Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

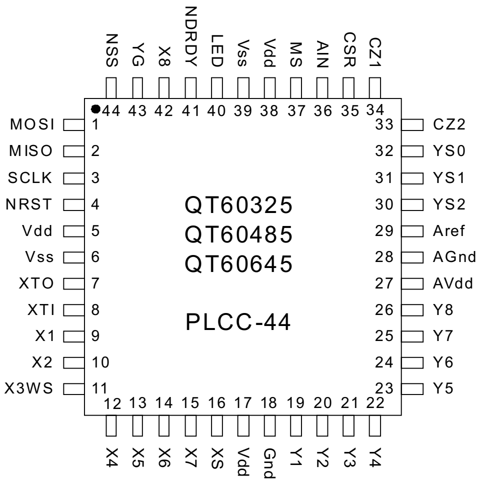
Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

QT60325, QT60485, QT60645

PRELIMINARY

32, 48, 64 KEY QMatrix™ KEYPANEL SENSOR ICs

- Advanced second generation QMatrix controllers
- Up to 32, 48 or 64 touch keys through any dielectric
- 100% autocal for life - no adjustments required
- SPI Slave or Master/Slave interface to a host controller
- Keys individually adjustable for sensitivity, response time, and many other critical parameters
- Sleep mode with wake pin
- Synchronous noise suppression
- Mix and match key sizes & shapes in one panel
- Adjacent key suppression feature
- Panel thicknesses to 5 cm or more
- Low overhead communications protocol
- 44-pin TQFP package



APPLICATIONS -

- Security keypanels
- Industrial keyboards
- Appliance controls
- Outdoor keypads
- ATM machines
- Touch-screens
- Automotive panels
- Machine tools

The QT60325, QT60485, and QT60645 digital charge-transfer (“QT”) QMatrix™ ICs are designed to detect human touch on up to 32, 48, or 64 keys respectively using a scanned, passive X-Y matrix. It will project the keys through almost any dielectric, e.g. glass, plastic, stone, ceramic, and even wood, up to thicknesses of 5 cm or more. The touch areas are defined as simple 2-part interdigitated electrodes of conductive material, like copper or screened silver or carbon deposited on the rear of a control panel. Key sizes, shapes and placement are almost entirely arbitrary; sizes and shapes of keys can be mixed within a single panel of keys and can vary by a factor of 20:1 in surface area. The sensitivity of each key can be set individually via simple functions over the SPI port, for example via Quantum’s QmBtn program. Key setups are stored in an onboard eeprom and do not need to be reloaded with each powerup.

These ICs are designed specifically for appliances, electronic kiosks, security panels, portable instruments, machine tools, or similar products that are subject to environmental influences or even vandalism. They permit the construction of 100% sealed, watertight control panels that are immune to humidity, temperature, dirt accumulation, or the physical deterioration of the panel surface from abrasion, chemicals, or abuse. To this end the devices contain Quantum-pioneered adaptive self-calibration, drift compensation, and digital filtering algorithms that make the sensing function robust and survivable.

The part can scan matrix touch keys over LCD panels or other displays when used with clear ITO electrodes arranged in a matrix. It does not require 'chip on glass' or other exotic fabrication techniques, thus allowing the OEM to source the matrix from multiple vendors. Materials such as such common PCB materials or flex circuits can be used.

External circuitry consists of an opamp, R2R ladder-DAC network, a common PLD, a jfet switch, and a small number of resistors and capacitors, which can fit into a footprint of roughly 8 sq. cm (1.5 sq. in). Control and data transfer is via a SPI port which can be configured in either a Slave or Master/Slave mode.

QT60xx5 ICs make use of an important new variant of charge-transfer sensing, transverse charge-transfer, in a matrix format that minimizes the number of required scan lines. Unlike some older technologies it does not require one sensing IC per key.

