PTC Subsystem Firmware

PTC Subsystem Firmware User's Guide

Scope

This User's Guide introduces the PTC (Peripheral Touch Controller) Subsystem Firmware and describes the operation of the software settings provided by the drivers. The effects of each parameter and setting required to ensure proper operation of PTC are clearly described.

This document refers to PTC Subsystem Firmware Version 6.0 or later.

Reference Documents

Title	Document Type	Lit. No.
QTouch on SAMA5D2 MPU	Application Note	AN2472
QTAN0079 - Buttons, Sliders and Wheels Sensor Design Guide	Application Note	10752

Table of Contents

1. Overview

1.1 Description

The PTC subsystem is intended for autonomously performing capacitive touch sensor measurements. The external capacitive touch sensor is typically formed on a PCB, and the sensor electrodes are connected to the analog charge integrator of the PTC using the device I/O pins.

In Mutual Capacitance mode, the PTC requires one pin per X line (drive line) and one pin per Y line (sense line). In Self-capacitance mode, the PTC requires only one pin with a Y-line driver for each self-capacitance sensor.

1.2 Features

- Implements low-power, high-sensitivity, environmentally robust capacitive touch buttons, sliders, and wheels
  - One pin per electrode - no external components
  - Zero drift over temperature and supply/reference ranges
  - No need for temperature or supply/reference compensation
- “On demand” or “Timed” measurement
- Supports mutual capacitance and self-capacitance sensing
  - Up to 8 buttons in Self-capacitance mode
  - Up to 64 buttons in Mutual Capacitance mode
  - Supports Lumped mode configuration(1)
- Calibration
  - Load compensating charge sensing
  - Parasitic capacitance compensation together with the electrode capacitance
- Adjustable gain for higher sensitivity
  - Analog gain 1 to 16
  - Digital gain 1 to 32
- Noise immunity
  - Hardware noise filtering by accumulation 1 to 64
  - Adjacent Key Suppression, removal of false detection(2)
  - Frequency hopping: noise signal de-synchronization for high conducted immunity(3)
- Provided PTC Subsystem Firmware(4)
- Acquisition module (node definitions, pPP and PTC management) is product-dependent, which implements all hardware-dependent operations for configuration and measurement of capacitive touch or proximity sensors.
- Signal conditioning module (frequency hopping) applies algorithmic and feedback control methods to improve the quality of measurement data captured by an acquisition module.
- Post-processing modules (Key, Scroller) interpret measurement data in the context of a capacitive touch or proximity sensor.
- Scroller module defines Slider and Wheels configuration and data. It operates based on the key module settings.

Note:

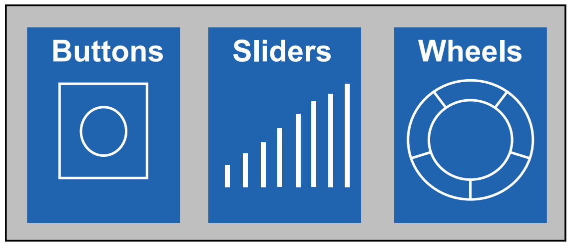
1. A lumped sensor is implemented as a combination of multiple sense lines (self-capacitance measurement) or multiple drive and sense lines (mutual capacitance measurement) to act as one single button sensor. This provides the application developer with greater flexibility in the touch sensor implementation.
2. The PTC incorporates the Adjacent Key Suppression (AKS) technology, which can be selected on a per-key basis. The AKS technology is used to suppress multiple key presses based on relative signal strength. This feature assists in solving the problem of surface moisture which can bridge a key touch to an adjacent key, causing multiple key presses.
3. This PTC subsystem supports frequency hopping, which tries to select a sampling frequency that does not clash with noise at specific frequencies elsewhere in products or product operating environments. Frequency hopping tries to hop away from the noise.
4. It is necessary to use the firmware provided by Microchip in order to use the PTC subsystem.

2. Capacitive Touch Sensors

2.1 General

Capacitive touch sensors replace conventional mechanical interfaces and operate with no mechanical wear, and are closed to the environment. They provide greater flexibility in industrial design and result in differentiating end product design.

Figure 2-1. Sensor Types



2.2 Definitions

2.2.1 Electrodes

Electrodes are typically areas of copper on a printed circuit board but can also be areas of clear conductive indium tin oxide (ITO) on a glass or plastic touchscreen.

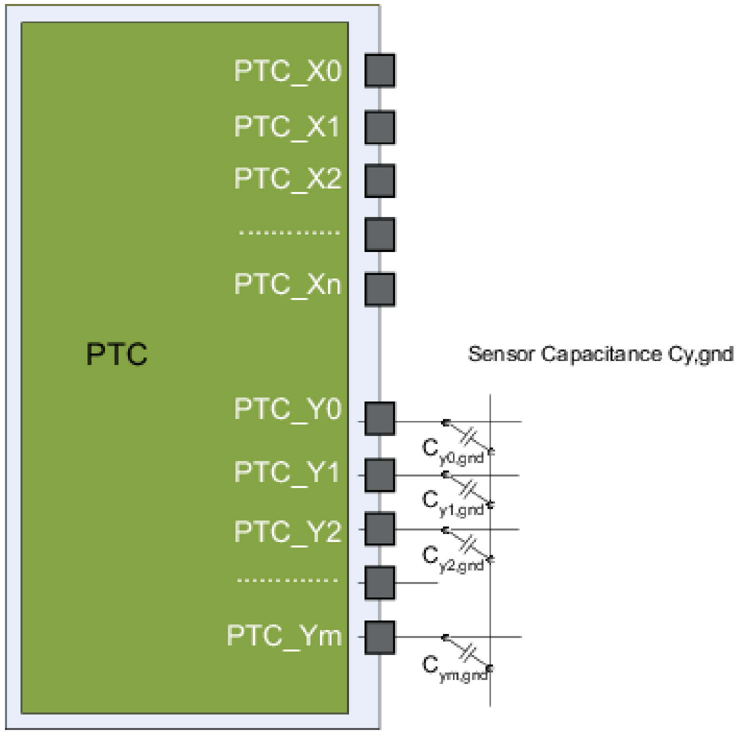
- X Line: The drive electrode (or drive line) used for mutual capacitance measurement.
- Y Line: The sense electrode (or sense line) used for mutual and self-capacitance.

The X and Y electrodes cannot be touched directly in electrical DC coupling; they must be touched in AC coupling. The electrode metal needs to be covered with an electrical insulation. If the finger touches the metal, the measurement will be erroneous and the PTC may stop to work properly.

2.2.2 Self-capacitance Sensor

A sensor with one connection to one of its parts (Y electrode). The self-capacitance from Y to earth is measured.

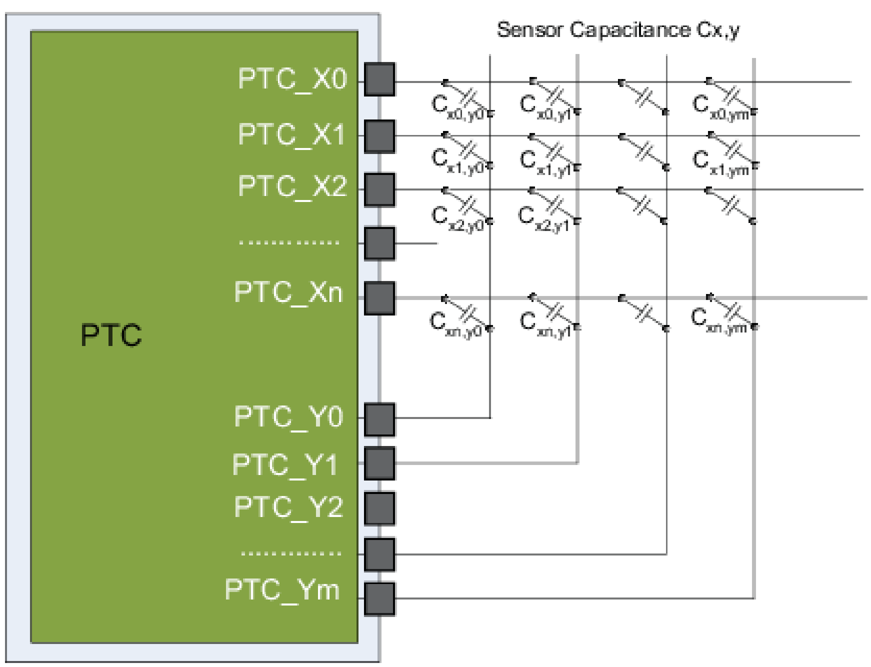
Figure 2-2. Self-capacitance Sensor



2.2.3 Mutual Capacitance Sensor

A sensor with connections to two of its parts: an X (drive) electrode and a Y (sense) electrode. The mutual capacitance from X to Y is measured.

Figure 2-3. Mutual Capacitance Sensor



2.2.4 Node or Channel

One of the capacitive measurement points at which the sensor controller can detect a capacitive change ("node" and "channel" have the same meaning).

2.2.5 Sensor

A node, or a group of nodes, used to form a touch sensor. Sensors are of four types: button, wheel, slider, surface. The touch software Key module is required by higher level touch processing modules such as a scroller and surface, in which case the 1-D and 2-D position sensors are implemented as an array of touch keys. The touch Key module manages detection of a touch contact, where higher level module(s) carry out position interpolation, contact tracking, etc.

Key or Button: a single channel which forms a single key type sensor. A zero-dimensional sensor is one that represents a single point of contact. All sensors are key-based.

Scroller: a linear sensor slider or a wheel.

A linear sensor may have any physical shape, with or without a wrap-around from the last sensor to the first. A sensor with wrap-around is configured as a ‘Wheel’, while one without is configured as a ‘Slider’. In the case of the wheel, a touch contact centered on the first key uses the last key for ‘left’ interpolation and vice-versa, while the slider option implements a dead band at the ends.

Slider: a group of channels forms a slider sensor to detect the linear position of touch.

A linear sensor utilizes the touch delta of two or more adjacent sensor nodes arranged in a row to calculate the position of a touch contact along that row. The sensor layout is designed and the threshold configured in such a way that a contact anywhere along the sensor will cause:

1. A touch delta exceeding the threshold on at least one sensor node
   The node with the strongest touch delta is determined to be the center node of the touch contact and identified as the approximate location of the touch contact.
2. Some touch delta on neighboring nodes, used for position interpolation between nodes
   The relative delta on the nodes to the left and right of the center node are used to adjust the calculated touch position towards the side with the strongest delta.

Rotor or Wheel: a one-dimensional sensor detects the linear movement of a finger during touch (that is, along a single axis). This type of sensor is a group of channels which forms a wheel sensor to detect angular positions of touch.

QTouch® Surface and Touchpads: (also known as QMatrix technology) two-dimensional sensors.

Where a linear sensor is physically implemented as a line of keys, the same approach may be extended to 2D position detection through a grid of keys. The keys are designed in such a way that interpolation may be made in either the vertical or the horizontal direction, and multiple separate touch contacts may be individually resolved in their interpolated positions.

2.2.6 Interpolated Sensor

A type of sensor that uses the shape of the electrodes to spatially interpolate the electric fields above the sensor. This technique increases the resolution on the touch sensor.

2.2.6.1 Spatially Interpolated Mutual Capacitance Slider or Wheel

Figure 2-4. Interpolated Mutual Capacitance 4-Node Slider Example

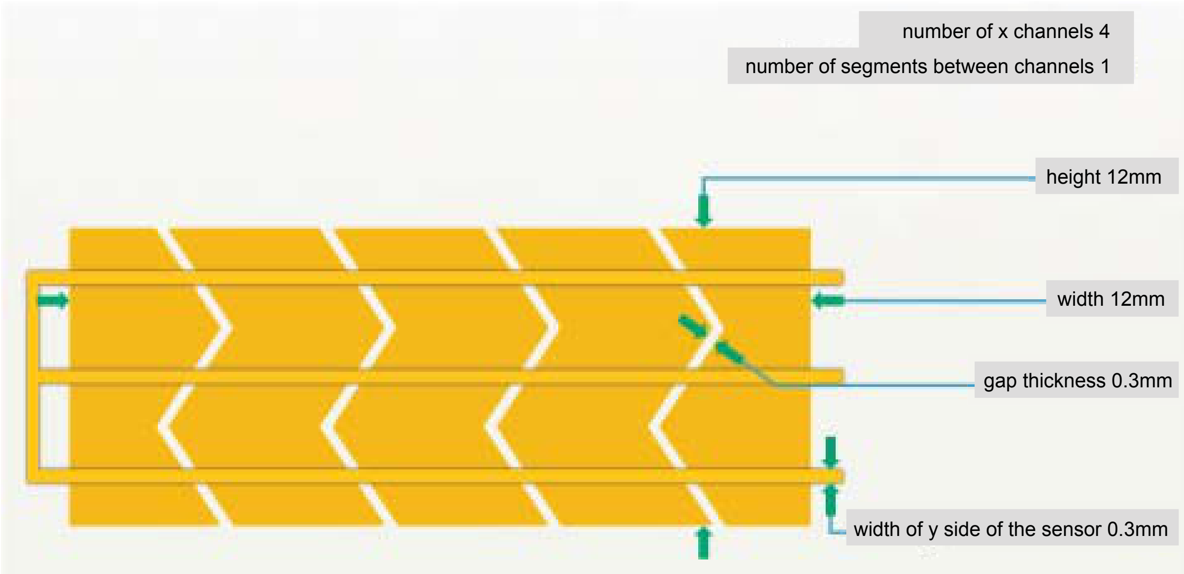
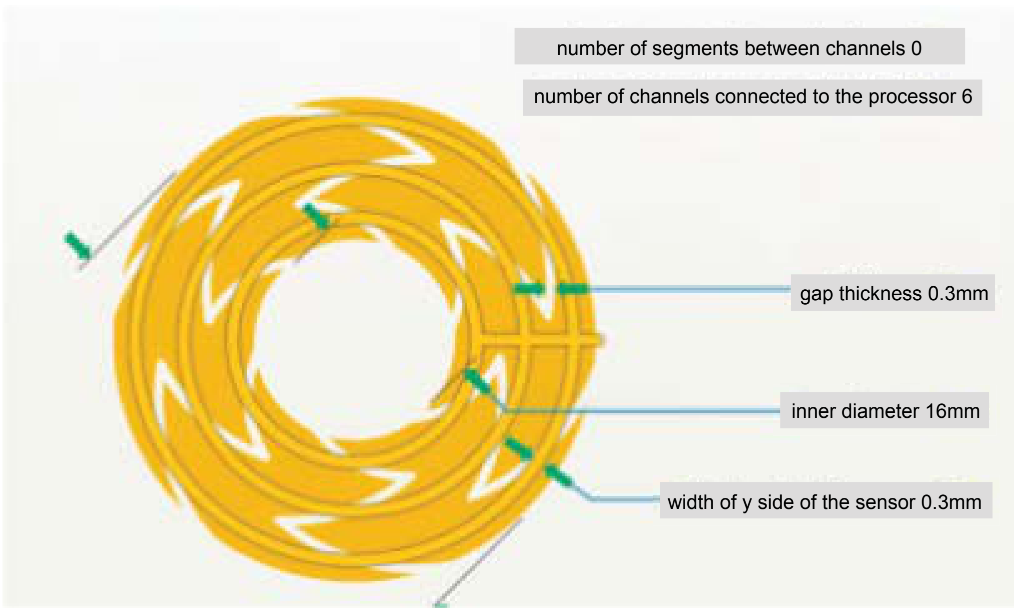


