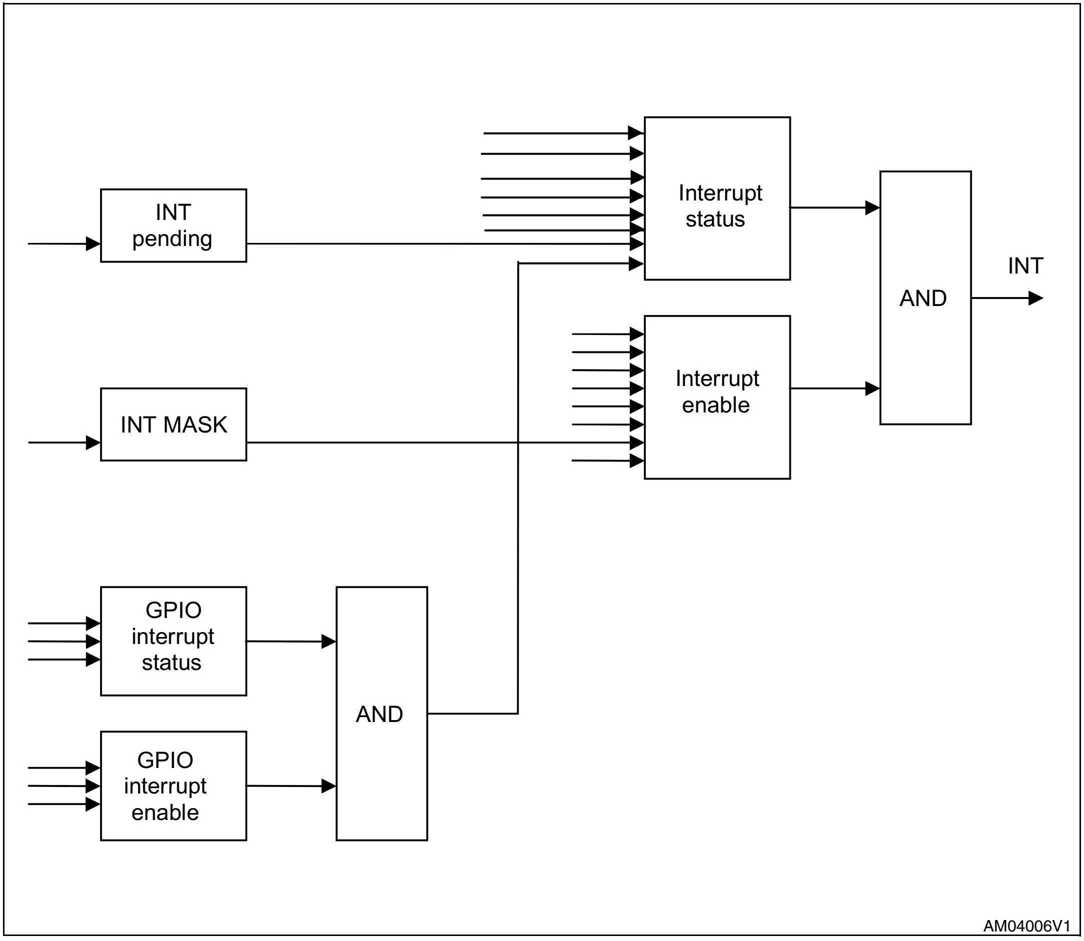
[0] **TOUCH CLOCK DISABLE**:
Write ‘1’ to disable the clock to TOUCH unit.

Table 7. Sensor clock setting

Mode	Divider	Sensor clock [2:0]	Active	Calibration
Operational (6.5 MHz)	1	000	12.8 kHz	100 kHz
	2	001	6.4 kHz	50 kHz
	4	010	3.2 kHz	25 kHz
	8	011	1.6 kHz	12.5 kHz
	16	1xx	800 Hz	6.25 kHz
Autosleep (200 kHz)	1	000	400 Hz	3.2 kHz
	2	001	200 Hz	1.6 kHz
	4	010	100 Hz	800 Hz
	8	011	50 Hz	400 Hz
	16	1xx	25 Hz	200 Hz

6 Interrupt controller module

Figure 7. Interrupt controller module block diagram



INT_CTRL Interrupt control register

7	6	5	4	3	2	1	0
RESERVED					POLARITY	TYPE	INT_EN

Address: 0x08

Type: R/W

Reset: 0x01

Description: This register is used to enable control of the polarity, edge/level and enabling of the interrupt system device.