AVAILABLE OPTIONS

T_A	TQFP
0^0C to $+70^0$C	QT60325-S
0^0C to $+70^0$C	QT60485-S
0^0C to $+70^0$C	QT60645-S
-40^0C to $+105^0$C	QT60325-AS
-40^0C to $+105^0$C	QT60485-AS
-40^0C to $+105^0$C	QT60645-AS

QUANTUM
RESEARCH GROUP

Copyright © 2001 Quantum Research Group Ltd
Pat Pend. R1.04/02.02

Contents

Table 1.1 Device Pin List

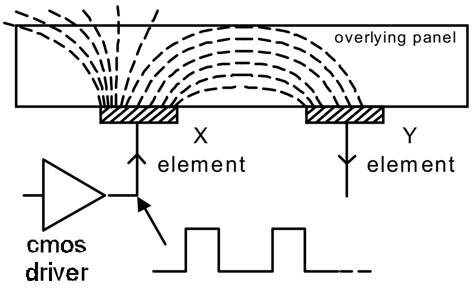
Pin	Name	Type	Description
1	MOSI	I/O PP	Master-Out / Slave In SPI line. In Master/Slave SPI mode is used for both communication directions. In Slave SPI mode is the data input (in only).
2	MISO	I/O PP	Master-In / Slave Out SPI line. Not used in Master/Slave SPI mode. In Slave mode outputs data to host (out only).
3	SCK	I/O PP	SPI Clock. In Master mode is an output; in Slave mode is an input
4	RST	I	Reset input, active low reset
5	Vdd	Pwr	+5 supply
6	Vss	Pwr	Ground
7	XTO	O	Oscillator drive output. Connect to resonator or crystal. Can drive a charge pump circuit for Vee supply
8	XTI	I	Oscillator drive input. Connect to resonator or crystal, or external clock source.
9	X0	O	X0 Drive matrix scan / R2R DAC Ladder drive
10	X1	O	X1 Drive matrix scan / R2R DAC Ladder drive
11	X2WS	O	X2 Drive matrix scan / R2R DAC Ladder drive / Wake from Sleep / Sync to noise source
12	X3	O	X3 Drive matrix scan / R2R DAC Ladder drive
13	X4	O	X4 Drive matrix scan / R2R DAC Ladder drive
14	X5	O	X5 Drive matrix scan / R2R DAC Ladder drive
15	X6	O	X6 Drive matrix scan / R2R DAC Ladder drive
16	XS	O	X summation / R2R DAC Ladder drive
17	Vdd	Pwr	+5 supply
18	Vss	Pwr	Ground
19	YC0	O	Y 0 Line clamp control
20	YC1	O	Y 1 Line clamp control
21	YC2	O	Y 2 Line clamp control
22	YC3	O	Y 3 Line clamp control
23	YC4	O	Y 4 Line clamp control
24	YC5	O	Y 5 Line clamp control
25	YC6	O	Y 6 Line clamp control
26	YC7	O	Y 7 Line clamp control
27	AVdd	Pwr	+5 supply for analog sections
28	AGnd	Pwr	Ground
29	Aref	Pwr	Analog reference, connect to Vcc
30	YS2	O	Transfer switch control bit 2
31	YS1	O	Transfer switch control bit 1
32	YS0	O	Transfer switch control bit 0
33	CZ2	O	Charge cancellation drive for CZ2 capacitor
34	CZ1	O	Charge cancellation drive for CZ1 capacitor
35	CSR	O	Charge integrator reset line. Active high or active low (select polarity via Setups)
36	Ain	I	Analog input from amplifier
37	MS	I/O OD	SPI Mode / Sync out. Connect via 10k resistor to Vcc or Gnd for mode. Scope sync yields Pulse.
38	Vdd	Pwr	+5 supply
39	Vss	Pwr	Ground
40	LED	O	Active low LED status drive / Activity indicator
41	DRDY	O OD	Data ready output for Slave SPI mode; active low
42	X7	O	X7 Drive matrix scan
43	YG	O	Y gate control to drive Y dwell timing circuit
44	SS	IO OD	Slave select for SPI direction control; active low

I/O: I = Input
O = Output
Pwr = Power pin
I/O = Bidirectional line
PP = Push Pull output drive
OD = Open drain output drive

1 Overview

QMatrix devices are digital burst mode charge-transfer (QT) sensors designed specifically for matrix geometry touch controls; they include all signal processing functions necessary to provide stable sensing under a wide variety of changing conditions. Only a few external parts are required for operation. The entire circuit can be built within 8 square centimeters of PCB area.