Figure 2-5. Interpolated Mutual Capacitance 3-Node Wheel Example



2.2.6.2 Spatially Interpolated Self-capacitance Slider or Wheel

Figure 2-6. Interpolated Self-capacitance 4-Node Slider Example

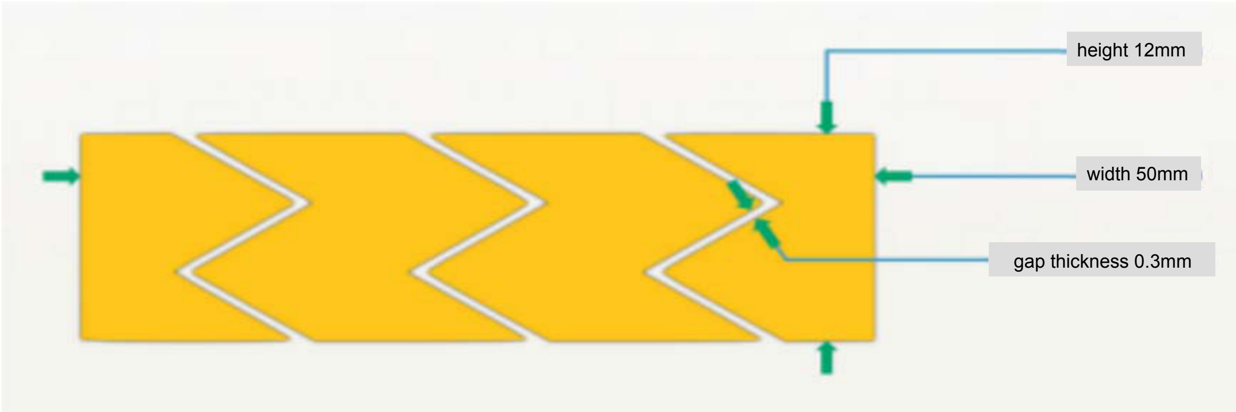
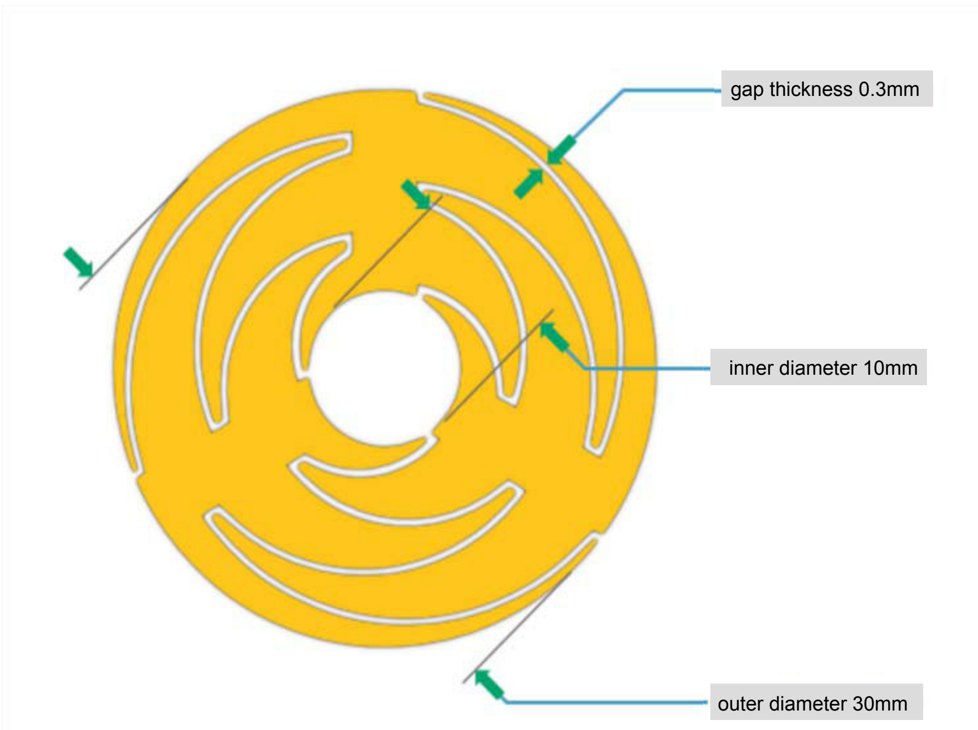


Figure 2-7. Interpolated Self-capacitance 3-Node Wheel Example



This technique is useful for increasing linearity and resolution of the slider. For a 3-node slider, we can get more than 3 positions by using the Resolution parameter up to 255 positions. The spatial interpolation will then linearize the interpolated positions.

2.2.6.3 Touch Separation

Touch separation is the minimum distance required between the edges of two fingers for the sensor to recognize them as two distinct touches. The touch separation that can be supported depends on the geometry of the sensor (this exclude the interpolated sensor).

2.2.7 Lumped Mode

PTC features a Lumped mode configuration that allows combining multiple Y-lines (self-capacitance) or multiple X- and Y-lines (mutual capacitance) to form a single sensor. This feature allows combining multiple physical sensors nodes and configures them as a single sensor called a ‘lumped sensor’. The use of Lumped mode improves power consumption and response time. In applications with a large number of keys, the sensors can be arranged in groups to form multiple lumped sensors. Scanning can be performed only on the lumped sensors. When one of the lumped sensors shows touch detection, only the keys within that lumped sensor is individually measured to determine which key is actually touched. This improves the efficiency of the system since a lesser number of measurement cycles are needed compared to scanning for all the individual keys.

- Improve the touch sensor responsiveness by reducing the number of measurements and therefore the time required for initial touch detection
- Fast position resolution by binary search
- Improved moisture rejection through ‘All but one’ key lumping in a touch button application
- Provide wake-on-touch functionality on any key (up to maximum capacitance limits) with significantly lower power consumption as only one sensor measurement is required for all keys
- Dual purpose sensor electrodes: e.g. individual keys may be lumped together to form a proximity sensor. Touch detection on a lumped sensor is implemented in the same way as a single node touch button.

Capacitive load of a lumped sensor should not exceed the maximum limit of ~30pF for both self- and mutual capacitance. The designer must select an appropriate number of keys to form a lumped sensor ensuring this limit is not exceeded. If this limit is exceeded, the firmware states a calibration error. Mutual sensors have generally smaller compensation values, so it is possible to lump more of them together, before calibration saturation.

Low-power sensor configuration only allows the use of one channel. Thus, in principle, a slider or a rotor cannot be configured as a low-power sensor since they are composed of multiple sensor channels. However, all the channels of a slider or rotor can be configured as a single lumped sensor and this can be configured as a low-power sensor.

2.2.8 Sensitivity

Sensitivity refers to the magnitude of the touch delta for a finger touch on the sensor. Sensitivity depends on the geometry of the sensor pattern and analog Gain setting used. Sensitivity increases with the delta signal amplitude and a smaller threshold.

2.2.9 Proximity Mode

Proximity is when a touch panel lights up before it is actually in contact (often associated with Lumped mode). An extension of the button is a proximity sensor. A single sensor node is monitored for a change in capacitance exceeding a preconfigured threshold. In the same way as the button, the sensor is considered to be ‘In Detect’ when that threshold is exceeded. Once in detect, a relative measurement of the contact distance is made by scaling the touch delta between two thresholds, the initial ‘Detect’ threshold and a second ‘Full Contact’ threshold.

2.2.10 Jitter

Jitter is the variation in the reported touch position when a stationary finger touch is present on the sensor. It represents the overall noise in the system. Jitter can be reduced by increasing the Filter Level setting. Jitter is the peak-to-peak variance in the reported location for an axis when a fixed touch is applied. Typically jitter is random in nature and has a Gaussian distribution; therefore measurement of peak-to-peak jitter must be conducted over some period of time, typically a few seconds. Jitter is typically measured as a percentage of the axis in question.

2.2.11 Cross-talk

Cross-talk refers to the delta change caused in the nodes adjoining a touched node. The amount of cross-talk present depends on the geometry of the sensor pattern.

2.2.12 Linearity Error

Linearity error refers to the deviation between the reported touch position and the actual touch position as the finger slides over the sensor.

2.2.13 Resolution in DPI

Resolution refers to the number of distinct touch positions reported when a finger moves in a straight line along the horizontal or vertical axis of the sensor. With a 5.08 cm long slider and a resolution parameter defined at 8 bits maximum, the resolution is 128 DPI (dots per inch, 2,54 cm).

2.2.14 Response Time

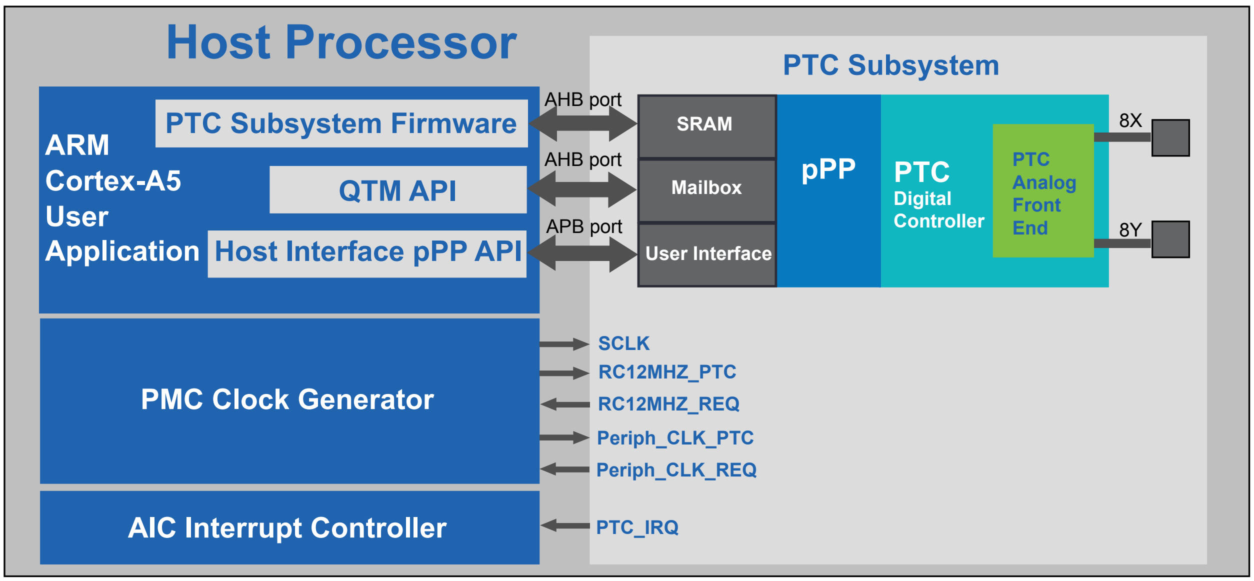
It is agreed that for a human, eight key touches per second is a maximum. Detecting one touch per ~128 ms is good enough for user interface applications. For a panel of 64 nodes, each node should be detected in 2 ms by the PTC and the microcontroller processing.

3. QTouch Technology with PTC Subsystem

3.1 Introduction

The PTC analog front end and the digital controller are not managed directly by the host processor; the host processor can be an ARM processor, or another processor. A coprocessor is introduced to manage all functionalities of the PTC subsystem. This processor, together with some peripherals, is called a Pico Power Processor (pPP). In that case a pPP program is needed, named PTC Subsystem Firmware; this program is loaded by the host processor in a shared SRAM memory area.

Figure 3-1. PTC Subsystem Block Diagram



The existing product refers to QTouch Technology as the method for capacitive sensor touch detection. The method is referenced as PTC (Peripheral Touch Controller) and embedded in the PTC subsystem.

This method involves the use of an Analog-to-Digital (ADC) converter to accurately measure the capacitance of a connected sensor and the changes in capacitance of that sensor when a touch is confirmed. The PTC is an alternative method of acquiring capacitive touch and does not require an external integrator capacitor (Cs). Integration is performed in the digital domain, and charge transfer for a single pulse is achieved using the internal sample-and-hold (Csh) capacitor.

- This enables implementation for low-power, high-sensitivity, environmentally robust capacitive touch buttons, sliders, wheels and surface
- Advantages are:
  - One pin per electrode - no external components [see note below]
- Zero drift over the temperature and supply/reference range
- No need for temperature or supply/reference compensation

Note: In high-noise environments, an external resistor may be required for filtering in Self-capacitance mode.