[7:3] **RESERVED**

[2] **POLARITY**:

'0' for active low

'1' for active high

For active low operation, the INT pin should be externally pulled high. The INT pin is pulled to GND when there is a pending interrupt.

For active high operation, the INT pin should be externally pulled to GND. In this mode, the INT pin is pulled to V_{CC} by the device when there is a pending interrupt.

[1] **TYPE**:

'0' for level trigger

'1' for edge trigger (pulse width is 200 nS)

[0] **INT_EN**:

'0' to disable all interrupts

'1' to enable all interrupts

INT_EN Interrupt enable register

7	6	5	4	3	2	1	0
GPIO	RESERVED				GEN	FIFO	POR

Address: 0x09

Type: R/W

Reset: 0x01

Description: This register is used to enable the interruption from a system related interrupt source to the host. Writing ‘1’ in this register enables the corresponding interrupt event to generate interrupt signal at the INT pin. Note that even if the interrupt is not enabled, an interrupt event is still reflected in the interrupt status register.

[7] **GPIO**:
One or more level transition in enabled GPIOs

[6:3] **RESERVED**
Must be set to ‘0’ at all times.

[2] **GEN**:
System INT (A2I, I2A, EOC)

[1] **FIFO**:
Data available in FIFO. This interrupt can be cleared only if FIFO is empty.

[0] **POR**:
Power-on reset

INT_STA Interrupt status register

7	6	5	4	3	2	1	0
GPIO	RESERVED				GEN	FIFO	POR

Address: 0x0A

Type: R/W

Reset: 0x09

Description: This register is used to enable the interruption from a system related interrupt source to the host. Regardless of whether or not the IESYSIOR bits are enabled, the ISSYSIOR bits are still updated. Writing ‘1’ clears a bit in this register. Writing ‘0’ has no effect.

[7] **GPIO**:
One or more level transition in enabled GPIOs

[6:3] **RESERVED:**
Some of these bits might be set to '1' by hardware during normal operation. The content of these bit is for internal operation and are not required for normal use of device.

[2] **GEN**:
System INT (A2I, IA2, EOC)

[1] **FIFO**:
Data available in FIFO

[0] **POR**:
Power-on reset

GPIO_INT_EN GPIO interrupt enable registerl

7	6	5	4	3	2	1	0
RESERVED					IEG		

Address: 0x0B, 0x0C

Type: R/W

Reset: 0x00

Description: The GPIO interrupt enable register is used to enable the interruption from a particular GPIO interrupt source to the host. The IEg[2:0] bits and the interrupt enable mask bits correspond to the GPIO[2:0] pins.

[7:3] **RESERVED**

[2:0] **IEG[2:0]**

Interrupt enable GPIO mask (where x = 2 to 0)

Writing a ‘1’ to the IE[x] bit enables the interruption to the host.

GPIO_INT_STA GPIO interrupt status register

7	6	5	4	3	2	1	0
ISG							

Address: 0x0D, 0x0E

Type: R/W

Reset: 0x00

Description: The GPIO interrupt status register LSB monitors the status of the interruption from a particular GPIO pin interrupt source to the host. Regardless of whether or not the IEGPIOR bits are enabled, the INT_STA_GPIO_LSB bits are still updated. The ISG[2:0] bits are the interrupt status bits correspond to the GPIO[2:0] pins.

[7:0] **ISG[x]**:

Interrupt status GPIO (where x = 2 to 0)

Read:

Interrupt status of the GPIO[x]. Writing ‘1’ clears a bit. Writing ‘0’ has no effect.