Figure 1-1 Field flow between X and Y elements



QMatrix devices include charge cancellation methods which allow for a wide range of key sizes and shapes to be mixed together in a single touch panel. These features permit the construction of entirely new classes of keypads never before contemplated, such as touch-sliders, back-illuminated keys, and complex warped panel shapes, all at very low cost.

The devices use an SPI interface running at up to 1.5MHz to allow key data to be extracted and to permit individual key parameter setup. The interface protocol uses simple single byte commands and responds with single byte responses in most cases. The command structure is designed to minimize the amount of data traffic while maximizing the amount of information conveyed.

In addition to normal operating and setup functions the device can also report back actual signal strengths and error codes.

QmBtn software for the PC can be used to program the IC as well as read back key status and signal levels in real time.

QMatrix parts employ transverse charge-transfer ('QT') sensing, a new technology that senses changes in the charge forced across an electrode by a digital edge.

The parts are electrically identical with the exception of the number of keys which may be sensed.

Figure 1-3 Field Flows When Touched

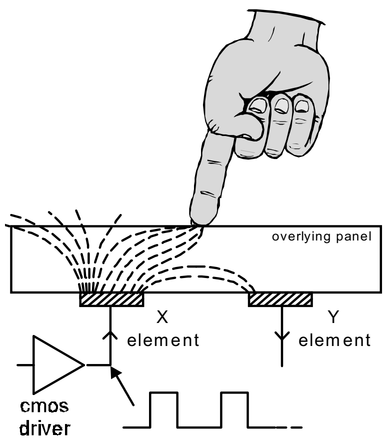
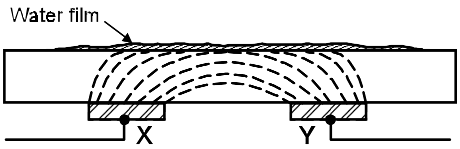


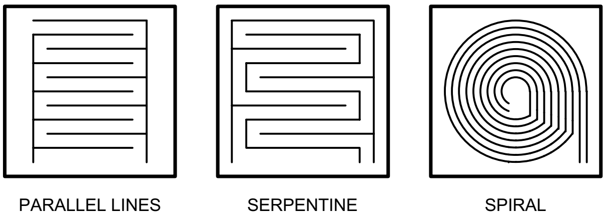
Figure 1-4 Fields With a Conductive Film



1.1 Field Flows

Figure 1-1 shows how charge is transferred across an electrode set to permeate the overlying panel material; this charge flow exhibits a high dQ/dt during the edge transitions of the X drive pulse. The charge emitted by the X electrode is partly received onto the corresponding Y electrode which is then processed. The parts use 8 'X' edge-driven rows and 8 'Y' sense columns to permit up to 64 keys. Keys are typically formed from interleaved conductive traces on a substrate like a flex circuit or pcb (Figure 1-2).

Figure 1-2 Sample Electrode Geometries



The charge flows are absorbed by the touch of a human finger (Figure 1-3) resulting in a decrease in coupling from X to Y. Thus, received signals decrease or go negative with respect to the reference level during a touch.

Water films cause the coupled fields to increase slightly, making water films easy to distinguish from touch.