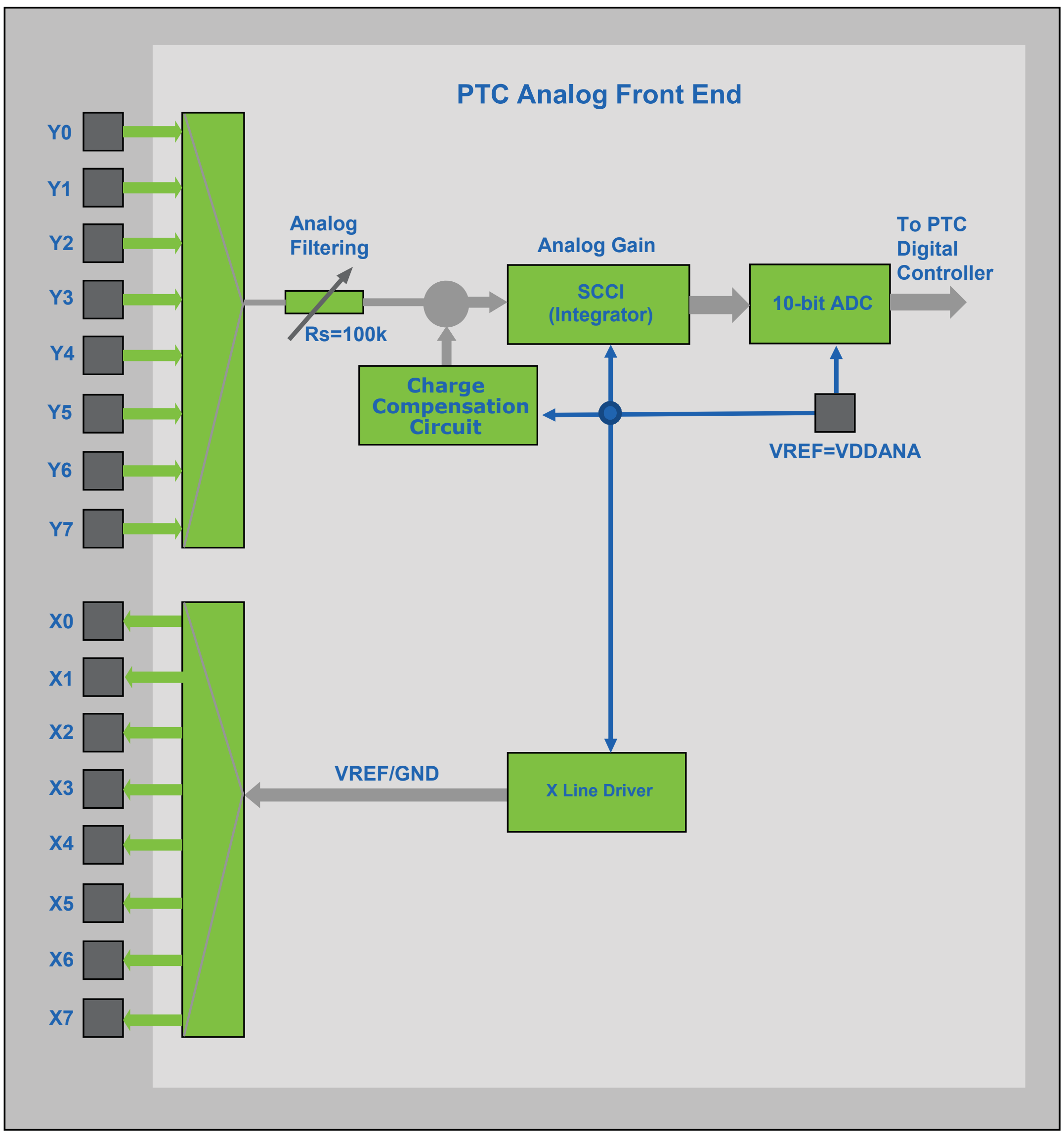
The firmware for the PTC subsystem can be used for touch sensor pin configuration, acquisition parameter setting as well as periodic sensor data capture and status update operations. The firmware interfaces with the PTC to perform the necessary action. The PTC interfaces with the external capacitive touch sensors and is capable of performing self- and mutual capacitance method measurements. The firmware features Low-power and Lumped mode configuration.

3.2 PTC Analog Front End

The analog front end consists of X-line drivers, a sensor capacitance compensation circuit and a parasitic capacitance insensitive analog Switched Capacitor Charge Integrator (SCCI). The integrator is connected to sensor Y-lines via an analog multiplexer. When the PTC digital controller is enabled, the SCCI output is automatically connected to the ADC input.

The external capacitive touch sensor is typically formed on a PCB and the sensor electrodes are connected to the Analog Charge Integrator of the PTC AFE via I/O port pins. The PTC AFE supports mutual capacitance sensors organized as capacitive touch matrices in different X-Y configurations (QTouch Surface). The PTC AFE requires one pin per X-line and one pin per Y-line. No external components are needed. The PTC AFE also supports “self-capacitance touch sensors” (QTouch). In Self-capacitance mode, the PTC AFE requires just one Y-line pin per self-capacitance sensor.

Figure 3-2. PTC Analog Front End



3.2.1 PTC Main Capacitances

The following figures show how the main capacitances are connected at a specific moment but do not describe the actual switching behavior of the measurement.

Figure 3-3. Self-capacitance Mode Block Diagram

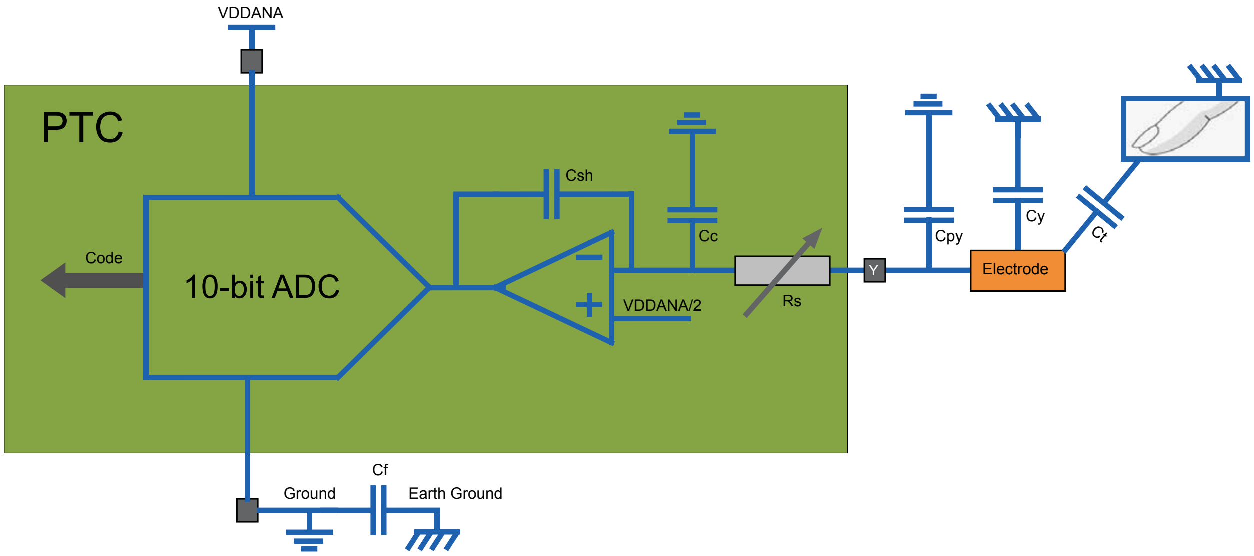
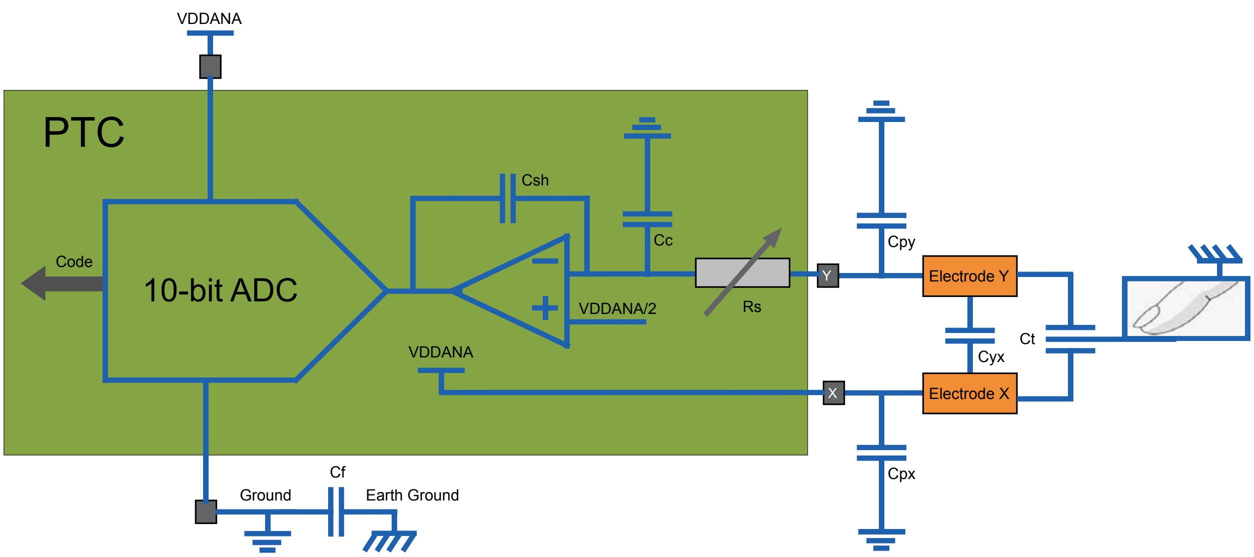


Figure 3-4. Mutual Capacitance Mode Block Diagram



3.2.2 Capacitance and Resistance Definitions

3.2.2.1 Resistance (Rs)

The Rs (programming 0, 20, 50 and 100k Ohm) is inserted is series with the electrode sensor. The Rs resistor can be tuned on to improve EMC/EMI/ESD performance of the sensor. The sensor capacitance is combined with the programmed serial resistor on the Y line (sense). This combination constitutes a hardware (Rs, Csh, Cc) first order low pass filter that can be considered as a first level of noise protection. This first level of protection enables to reduce the scale of the noise present on the acquisition signal with less impact on the acquisition time, compared to other digital filtering solutions using many samples. Nevertheless, when increasing Rs, it is necessary to tune the CSD settling time parameter to a higher value, increasing also the acquisition time.

3.2.2.2 Electrode Capacitance to Earth (Cy, Cyx)

Self-capacitance or mutual capacitance of the sense electrode (between 2 and 30pF, minimum 6x6 mm electrode) has specific design rules described in Application Note “Buttons, Sliders and Wheels Touch Sensor Design Guide” (see Reference Documents).

3.2.2.3 Compensation Capacitances (Cc)

The PTC has an internal compensation circuit which is used to compensate the sensor capacitance. Both Self-capacitance and Mutual Capacitance sensing modes have the same compensation range. However, Mutual Capacitance mode can compensate more parasitic capacitance compared to Self-capacitance mode. The calibration Cc is adjusted to match the sensor load up to 30pF.

In Self-capacitance mode, the compensation capacitor equals the external sensor plus parasitic capacitances (Cy +Cp). In Mutual Capacitance mode, the compensation capacitor only equals the sensor capacitance (Cyx). The parasitic capacitances (Cpy) are compensated by the virtual earth input of the integrator.

This calibration process is automated by the PTC digital controller and just needs an Enable signal from the firmware to be performed.

Typical compensation circuit value for Self-capacitance mode ranges from 10 to 25 pF. For Mutual Capacitance mode, this value is around 2 to 7 pF.

The compensation circuit value is affected by sensor size and the ground surrounding the sensor or trace. The compensation circuit value ranges from 0.007 pF to 30 pF.

If the compensation circuit value exceeds the limit, to reduce the value, use a mesh instead of a solid plane in the sensor and ground plane.

Refer to Application Note “Buttons, Sliders and Wheels Sensor Design Guide” (see Reference Documents).

3.2.2.4 Touch Capacitance to Earth (Ct)

Ct is caused by a finger touch over the Sense Electrode (about 1 pF). This capacitance modifies the value of the Cy+Cpy electrode capacitance.

3.2.2.5 Parasitic Capacitances (Cpx, Cpy)

Cpy and Cpx are from board routing and any plane at proximity. In Self-capacitance mode, this capacitance is added to the Cy value and is calibrated by the Cc capacitance. The parasitic capacitance on Cx does not matter because it is driven by a supply voltage or ground.

3.2.2.6 Earth-to-Ground Coupling (Cf)

The coupling capacitance between circuit ground and earth ground: (Cf >> Cy & Ct)

3.2.3 Principle of Operation

The self-capacitance measurement method involves charging a sense electrode of unknown capacitance to a known potential. The resulting charge is transferred into a measurement circuit. By measuring the charge with one or more charge-and-transfer cycles, the capacitance of the sense plate can be determined.

The mutual capacitance measurement method uses a pair of sensing electrodes. One electrode acts as an emitter into which a charge consisting of logic pulses is driven in Burst mode. The other electrode acts as a receiver that couples to the emitter using the overlying panel dielectric. When a finger touches the panel, the field coupling is reduced, and touch is detected.

3.2.3.1 Acquisition Operation

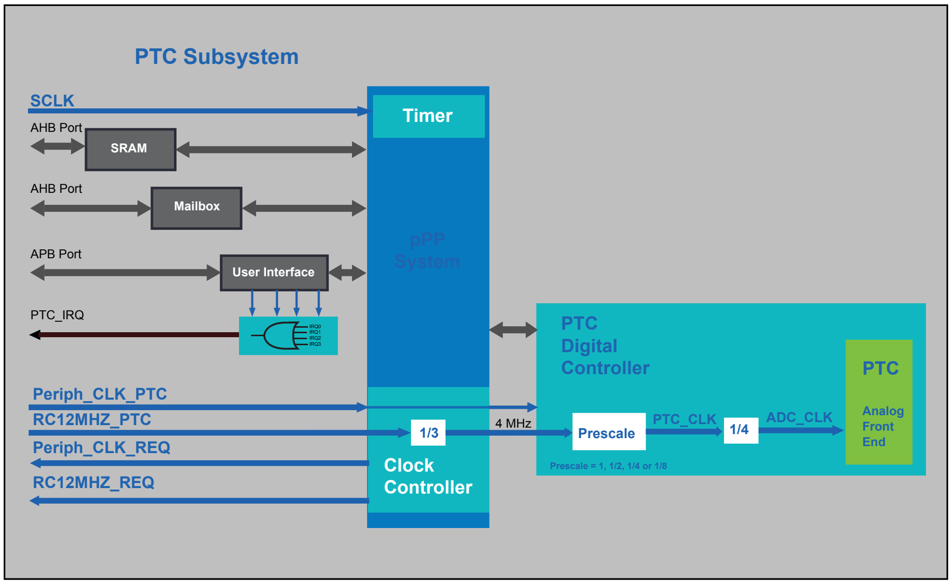
The PTC method is a single pin acquisition method.