7 GPIO controller

A total of 3 GPIOs are available in the STMPE321. The GPIO controller contains the registers that allow the host system to configure each of the pins into either a GPIO or Touch input. Unused GPIOs should be configured as outputs to minimize power consumption.

A group of registers is used to control the exact function of each of the 3 GPIOs. The registers and their respective addresses are listed in *Table 8.*

Table 8. GPIO controller registers summary map

Address	Register name	Description	Auto-increment
0x10	GPIO_MR_LSB	GPIO monitor pin state register	Yes
0x11	GPIO_MR_MSB		
0x12	GPIO_SET_LSB	GPIO set pin state register	Yes
0x13	GPIO_SET_MSB		
0x14	GPIO_DIR_LSB	GPIO set pin direction register	Yes
0x15	GPIO_DIR_MSB		
0x16	GPIO_FUNCT_LSB	GPIO function register	Yes
0x17	GPIO_FUNCT_MSB		

All GPIO registers are named GPxx, where:

Xxx represents the functional group

For LSB registers:

7	6	5	4	3	2	1	0
RESERVED					IO-2	IO-1	IO-0

For MSB registers:

7	6	5	4	3	2	1	0
RESERVED							

The function of each bit is shown in *Table 9*:

Table 9. GPIO control bits function

Register name	Function
GPIO monitor pin state	Reading this bit yields the current state of the bit. Writing has no effect.
GPIO set pin state	Writing '1' to this bit causes the corresponding GPIO to go to '1' state Writing '0' to this bit causes the corresponding GPIO to go to '0' state
GPIO set pin direction	'0' sets the corresponding GPIO to input state, and '1' sets it to output state. All bits are '0' on reset.
GPIO function	'1' sets the corresponding GPIO to function as GPIO, and '0' sets it to touchkey mode.

8 Capacitive touch module registers

Table 10. TOUCH_FIFO summary table

Address	Function
0x18	FIFO-0, LSB
0x19	FIFO-0, MSB
0x1A	FIFO-1, LSB
0x1B	FIFO-1, MSB
0x1C	FIFO-2, LSB
0x1D	FIFO-2, MSB
0x1E	FIFO-3, LSB
0x1F	FIFO-3, MSB

TOUCH_FIFO Touch FIFO

7	6	5	4	3	2	1	0
T7	T6	T5	T4	T3	T2	T1	T0

Address: 0x19, 0x18

Type: R

Reset: 0x00

Description: TOUCH_FIFO is the access port for the internal 4-level FIFO used for buffering the touch events. While it is possible to access each byte in the data structure directly, it is recommended that the FIFO is accessed only via the 0x18 address.
The FIFO must be accessed in multiples of 2 bytes (LSB, MSB). For the STMPE321, MSB is reserved and LSB contains a snapshot of the recent touch event.
Where Tn is touch status of touch sensing channel n.

FEATURE_SELECT Feature select

7	6	5	4	3	2	1	0
RESERVED					AFS[1:0]		Filter EN

Address: 0x20

Type: R/W

Reset: 0x04

Description: Controls AFS (advanced filtering system and second level filtering feature)

[7:3] **RESERVED**

[2:1] **AFS[1:0]**:
“00’: reserved
“01’ AFS mode 1 (only 1 strongest key)
‘10’: AFS mode 2 (all keys that are above threshold)
‘11’: AFS mode 3 (the 2 strongest keys)

[0] **Filter EN**:
Write '1' to enable filter

ETC_WAIT Wait time setting

7	6	5	4	3	2	1	0
ETC_WAIT[7:0]							

Address: 0x21

Type: R/W

Reset: 0x27

Description: Sets the wait time between the calibration and the last button touch

[7:0] **ETC_WAIT[7:0]**:
ETC wait time = ETC_Wait[7:0] *64 + sensor clock period

A "non-touch" condition must persist for this wait time, before an ETC operation is carried out

Range: 5 mS - 20 s