First, the key being measured is fully charged with a voltage pulse and this charge is shared with the compensation Cc capacitance. A touch event will unbalance this charge sharing.

Secondly, the sensor charge is transferred to the internal sample and holds Csh. The voltage which remains on the Csh is then measured by the ADC. A touch on the sensor will contribute to the capacitance of the sensor and will create changes in the voltages measured for each of the pulses.

In a mutual capacitance measurement, the principle is the same but the capacitance is between two electrodes.

Figure 3-5. PTC Subsystem Clock Schematic



The RC12HMZ clock is internally divided by 3 in the PTC subsystem, and so a 4 MHz clock is provided to the PTC digital controller. This controller can divide the clock further by 1, 2, 4 or 8 to slow down the PTC clock. The prescaled clock PTC_CLK is divided by 4 to supply an ADC_CLK to the PTC analog front end. The ADC data rate is defined by the controller. The typical value is about 33 kHz to 66 kHz depending on the timing configuration.

The 10-bit ADC output code is supposed to be ideally at 512 (without digital gain) if no touch occurs in self- or mutual capacitance measurement. One ADC single conversion uses 26 ADC_CLK cycles. With a PTC_CLK/4 = 1 MHz, the result is approximately 38 ksps.

In mutual capacitance measurement, the Ct touch capacitance decreases the sensor capacitance. The voltage in Csh decreases and the ADC code exceeds 512. The delta is then positive.

In self-capacitance measurement, the mechanism is similar, plus a sign correction to see an always-positive delta.

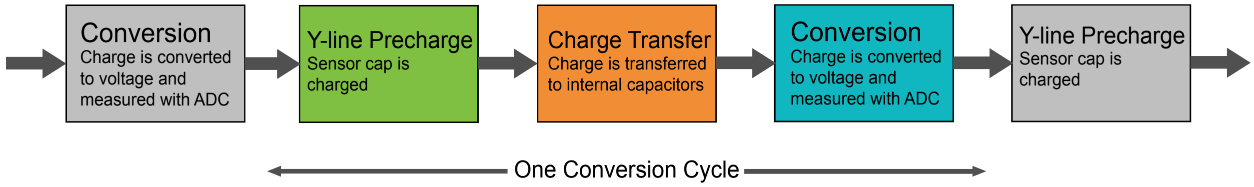
3.2.3.2 Analog Gain

The gain of the analog integrator is adjustable by modifying the integrator capacitance Csh. The gain is a ratio of capacitance Ct/Csh, there is no decimation of the data rate frequency.

3.2.3.3 Acquisition Time

The acquisition time depends on the operation performed during the conversion cycle.

Figure 3-6. Conversion Cycle



The timing of the PTC acquisition depends on the 12 MHz RC clock frequency divided by 3 in the pPP and the following prescale “prsc”. The final clocks used are:

- 4 MHz divided by 4 for fixed and not programmable delays
- ADC_Clock=4 MHz divided by 4xPrsc for programmable delays

ADC_clock defines a time base period at Tb=1 µs, 2 µs, 4 µs or 8 µs. It is recommended to keep the ADC_clock at the fastest value at 1 MHz (Tb=1 µs). This can be extended by adding up to 255 CSD PTC cycles, usually at CSD=3 to 10. The fasted convertion speed for one channel is 35 µs.

The timing definition when using more than one conversion cycle is an acquisition time. This depends on the configured filtering levels and the different software algorithms used to eliminates the noise.

The number of conversion cycles for a reliable PTC measurement can be increased by the filtering level up to 64 conversion cycles.

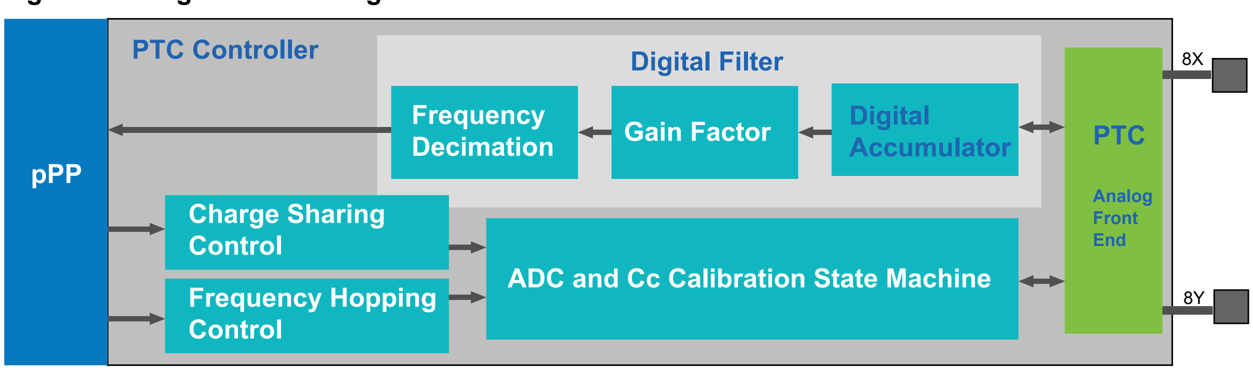
Other parameters have a significant influence on the timing, like detection integration and drift and recalibration control.

3.3 PTC Digital Controller

A part of the signal processing is handled by a hardware controller controlled by the pPP firmware.

The PTC digital controller is a peripheral of the pPP. It is intended for acquiring capacitive touch sensor and capacitive proximity sensor signals under limited firmware control by the controlling processor. The PTC digital controller consists of an Analog Charge Integrator and a 10-bit ADC Controller, 16-bit Digital Accumulator for the ADC results and a State Machine taking care of sensor sampling and digital accumulation sequence.

Figure 3-7. Signal Processing and Calibration Block



3.3.1 PTC Digital Controller Operations

- Control of the ADC 10-bit SAR state machine single ADC conversion or Free Run mode (comparator and ADC data/accumulator register)
- Digital gain up to 32 and averaging up to 64 ADC codes
- Selection of the filtering resistance (0, 20, 50 or 100 kΩ)

- Adjustment of the compensation capacitor up to 30 pF
- Adjustment of the integration capacitor up to 30 pF
- Frequency hopping[(1)] implementation (modification of the sampling rate to avoid synchronous parasitic noise)
- Channel Share Delay Selection CSD[(2)] (settling time)
- Prescaling (1, 1/2, 1/4, 1/8), 4 MHz down to ADC_CLK

Note:

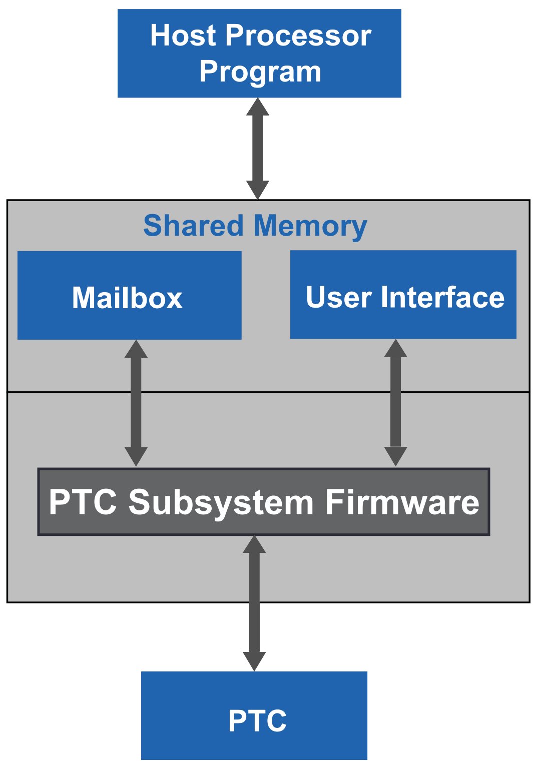
1. A programmable sampling delay can be used to choose (modify) the sampling frequency that is best suited in an application where other periodic noise sources may otherwise disturb the sampling. Frequency hopping can also be modified automatically from one sampling cycle to another, by setting the software driver parameters.
2. CSD bits define the delay when changing input channels. The delay allows the analog circuits to settle on a new (Y) channel or channel pair (X-Y). The delay is application-dependent, and therefore this option enables the user to select a suitable delay. The delay is expressed as a number of ADC clock cycles.

4. Firmware and QTouch Modular Libraries (QTML)

4.1 The Firmware

The firmware manages all operations with pPP and the PTC subsystem. The firmware is a binary file copied to the SRAM code area at the address defined by the memory map. In addition to the PTC subsystem, the firmware manages pPP onboard timers, and communications with the host processor. pPP uses QTML (QTouch Modular Library) to manage the PTC system. QTML consists of a number of preconfigured libraries which handle specific touch functionalities. Application developers cannot modify the libraries; they can configure the PTC touch system and receive touch-related data. Configuration information is written by the host processor to shared memory. The pPP then reads the information and configures the PTC. Once configured and when PTC is measuring touch data, the pPP updates the shared memory with touch data.

Figure 4-1. Firmware Block Diagram



For efficient programming of the PTC technology, it is necessary to understand all the terms and definitions used in the header files and driver provided by the softpack. This document uses definitions provided in the component header files ppp.h and qtm.h. The Host program, not provided by the softpack, is an end-user application development using the component and driver header file.

The structures are organized in individual or group configurations and data for node, key and scroller (slider and wheel). See "Configuration objects" and "Data objects" lists below.

- Configuration objects:
  - Node Configuration

  - Node Group Configuration
  - Key Group Configuration
  - Key Configuration
  - Scroller Group Configuration
  - Scroller Configuration
  - Auto Scan Configuration
  - Frequency Hopping Auto-tune Configuration
  - Frequency Hopping Frequencies
- Data objects:
  - Node Data
  - Key Data
  - Key Group Data
  - Scroller Data
  - Frequency Hopping Auto-tune Data
  - Touch Events

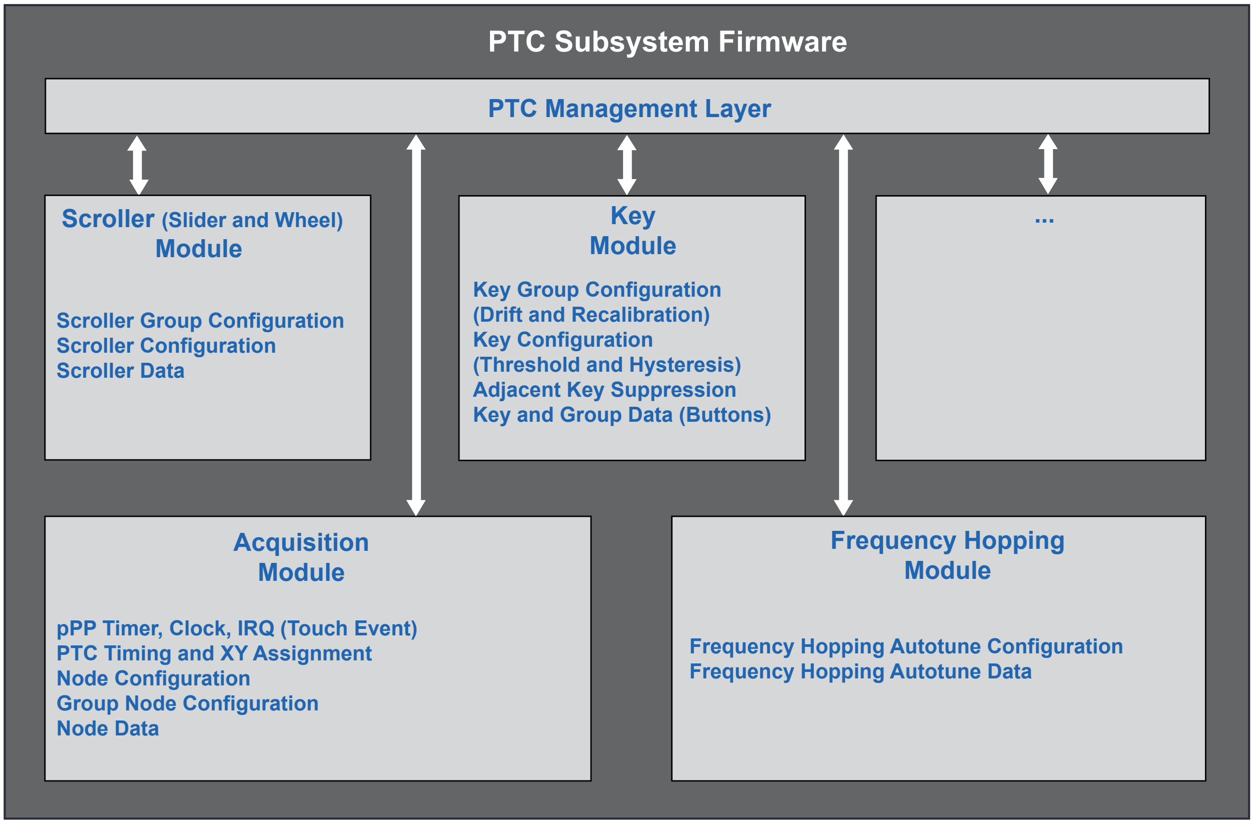
4.2 QTouch Modular Library

QTouch Modular Library provides the touch sensing functionality of QTouch Library under a modular architecture. User options for each module are configured in the Host application code, and shared with the library module through the mailbox SRAM area. Configurations may be modified on-the-fly by application code in between measurement sweeps of the touch sensors.

List of modules:

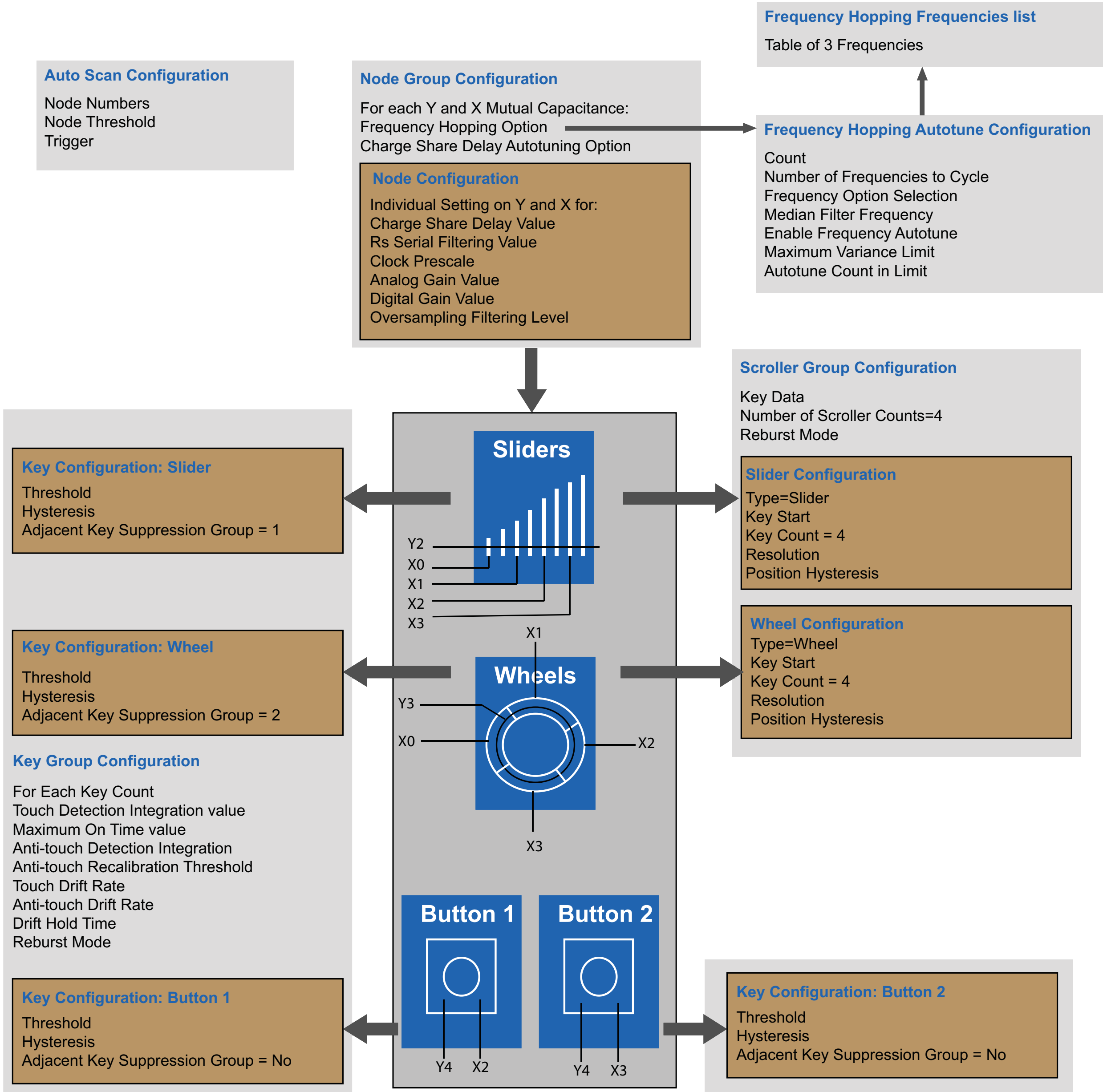
- The acquisition module (node definitions, pPP and PTC management) is product-dependent. It implements all hardware-dependent operations for configuration and measurement of capacitive touch or proximity sensors.
- The signal conditioning (frequency hopping) module applies algorithmic and feedback control methods to improve the quality of the measurement data captured by the acquisition module.
- The post-processing module (Key, Scroller) interprets measurement data as keys or scrollers.
- The scroller module manages slider and wheel functionality. It uses the output from the Key processing module as a data source for slider/wheel calculations.

Figure 4-2. PTC Subsystem Firmware Block Diagram



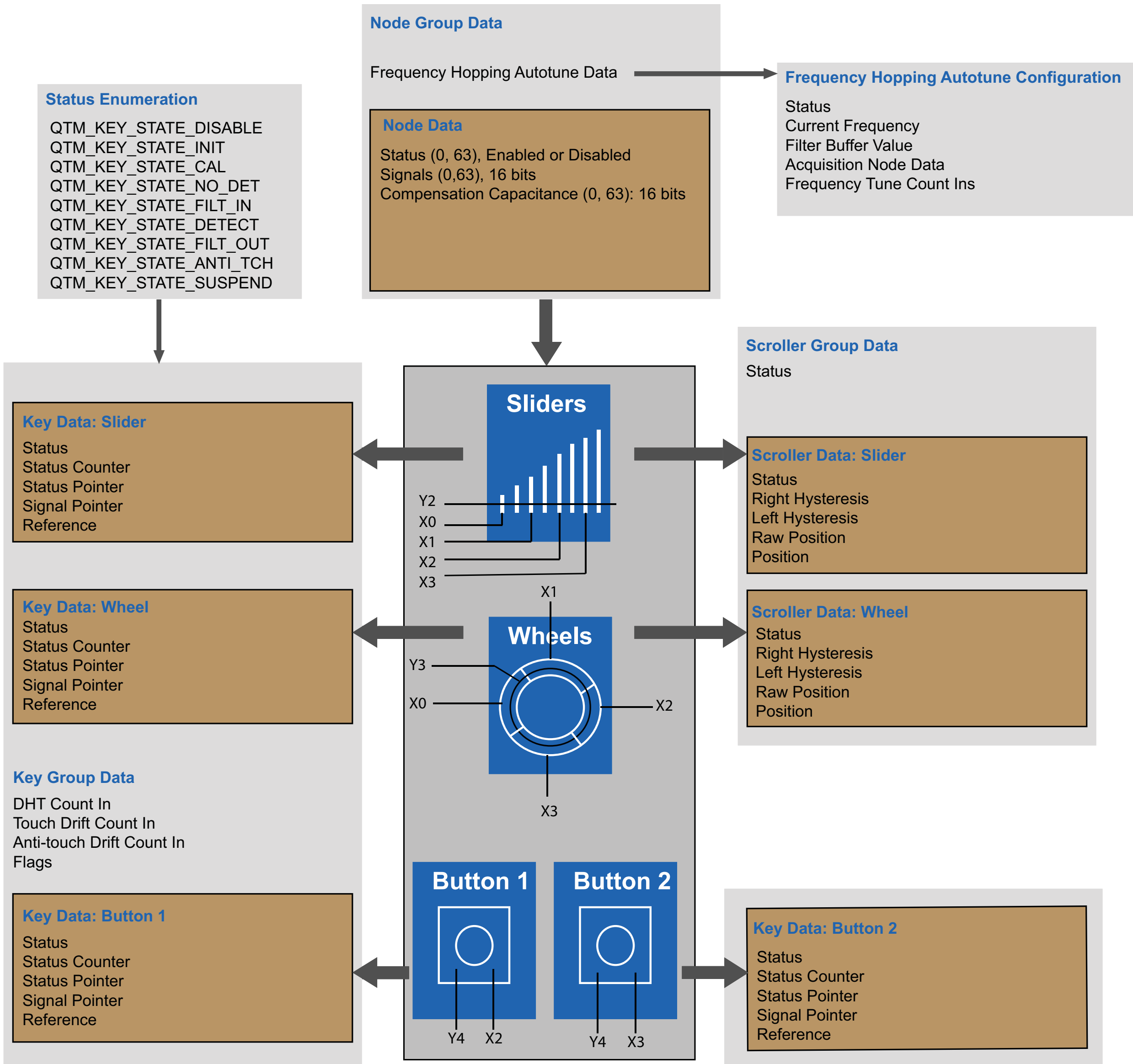
4.3 Configuration Parameters Example

Figure 4-3. Configuration Parameters Example for QT1 Mutual Capacitance



4.4 Data Parameters Example

Figure 4-4. Data Parameters Example for QT1 Mutual Capacitance



4.5 Mailbox Mapping

SAMA5D2 Firmware Memory Map

SRAM Base address: PPP_ADDR=0x00800000

SRAM length: PPP_SRAM_LEN=0x4000

Mailbox base address: PPP_MAILBOX_ADDR=0x00804000

Note: All registers are 8-bit registers. R/W stands for Read/Write.

Table 4-1. Command and Node Configuration and Data

					7	6	5	4	3	2	1	0
Offset	**Structure Name**	**Field**	**Bits**	**Access**	**Register**							
0x0000	cmd	id	7:0	R/W	id							
0x0001			15:8	R/W	-	-	-	-	-	-	-	-
0x0002		addr	7:0	R/W	addr							
0x0003			15:8	R/W	addr							
0x0004		data	7:0	R/W	Data							
0x0005			15:8	R/W	Data							
0x0006			23:16	R/W	Data							
0x0007			31:24	R/W	Data							
0x0100	node_group_config	count	7:0	R/W	-	-	count					
0x0101			15:8	R/W	-	-	-	-	-	-	-	-
0x0102		ptc_type	7:0	R/W	-	self	mutual	-	-	-	-	-
0x0103		freq_option	7:0	R/W	-	-	-	spread	freq			
0x0104		calib_option	7:0	R/W	-	-	-	-	calib	tau		
0x0105		unused	7:0	R/W	-	-	-	-	-	-	-	-
0x0106	node_config[64]	mask_x	7:0	R/W	X7	X6	X5	X4	X3	X2	X1	X0
0x0107			15:8	R/W	-	-	-	-	-	-	-	-
0x0108		mask_y	7:0	R/W	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
0x0109			15:8	R/W	-	-	-	-	-	-	-	-
0x010A			23:16	R/W	-	-	-	-	-	-	-	-
0x010B			31:24	R/W	-	-	-	-	-	-	-	-
0x010C		csd	7:0	R/W	csd							
0x010D		rsel	7:0	R/W	-	-	-	-	-	-	rsel	
0x010E		prsc	7:0	R/W	-	-	-	-	-	-	prsc	

					7	6	5	4	3	2	1	0
Offset	**Structure Name**	**Field**	**Bits**	**Access**	**Register**							
0x010F		gain_analog	7:0	R/W	-	-	-	-	-	analog gain		
0x0110		gain_digital	7:0	R/W	-	-	-	-	-	digital gain		
0x0111		oversampling	7:0	R/W	-	-	-	-	-	oversampling		
0x0406	node_data[64]	status	7:0	R	-	-	-	-	status			
0x0407		unused	7:0	R	-	-	-	-	-	-	-	-
0x0408		signals	7:0	R	signal							
0x0409			15:8	R	signal							
0x040A		comp_caps	7:0	R	C1	C0						
0x040B			15:8	R	C3	C2						

Table 4-2. Key Configuration and Data

Offset	Structure Name	Field	Bits	Access	Register							
0x0586	key_group_config	count	7:0	R/W	-	-	count					
0x0587			15:8	R/W	-	-	-	-	-	-	-	-
0x0588		touch_di	7:0	R/W	touch_di							
0x0589		max_on_time	7:0	R/W	max_on_time							
0x058A		anti_touch_di	7:0	R/W	anti_touch_di							
0x058B		anti_touch_recal_thr	7:0	R/W	-	-	-	-	-	anti_touch_recal_thr		
0x058C		touch_drift_rate	7:0	R/W	touch_drift_rate							
0x058D		anti_touch_drift_rate	7:0	R/W	anti_touch_drift_rate							
0x058E		drift_hold_time	7:0	R/W	drift_hold_time							
0x058F		reburst_mode	7:0	R/W	-	-	-	-	-	-	reburst_mode	
0x0590	key_config [64]	threshold	7:0	R/W	threshold							
0x0591		hysteresis	7:0	R/W	-	-	-	-	-	-	hysteresis	
0x0592		aks_group	7:0	R/W	-	-	-	-	-	aks_group		
0x0593		unused	7:0	R/W	-	-	-	-	-	-	-	-
0x0690	key_data [64]	status	7:0	R	-	-	-	-	status			
0x0691		status_counter	7:0	R	?	?	?	?	?	?	?	?
0x0692		node_struct_ptr	7:0	R	node_struct_ptr							
0x0693			15:8	R	node_struct_ptr							
0x0694		reference	7:0	R	reference							
0x0695			15:8	R	reference							

Table 4-3. Scroller Configuration and Data

Offset	Structure Name	Field	Bits	Access	Register							
0x0810	auto_scan_config	unused	7:0	R/W	-	-	-	-	-	-	-	-
0x0811			15:8	R/W	-	-	-	-	-	-	-	-
0x0812		node_number	7:0	R/W	-	-	number					
0x0813			15:8	R/W	-	-	-	-	-	-	-	-
0x0814		node_threshold	7:0	R/W	threshold							
0x0815		trigger	7:0	R/W	-	-	-	-	trigger			
0x0816	scroller_group_config	key_data	7:0	R/W	key_data							
0x0817			15:8	R/W	key_data							
0x0818		count	7:0	R/W	-	-	-	-	-	-	count	
0x0819		Unused	7:0	R/W	-	-	-	-	-	-	-	-
0x081A	scroller_config [4]	type	7:0	R/W	-	-	-	-	-	-	-	type
0x081B		unused	7:0	R/W	-	-	-	-	-	-	-	-
0x081C		key_start	7:0	R/W	-	-	key_start					
0x081D			15:8	R/W	-	-	-	-	-	-	-	-
0x081E		key_count	7:0	R/W	-	-	key_count					
0x081F		resolution_deadband	7:0	R/W	resolution	deadband						
0x0820		position_hysteresis	7:0	R/W	position_hysteresis							
0x0821		unused	7:0	R/W	-	-	-	-	-	-	-	-
0x0822		contact_min_threshold	7:0	R/W	-	-	-	-	-	-	-	-
0x0823			15:8	R/W	-	-	-	-	-	-	-	-
0x0842	scroller_data [4]	status	7:0	R	-	-	-	-	status			
0x0843		right_hyst	7:0	R	-							
0x0844		left_hyst	7:0	R	-							
0x0845		unused	7:0	R	-	-	-	-	-	-	-	

Offset	Structure Name	Field	Bits	Access	Register							
0x0846		raw_position	7:0	R	-	-	-	-	-	-	-	-
0x0847			15:8	R	-	-	-	-	-	-	-	-
0x0848		position	7:0	R	-							
0x0849			15:8	R								
0x084A		contact_size	7:0	R								
0x084B			15:8	R	-							

Table 4-4. Frequency Hopping Configuration and Data

Offset	Structure Name	Field	Bits	Access	Register							
0x086A	fh_autotune_config	count	7:0	R/W	-	-	-	-	-	-	-	-
0x086B		num_freqs	7:0	R/W	-	-	-	-	-	-	num_freqs	
0x086C		freq_option_select	7:0	R/W	-	-	-	-	-	-	-	-
0x086D			15:8	R/W	-	-	-	-	-	-	-	-
0x086E		median_filter_freq	7:0	R/W	-	-	-	-	freq			
0x086F			15:8	R/W	-	-	-	-	-	-	-	-
0x0870		enable_freq_autotune	7:0	R/W	-	-	-	-	-	-	-	enable
0x0871		max_variance_limit	7:0	R/W	max_variance_limit							
0x0872		auto_tune_count_in _limit	7:0	R/W	autotune_count_in_limit							
0x0873		unused	7:0	R/W	-	-	-	-	-	-	-	-
0x0874	fh_autotune_data	status	7:0	R	reburst	-	-	-	-	-	position change	touch active
0x0875		current_freq	7:0	R	-	-	-	-	freq			
0x0876		filter_buffer	7:0	R	filter_buffer							
0x0877			15:8	R	filter_buffer							
0x0878		acq_node_data	7:0	R	signal							
0x0879			15:8	R	signal							
0x087A		freq_tune_count_ins	7:0	R	-	-	-	-	-	-	-	-
0x087B			15:8	R	-	-	-	-	-	-	-	-
0x087C	fh_freq	freq0	7:0	R/W	-	-	-	-	freq0			
0x087D		freq1	7:0	R/W	-	-	-	-	freq1			
0x087E		freq2	7:0	R/W	-	-	-	-	freq2			
0x087F		unused	7:0	R	-	-	-	-	-	-	-	-

Table 4-5. IRQs

Offset	Structure Name	Field	Bits	Access	Register							
0x0880	key_event_id[8]	event	7:0	R	event7	event6	event5	event4	event3	event2	event1	event0
0x0888	key_event_state[8]	status	7:0	R	state7	state6	state5	state4	state3	state2	state1	state0
0x088A	scroller_event_id[4]	event	7:0	R	event7	event6	event5	event4	event3	event2	event1	event0
0x088E	scroller_event_state[4]	status	7:0	R	state7	state6	state5	state4	state3	state2	state1	state0

4.6 Firmware Initialization

When the pPP is started by filling the Command register with RUN value, the firmware loads the configuration and parameters supplied by the mailbox. Then the firmware performs a first calibration of the sensor nodes. This establishes the first values of reference for all nodes used by the key detection process. This reference can be readjusted later depending on the Drift and Recalibration parameters conditions.

Table 4-6. Command ID Values

Command	Decimal Value
QTM_CMD_FIRM_VERSION	8
QTM_CMD_DEEPSLEEP	16
QTM_CMD_ACQ	17
QTM_CMD_INIT	18
QTM_CMD_RUN	19
QTM_CMD_STATUS	20
QTM_CMD_STOP	21
QTM_CMD_UPDATE_TOUCH_CFG	22
QTM_CMD_SET_ACQ_MODE_ON_DEMAND	23
QTM_CMD_SET_ACQ_MODE_TIMER	24
QTM_CMD_SET_ACQ_MODE_WCOMP	25
QTM_CMD_SET_TIMER_INTERVAL	26
QTM_CMD_STOP_TIMER	27
QTM_CMD_START_TIMER	28
QTM_CMD_RESET	29

4.7 Acquisition Module

This section describes the various parameters that can be used to describe the sensor's performance.

4.7.1 Node Configuration

4.7.1.1 mask_X or mask_Y

Each field of the mask_X and mask_Y registers can be 1 or 0 to individually activate the X (drive) and the Y (sense). The measurement sensing will be performed on the activated Y in self-capacitance and in selected XY in mutual capacitance configuration. When activated, the X is driven synchronously with the Y. Up to 64 measurement combinations are available. Multiple X drives can be combined as a single X drive. Also, multiple Y can be combined as a single Y. This process is called Lumped mode and is managed in software configuration. Care should be taken with lumped Y combinations so as not to exceed internal compensation requirements (see Calibration using Capacitance Compensation).

4.7.1.2 Charge Share Delay (CSD)

CSD is the number of delay cycles of the PTC clock to ensure charging of sensor node capacitance charges between Cc and Cy through Rs. CSD should be increased incrementally until the signal remains stable from one setting to the next. This is the optimal signal level and indicates that the charge transfer is complete. Programmable values range from 0 to 255 ADC_clock cycles (ADC clock is PTC_clock/4/ Prescale=1MHz/Prescale, with programmable prescale 1, 2, 4, 8).

Increasing the prescaler has the effect of extending the charge time, too. So the usable range may be 4 to 16 or 4 to 32.

Each Yn channel can have a different value.

4.7.1.3 Resistance Selection (Rsel)

Rsel is the field used to program the Rs resistance in series with the Y sense input. Four values of resistance can be programmed. In Self-capacitance mode, an external resistor is recommended as the internal RSel is not effective.

- 0 for 0 Ohm
- 1 for 20 kOhm
- 2 for 50 kOhm
- 3 for 100 kOhm

4.7.1.4 Prescale (Prsc)

The PTC digital controller embeds a clock prescale to divide the 4 MHz input clock to generate the PTC_clock.

This prescale can be programmed using field prsc as follows:

- 0 for no division
- 1 for division by 2
- 2 for division by 4
- 3 for division by 8

4.7.1.5 Analog Gain

PTC gain possibilities are 1, 2, 4, 8 and 16, usable for both analog gain and digital gain.

The Analog Gain field can be programmed from 0 to 4:

- 0 for gain by 1 (recommended default and usage)
- 1 for gain by 2
- 2 for gain by 4
- 3 for gain by 8
- 4 for gain by 16

4.7.1.6 Digital Gain

Digital gain is a code in a register that is accumulated by the number of gain values 1 to 16. There is a decimation in frequency data rate by the gain factor (number of samples needed in the accumulation). The accumulated sum is scaled to digital gain.

The Digital Gain field can be programmed from 0 to 4:

- 0 for no gain
- 1 for gain by 2
- 2 for gain by 4
- 3 for gain by 8
- 4 for gain by 16
- 5 for gain by 32

4.7.1.7 Oversampling

Oversampling is the number of samples to accumulate for each measurement. PTC filter levels are 1, 2, 4, 8, 16, 32 and 64. This is a digital averaging of the ADC output code. It is the same operation performed by the digital gain accumulator; this is why the PTC Filter Level parameter needs to be bigger than or equal to the digital gain value.

The oversampling field can be programmed from 0 to 6:

- 0 for no filter
- 1 for filter level by 2
- 2 for filter level by 4
- 3 for filter level by 8
- 4 for filter level by 16
- 5 for filter level by 32
- 6 for filter level by 64

4.7.2 Node Group Configuration

4.7.2.1 num_sensor_nodes (Count)

Count is the number of nodes defined in the node configuration array.

The different combinations can be lumped, or they may use a different parameter setup defined by the application developer. This field is 6 bits and so the number of possibilities is 64.

4.7.2.2 acq_sensor_type (Ptc type)

If the node group is self-cap, then set 1 in the field and 0 in the Mutual Capacitance field.

If the node group is mutual-cap, then set 1 in the field and 0 in the Self-capacitance field.

The node group cannot be both self and mutual.

4.7.2.3 Freq Option

- Freq_option = PTC_FREQ_SEL_n, n=0 to 15

Preset to one specific frequency (with frequency hopping disabled) away from the noisy frequency.

This option inserts a delay cycle between measurements, where 0 is the shortest delay, and 15 the longest.

The PTC acquisition frequency is dependent on the generic clock input to PTC and the PTC clock prescale setting. This delay setting inserts “n” PTC clock cycles between consecutive measurements on a given sensor, thereby changing the PTC acquisition frequency. The FREQ_FREQ_SEL_1 setting inserts 1 ADC clock cycle between consecutive measurements. The FREQ_FREQ_SEL_14 setting inserts 14 ADC clock cycles. Hence, a higher delay setting increases the total time taken for capacitance measurement on a given sensor as compared to a lower delay setting. A desired setting can be used to avoid noise around the same frequency as the acquisition frequency.

- Freq_option = PTC_FREQ_SEL_SPREAD

Spread varies this delay from 0 to 15 in a sawtooth manner during the oversampling.

The delay is varied from 0 to 15 in a sawtooth manner on successive samples during oversampling to apply a wider spectrum of sampling frequency. Compared to single frequency acquisition, the frequency spread option reduces the sensitivity to noise at a particular ‘worst case’ frequency, but increases the range of noise frequencies around that worst case frequency which will show harmonic interference.

4.7.2.4 Calib Option

The Calib field offers 4 possibilities:

- 0 for no automatic tuning
- 1 for Rs automatic tuning
- 2 for prescale automatic tuning
- 3 for CSD automatic tuning

The Tau field is the convergence criterion of automatic tuning. Tau is the first order RC filter value.

- 0 for 2 Tau convergence
- 1 for 3 Tau convergence
- 2 for 4 Tau convergence
- 3 for 5 Tau convergence

Rs Automatic Tuning

The clock prescale and CSD are maintained at the configured setting, while the internal series resistor is adjusted to the maximum value which allows adequate charging for each sensor node. This feature is only available with PTC mutual capacitance acquisition. Programmable values are 0k, 20k, 50k, 100k.

Prescale Automatic Tuning

The series resistor and CSD are maintained at the configured setting, while the prescale is adjusted to the minimum value which allows adequate charging for each sensor node.

CSD Automatic Tuning

Both Prescale and Resistor are maintained at the configured setting, Charge Share Delay is adjusted to the minimum value which allows adequate charging for each sensor node. Incrementing CSD adds one cycle to the charge transfer phase of the acquisition sequence

Tau Calibration Target

Target used by the automatic tuning. It applies a limit to the charge transfer loss allowed, where a higher target setting ensures a greater proportion of full charge is transferred. This is evaluated as numbers (2 to 5) of TAU formed by Rs, Cc, Cy first order network. Three exclusive automatic tunings are available when selected.

4.7.3 Node Data

4.7.3.1 Status Node, Key and Scroller Data

The sensor touch status is the primary touch sensor information utilized by a user application. The sensor state can be Enabled or Disabled.

4.7.3.2 Signal Node Data

The signal is the level of most recent measured node, whether a touch occurs or not. The value of the signal is typically between 0 to 1023 and (0 to 1023) x (Digital Gain) when the gain is applied. The signal is the result of the ADC conversion, gain and digital filtering by accumulation. When there is no touch, the value is at mid-rail (512) or digital gain x 512 because comp_cap (compensation capacitance) has matched the sensor capacitance and the PTC is resting at mid-rail.

4.7.3.3 How to Compute the Cc Capacitance from Node Data comp_caps

Table 4-7. Node Data Parameter: comp_caps Code Register

Field	Bits	Access	Register	
comp_caps	7:0	R	C1	C0
	15:8	R	C3	C2

Compensation circuit value used in pF =

([comp_caps] & 0x0F)*0.007 +

([comp_caps] >> 4) & 0x0F)*0.07 +

([comp_caps] >> 8) & 0x0F)*0.7 +

([comp_caps]>> 12) & 0x3) * 7

Simplified as:

Compensation circuit value used in pF =0.007xC0+0.07xC1+0.7xC2+7xC3

4.8 Key Module

4.8.1 Key Configuration

The lowest level post-processing module is ‘Key’, which implements the functionality required for touch sensor operation. This module provides signal post-processing, environmental drift, touch detection, touch state machine and timing management for the implementation of application touch sensors. There are 64 possible key configurations corresponding to each X/Y node combination definition. The count of the key configurations corresponds to the node group count.

4.8.1.1 Threshold

A sensor’s (detect) threshold defines how much its signal must differ from its reference level to qualify as a potential touch detect.

The final detection confirmation must however satisfy the Detect Integrator (DI, see section “Touch Detection Integration and Anti-touch Detection Integration”) limit. Larger threshold values desensitize sensors since the signal must change more (i.e. requires larger touch) in order to exceed the threshold level. Conversely, lower threshold levels make sensors more sensitive. The threshold setting depends on the amount of signal swing that occurs when a sensor is touched. Thicker front panels or smaller electrodes usually have smaller signal swing on touch, and thus require lower threshold levels.

Typically, detect threshold is set to 50% of touch delta. Desired touch delta for a button is ~30 to 80 counts and for wheels or sliders ~50 to 120 counts. Large deltas also mean longer drift rates; they can also make the drift ineffective, so a balance must be maintained. Make sure to keep deltas between 50->120 using gain.

A threshold is an 8-bit field value defining a detection value. When a touch modifies the capacitance, this apparent change in capacitance (delta) is compared to the configured touch threshold, and if it exceeds the threshold then the sensor is deemed to be in detect. If a gain is applied on the node, the threshold is automatically adapted to the gained range.

4.8.1.2 Hysteresis

The PTC employs programmable hysteresis levels of n%, n=50, 25, 12.5, 6.25. The detect hysteresis is a percentage of the distance from the threshold level back towards the reference, and defines the point at which touch detection drops out. A 12.5%, the hysteresis point is closer to the threshold level than to the signal reference level. Hysteresis prevents chatter and works to make key detection more robust. Hysteresis is only used once the key has been declared to be in detection, in order to determine when the key should drop out. Excessive amounts of hysteresis can result in stuck keys that do not release. Conversely, low amounts of hysteresis can cause key chatter due to noise or minor amounts of finger motion. The usage of gain and hysteresis allows rejecting unwanted negative pulse on a delta value that does not exceed threshold hysteresis band. This helps reducing the filtering parameters and so improves the overall system performance without reducing its stability. This is the last step to perform when working on touch application noise immunity.

The Hysteresis field is the criterion of convergence of the automatic tuning as Tau=RC first order R-C filter.

- 0 for 50% of the threshold
- 1 for 25% of the threshold
- 2 for 12.5% of the threshold
- 3 for 6.25% of the threshold

4.8.1.3 AKS Group

AKS® stands for Adjacent Key Suppression.

In designs where the sensors are close together or configured for high sensitivity, multiple sensors might report detection simultaneously. To allow applications to determine the intended single touch, the touch library provides the user the ability to configure a certain number of sensors in an AKS group.

When a group of sensors are in the same AKS group, only the first strongest sensor will report detection. The sensor reporting detection will continue to report detection even if another sensor's delta becomes stronger. The sensor stays in detect until its delta falls lower than its detection threshold. If any more sensors in the AKS group are still in detect, only the strongest will report detection. At a given time point, only one sensor from each AKS group is reported to be in detect.

AKS_group features the group number for each key:

- AKS groups are numbered 1 to 7
- 0 for no AKS grouping

The firmware provides the ability to configure a sensor to belong to one of the Adjacent Key Suppression (AKS) groups.

AKS technology permits the suppression of multiple key presses based on relative signal strength. This feature assists in solving the problem of surface moisture which can bridge a key touch to an adjacent key, causing multiple key presses. This feature is also useful for panels with tightly-spaced keys, where a fingertip might inadvertently activate an adjacent key. AKS technology works for keys that are AKS-enabled anywhere in the matrix and is not restricted to physically adjacent keys.

The PTC has no knowledge of which keys are actually physically adjacent. When enabled for a key, Adjacent Key Suppression causes detections on that key to be suppressed if any other AKS-enabled key in the panel has a more delta signal deviation from its reference during the Detection Integration process. Once a key reaches detect it stays in detect as long as the touch remains, regardless of the signal strength on any other AKS-enabled keys.

This feature does not account for varying key gains (burst length) but ignores the actual detection threshold setting for the key. If AKS-enabled keys have different sizes, it may be necessary to reduce the gains of larger keys to equalize the effects of AKS technology. The signal threshold of the larger keys can be altered to compensate for this without causing problems with key suppression. Adjacent Key Suppression works to augment the natural moisture suppression of narrow-gated transfer switches creating a more robust sensing method.

In addition, when a key in an AKS group needs to be reburst, the entire AKS group will be reburst, leading to an increase in acquisition time if a reburst is required.

4.8.2 Key Group Configuration

This group configuration is valid for all defined keys. It manages the detection parameters, the sensor capacitance drift control and the reference recalibration process. This process uses the compensation of the capacitance in the PTC AFE.

4.8.2.1 Calibration using Capacitance Compensation

To maximize the dynamic range of the PTC measurement, ideally, the average of the measured signal is centered about the PTC mid-rail. An uncompensated signal will not meet this criteria, thus dynamic range limits may occur. The compensation mechanism balances the sensor capacitance with an internal capacitance. This balancing results in the average for the signal being centered at mid-value for the PTC. Once calibrated, the signal point is stored as the Key Data Reference. The compensation value is stored as the node data comp_cap value.

This calibration is started by the firmware during initialization of the PTC. This operation is automated and does not need to be programmed. The embedded algorithm performs a loop of several signal measurements and modifies the compensation capacitance until the signal mid-value is reached. This algorithm converges to the correct reference data if the sensor capacitance if not bigger than 30 pF. This operation is performed for each enabled node.

4.8.2.2 Drifting after Calibration

After calibration and reference storage, the signal value may drift to a point where either a false touch occurs (positive drift) or a touch does not register, depending on environmental factors such as temperature and humidity. Drift compensation ensures that these slow variations are compensated. Since drift in general is much slower than a touch, the drift calibration does not react to touch events.

Drift in a general sense means adjusting the reference level (of a sensor) to allow compensation for the effect of temperature (or other factor) on physical sensor characteristics. Decreasing the reference level for such compensation is called Negative drift & increasing reference level is called Positive drift. Specifically, the drift compensation should be set to compensate faster for increasing signals than for decreasing signals.

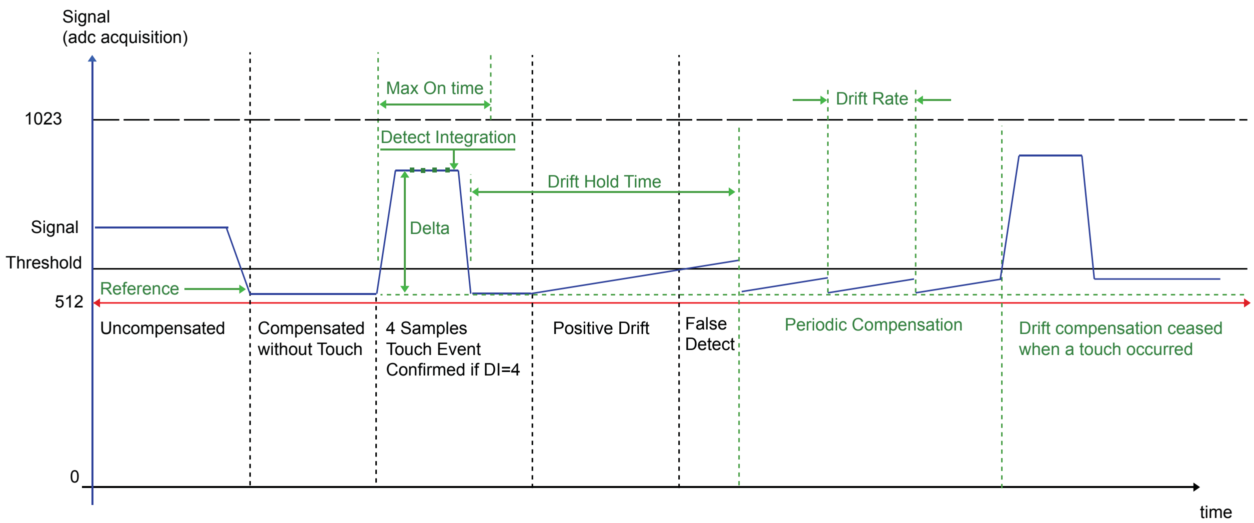
That drift will alter the reference value by 1 count if the signal is not equal to the reference at every drift period. The drift period is 200ms x drift rate. It is crucial that such drift be compensated for, otherwise false detections and sensitivity shifts can occur. Drift compensation occurs only while no detection is ongoing. Once a finger is sensed, the drift compensation mechanism ceases since the signal is detecting an object.

Drift compensation works only when the signal in question has not crossed the 'Detect threshold' level.

The drift compensation mechanism can be asymmetric. To make sure it occurs faster in one direction than in the other, simply change the appropriate setup parameters.

Signal values of a sensor tend to increase when an object (touch) is approaching the sensor or when sensor characteristics change with time and temperature. Increasing signals should not be compensated quickly, as an approaching finger could be compensated for partially or entirely before even touching the channel (towards touch drift).

Figure 4-5. Drift Compensation



The following parameters are involved in drift compensation:

- Touch_di
- Touch_drift_rate
- Touch_recal_thr

4.8.2.3 Drifting during Calibration

If the inherent capacitance compensation is run while a sensor is in touch, the compensation value will be too low, causing the reference to be below the desired value by one whole touch delta – hence no subsequent touches will register. Anti-touch-recalibration will detect this anomaly and correct it.

However, an object over the channel which does not cause detection, and for which the sensor has already made full allowance (over some period of time), could suddenly be removed, leaving the sensor with an artificially suppressed reference level and thus become insensitive to touch. In the latter case, the sensor should compensate for the object's removal by raising the reference level relatively quickly (away from touch drift).

Figure 4-6. Anti-touch Recalibration

Parameters involved in anti-touch drift compensation:

- Anti_touch_di
- Anti_touch_drift_rate
- Anti_touch_recal_thr

Those parameters have the same effect as their touch equivalent parameters, but in an anti-touch situation.

The term “anti-touch” means the reference felt below the specified Anti_touch_recal_thr level.

4.8.2.4 Count

The number of keys defined on the touch panel, including buttons, and all keys used in each slider or wheel.

4.8.2.5 Max on Time

The field Max_on_time is N x (200ms) with N=0 to 255.

If an object unintentionally contacts a sensor, resulting in a touch detection for a prolonged interval, it is usually desirable to recalibrate the sensor in order to restore its function, perhaps after a delay of some seconds.

The Maximum on Duration timer monitors such detections. If detection exceeds the timer’s settings, the sensor is automatically recalibrated. After a recalibration has taken place, the affected sensor once again functions normally even if it still in contact with the foreign object. Max on duration can be disabled by setting it to zero (infinite timeout), in which case the channel never recalibrates during a continuous detection.

4.8.2.6 Drift Hold Time

DHT = N x (200ms) with N=1 to 255, typical value about 4s.

Drift Hold Time (DHT) is used to restrict drift on all sensors while one or more sensors are activated. It defines the length of time the drift is halted after a key detection. This feature is useful in cases of high density keypads where touching a key or floating a finger over the keypad would cause untouched keys to drift, and therefore create a sensitivity shift, and ultimately inhibit any touch detection.

4.8.2.7 Reburst Mode

Requires a repeated measurement of specific sensors. It indicates that the application should restart measurement on the sensor group without waiting for the measurement cycle timeout.

- 0 for no reburst
- 1 for unresolved reburst
- 2 for all keys in reburst

4.8.2.8 Touch Detection Integration and Anti-touch Detection Integration

The field touch_di can be 0 to 255.

The field anti_touch_di can be 0 to 255.

The QTouch Library features a detect integration mechanism, which acts to confirm detection in a robust fashion. The detect integrator (DI) acts as a simple signal filter to suppress false detections caused by spurious events like electrical noise.

A counter is incremented each time the sensor delta has exceeded its threshold and stayed there for a specific number of acquisitions, without going below the threshold levels. When this counter reaches a preset limit (the DI value) the sensor is finally declared to be touched. If on any acquisition the delta is not seen to exceed the threshold level, the counter is cleared and the process has to start from the beginning. The DI process is applicable to a ‘release’ (going out of detect) event as well.

For example, if the DI value is 10, then the device has to exceed its threshold and stay there for 10 acquisitions in succession without going below the threshold level before the sensor is declared to be touched.

4.8.2.9 Touch Drift Rate and Anti-touch Drift Rate

The field touch_drift_rate is N=0 to 127.

The field anti _touch_drift_rate is N=0 to 127.

Period time is N x (200ms)

Drift compensation eliminates base value drifts resulting from environmental variations. The compensation interval is configurable and happens periodically once enabled in the sensor post-processing.

If any key is found to have a significant drop in signal delta (on the negative side), it is deemed to be an error condition. If this condition persists for more than the Drift rate, then an automatic recalibration is carried out. A counter is incremented each time the sensor delta is equal to the Anti-touch recalibration threshold and stayed there for a specific number of acquisitions. When this counter reaches a preset limit, the sensor is finally recalibrated. If on any acquisition the delta is greater than the recalibration threshold level, the counter is cleared and positive drifting is performed.

4.8.2.10 Anti-touch Recalibration Threshold

This field sets the negative threshold for triggering the anti-touch recalibration as a percentage of the touch threshold:

- 0 for 100% of the threshold
- 1 for 50% of the threshold
- 2 for 25% of the threshold
- 3 for 12.5% of the threshold
- 4 for 6.25% of the threshold

4.8.3 Key Data

There are 64 key data.

The simplest implementation of a capacitive sensor is a button, where the sensor consists of a single node (one electrode for self-capacitance, one pair of electrodes for mutual capacitance) and is interpreted as a binary state in Detect or Out of Detect.

4.8.3.1 Status

The sensor touch status is the primary touch sensor information utilized by a user application. The sensor state can have several values, as follows.

Table 4-8. Sensor Touch Status

Status	Content	Decimal Value
QTM_KEY_STATE_DISABLE	Disabled key	0
QTM_KEY_STATE_INIT	QTML structures have been initialized/implemented, but calibration has not been performed.	1
QTM_KEY_STATE_CAL	Calibration process is active. The processing algorithms will	2

Status	Content	Decimal Value
	determine the appropriate reference for the key.	
QTM_KEY_STATE_NO_DET	After calibration has been performed, this is the normal state for untouched sensors. This state can also be set when : • No detect has been confirmed from the QTM_KEY_STATE_FILT_OUT state, or • Detect is not confirmed when in QTM_KEY_STATE_FILT_IN.	3
QTM_KEY_STATE_FILT_IN	From QTM_KEY_STATE_NO_DET. Applied when Touch Signal > (reference +touch threshold parameter).	4
QTM_KEY_STATE_DETECT	TouchDetect_DI consecutive measurements are observed while remaining in QTM_KEY_STATE_FILT_IN, or Less than TouchDetect_DI consecutive measurements are observed while in QTM_KEY_STATE_FILT_OUT.	133
QTM_KEY_STATE_FILT_OUT	From QTM_KEY_STATE_DETECT. Applied when Signal < (reference +touch threshold parameter)	134
QTM_KEY_STATE_ANTI_TCH	When the measured touch signal has moved below the reference AND exceeds the ANTI TOUCH threshold. If consecutive signal levels are measured beyond the anti-touch-di count, then the sensor is reinitialized and calibrated.	7
QTM_KEY_STATE_SUSPEND	Suspended keys are not processed by QTML Key postprocessing. Suspended keys can drift, so they do not require recalibration when reactivated.	8

4.8.3.2 Node Structure Pointer

Pointer to the sensor node data structure.

4.8.3.3 Reference

The reference is the level of acquired signal when no touch occurs during the calibration phase. The value of the reference is typically 512 and (512) x (Digital Gain) when the gain is applied.

If the environmental situation changes on the touch panel, the key reference can change (drift) over time, when the key is not in detect. If this happens, a drift control process is used to allow the system to adjust to environmental changes, and is configured using associated drift parameters.

4.8.3.4 Delta Definition

The delta is not given as a field in the mailbox, but it is used internally by the firmware.

The delta is the absolute value of the difference between the reference and the signal.

Delta = abs(Signal - Reference)

The library takes care of the delta direction, so the user should only see a positive delta.

4.9 Scroller Module

Capacitive sensors may be implemented to simply detect contact as a button replacement, or functionally extended to provide a relative measurement of distance (proximity), 1D position (slider or wheel), 2D position (QTouch Surface), or 3D position (QTSurface with proximity).

In each case, the modular library detects a touch contact by a change in capacitance exceeding a preconfigured threshold. Once a contact has been confirmed, the various post-processing modules use the calculated touch delta to interpolate amongst neighboring sensors and calculate the location of the touch position or relative proximity.

4.9.1 Scroller Configuration

Four scroller configurations are possible.

4.9.1.1 Type

This field defines the type of the scroller - slider or wheel:

- Type=0 for slider
- Type=1 for wheel

4.9.1.2 key_start

This field defines the number of the scroller starting key, from 0 to 63.

The keys used in a scroller must be consecutive.

E.g. Key[key_start] ,Key[key_start+1]..Key[key_start+key_count-1]

4.9.1.3 key_count

This field defines the number of keys used in the scroller, from 2 to 63.

4.9.1.4 Resolution

The rotor or slider needs the position resolution (angle resolution in the case of a rotor, or linear resolution in the case of a slider) to be set. Resolution is the number of bits needed to report the position of rotor or slider. It can have values from 2 bits to 8 bits (resolution can be higher than this, but tradeoffs exist in resolution vs. overall response time vs. number of scrollers to be processed vs. size of scroller).

The programming value is a power of 2 with range 2 to 8, e.g. if a slider resolution is 7 bits, then the reported positions are in the range 0...127.

4.9.1.5 Dead Band

The dead band is a parameter only applied to sliders. It is the minimum and maximum counts at which the slider reports touch. This is a percentage of the scroller resolution value.

For example if the resolution is 10 bits (0-1023) and the dead band is 2%, then only the positions between minimum and maximum are reported.

- 2^n x 2% ≈ 20 minimum (n = bit resolution =10)
- 2^n - (2^n x 2%) ≈ 1000 maximum (n= bit resolution =10)

4.9.1.6 Position Hysteresis

The threshold used to reduce the number of position change updates. Can be useful in noisy environments or to determine the significance of the position change. Report a change in position if:

Current position > (previous position + position hysteresis)

Or

Current position < (previous position - position hysteresis)

4.9.2 Scroller Group Configuration

4.9.2.1 Key Data

Pointer to the start of key_data structure. Key data is the source data for scrollers. This parameter should not be modified by the end user.

4.9.2.2 Scroller Count Configuration

This field defines the number of scrollers to be monitored. Min 1, Max 4.

4.9.2.3 Enable Detect Reburst

Allows the firmware algorithm to restart an acquisition when considered wrong.

Reburst keys associated with the scroller if the scroller detects a touch. Used to provide responsive reporting of position along the scroller.

4.9.3 Scroller Data

4.9.3.1 Status

Reports current scroller status:

- Touch Active: Finger is detected.
- Scroller Reburst: Rebursting is active.
- Position Change: Position field is updated.

4.9.3.2 Right Hysteresis

Reports the position hysteresis assigned to the scroller from the scroller configuration.

4.9.3.3 Left Hysteresis

Reports the position hysteresis assigned to the scroller from the scroller configuration.

4.9.3.4 Raw Position

Unprocessed scroller position based on key data.

4.9.3.5 Position

The processed scroller position, when resolution, hysteresis, dead band (if slider) are applied. Value is 0 to 255.

Example: if slider resolution is 7 bits, then reported positions are in the range 0..127.

4.9.3.6 Contact Size

The largest touch delta measured across all keys associated with the scroller.

4.10 Frequency Hopping Module

- Noise Counter Measures

In any touch sensing application, the system designer must consider how electrical interferences in the target environment may affect the performance of the sensors.

Touch sensors with insufficient tuning can show failures in tests of either radiated or conducted noise, which can occur in the environment or power domain of the appliance or may be generated by the appliance itself during normal operation.

Frequency mode setting allows users to tune the PTC touch acquisition frequency characteristics to counter environment noise. The noise considered is a synchronous noise emitted by a source disturbing the PTC measurement at a given frequency. The white noise or random noise is not removed by this technique but can be reduced by an averaging (Oversampling mode).

Noise immunity comes at the cost of increased touch response time and power consumption. The system designer must carry out proper tuning of the touch sensors in order to ensure least power consumption. The QTouch Modular library has a number of user-configurable features which can be tuned to give the best balance between touch response time, noise immunity and power consumption.

- Frequency Hopping

This PTC subsystem supports frequency hopping, which tries to select a sampling frequency that does not interfere with noise at specific frequencies elsewhere in products or product operating environments. It tries to hop away from the noise. During the acquisition bursts, a sequence of pulses is emitted with a particular spacing, which equates to a particular sampling frequency. If the latter should coincide with significant noise generated elsewhere, touch sensing may be seriously impaired or false detections may occur.

The ‘Frequency Hop Autotune’ module applies a configurable cyclic frequency hopping algorithm, such that on each measurement cycle a different sampling frequency is used. A number of pre-configured frequencies are implemented in turn during consecutive measurement cycles.

To help reducing such noise, the burst frequency can use different modes.

Frequency modes:

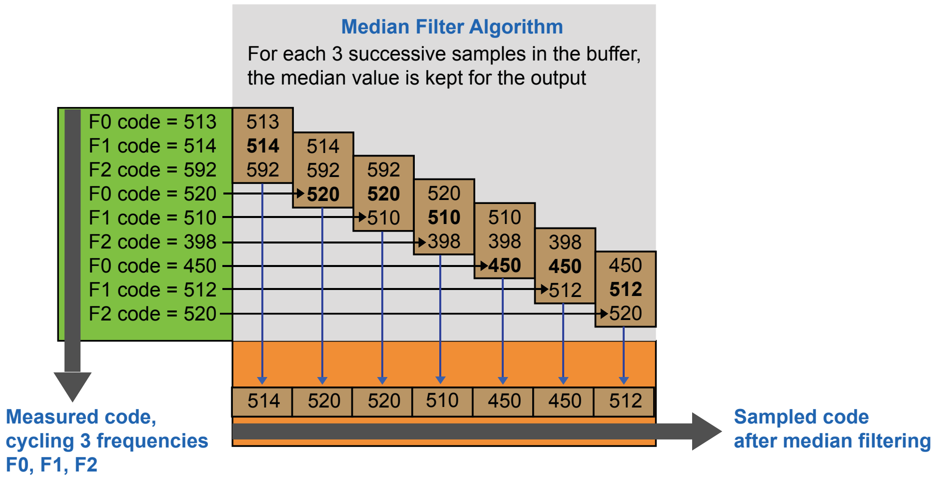
- Frequency Hopping Median Filter measures among 3 fixed frequencies.
- Frequency Hopping Auto-tune is set to sweep a configured range of frequencies with noise measurement.

4.10.1 Frequency Hopping Median Filter

This module applies a configurable cyclic frequency hopping algorithm, such that on each measurement cycle a different sampling frequency is used. A number of preconfigured frequencies are implemented in turn during consecutive measurement cycles.

The following figure illustrates the behavior of the median filter for 3 frequencies cycle / depth of median filter.

Figure 4-7. Median Filter Algorithm



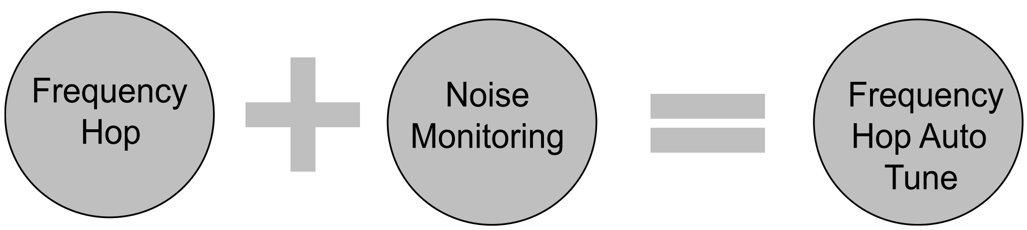
The median filter performs 3 successive acquisitions at 3 different frequencies defined in the field fh_freq.

Among those 3 data, the filter eliminates the bigger and the smaller ones, the remaining data is pulled out of the filter as a result of the acquisition. Then the frequency loop F0, F1, F2 will move one frequency step and a new acquisition is performed. This last result and the two previous ones are compared in the median filter to get a new output data.

4.10.2 Frequency Hopping Auto-tune

To perform autotuning, the signals measured on each sensor node are recorded for each selected frequency. When one frequency shows greater variance than others, that frequency is removed from the measurement sequence and replaced with another.

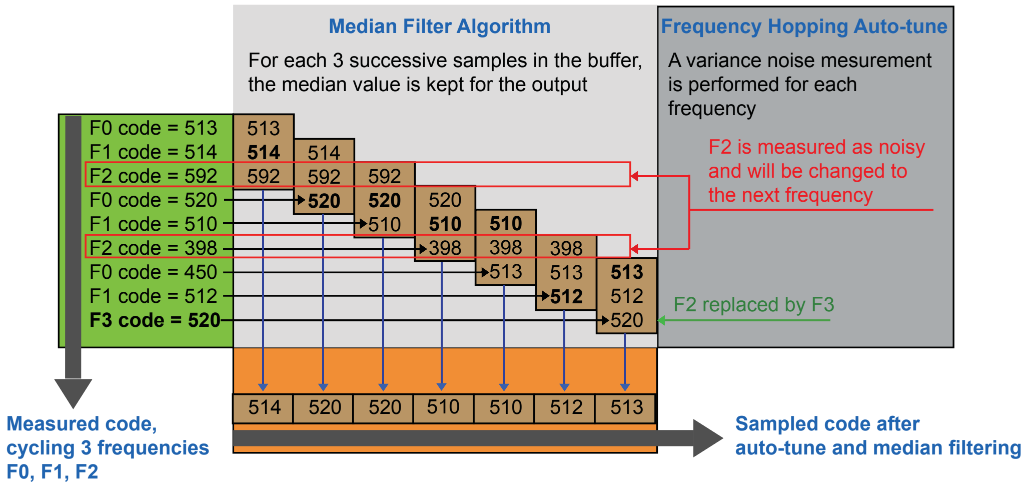
Figure 4-8. Frequency Hopping Auto-tune



Algorithm

- Stability of signals from all three frequencies is monitored separately.
- If the signal from a particular frequency is unstable, it is replaced and monitored.
- The other two frequencies are retained.
- If the replaced frequency is stable, it is retained, otherwise the next frequency is selected and monitored.
- This is continued until a stable frequency is found.
- If none of the frequencies is found to be stable, the one with the lowest signal variance is used.

Figure 4-9. Auto-tune Hopping



4.10.3 Frequency Auto-tune Configuration

4.10.3.1 Count

The number of sensors on the panel (e.g., one slider, one wheel and two buttons).

Value: 1 to 4.

4.10.3.2 Number of Frequency Auto-tune Configurations

For this device, the field num_freqs is a fixed value at n=3.

Where 3 frequencies are included in the cycle, a 3-point median filter is applied to the output data.

4.10.3.3 Frequency Option Select

Pointer to the specific entry in median filter frequencies for internal processing.

Not to be modified by the user.

4.10.3.4 Median Filter Frequency

Pointer to array of selected frequencies =freq[n]

4.10.3.5 Enable Frequency Auto-tune

This parameter is set for the two frequency hopping options: Median filter + Noise measurement.

Value is on or off.

4.10.3.6 Maximum Variance Limit

The signal variance required to trigger returning of hop frequency. Value=1 to 255.

This parameter requires a system evaluation. As initial starting point, set to approx. 75% of the smallest key threshold and then observe system performance.

4.10.3.7 Auto-tune Count in Limit

A user-defined parameter to decide the number of consecutive measurements with noise before the frequency should be changed.

4.10.3.8 Frequency Hopping Frequencies

The freq0, freq1, freq2 fields define the value of the initial 3 frequencies to be used. The frequency hopping autotune algorithm will change these and use PTC_FREQ_SEL 0-15, depending on the noise detected.

For more definitions, see Freq Option.

4.10.4 Frequency Auto-tune Data

4.10.4.1 Status

Currently not used, this field may be used to report frequency hop module status.

4.10.4.2 Current Frequency

Field freq is the value of the frequency being used during auto-tuning.

4.10.4.3 Filter Buffer

A pointer used internally, not accessible by the user. Filter buffer used to store past cycle signal values of sensors, to determine when to change frequencies

4.10.4.4 Acquisition Node Data

The field filter_buffer is the signal being measured when in Auto-tune mode, which corresponds to the freq field.

4.10.4.5 Frequency Tune Count In

A pointer used internally, not accessible by the user. A counter used to determine when to change frequencies.

5. Revision History

5.1 Rev. A - 11/2017

This is the initial released version of this user's guide.

The Microchip Web Site

Microchip provides online support via our web site at http://www.microchip.com/. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Customer Change Notification Service

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at http://www.microchip.com/. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.

- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable.”

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip’s code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Trademarks

ISBN: 978-1-5224-2323-2

Quality Management System Certified by DNV

ISO/TS 16949

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7289-7561

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820