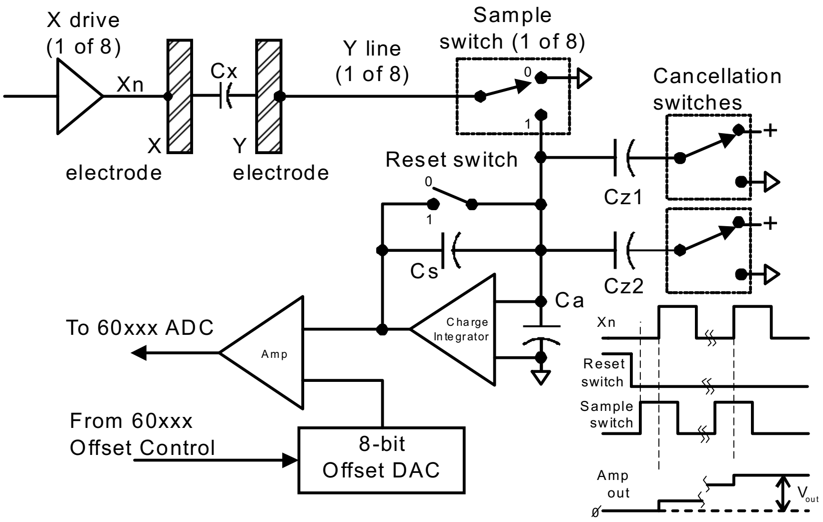
1.2 Circuit Model

An electrical circuit model is shown in Figure 1-5. The coupling capacitance between X and Y electrodes is represented by Cx. While the reset switch is open, a sample switch is gated so that it transfers charge flows only from the rising edge of X into a charge integrator. On the falling edge of X, the switch connects the Y line to ground to allow the charge across Cx to neutralize to zero. The voltage change on the output of the charge integrator after each X edge is quite small, on the order of a few tens of millivolts. Changes due to touch are typically under 0.1% of total integrator voltage. The X pulse can be repeated in a burst of up to 64 pulses to increase the change in integrator output voltage due to touch during an acquire (Section 3.6) to increase gain.

The charge detector is an opamp configured as an integrator with a reset switch; this creates a virtual ground input, making the Y lines appear low impedance when the sample switch is closed. This configuration effectively eliminates cross-coupling among Y lines while greatly lowering susceptibility to EMI. The circuit is also highly immune to capacitive loading on the Y lines, since stray C from Y to ground appears merely as a small parallel capacitance across a virtual ground.

The circuit uses an 8-bit ADC, with a subranging structure to effectively deliver a 14-bit total conversion 'space' (see Figure 1-6 and Section 3.3). In this way the circuit can tolerate very large

Figure 1-5 QT60xx5 Circuit Model

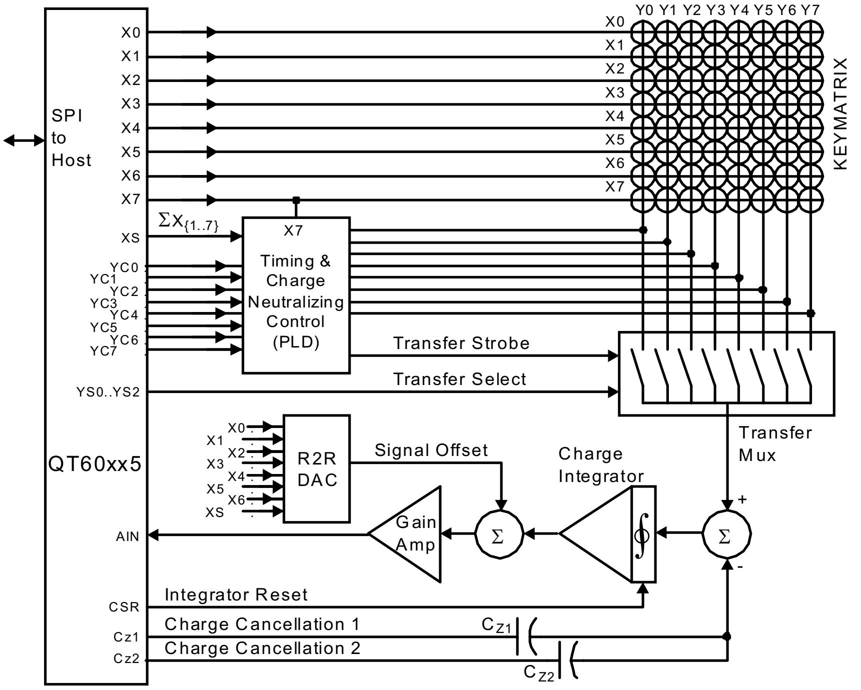


absolute signals yet still respond to very small signal changes. Subranging is provided by two offset mechanisms which can be thought of as 'coarse' and 'fine' offsets.

The 'coarse' method uses one or two switched Cz capacitors to subtract charge from the charge integrator to create up to two step offsets, to bring the analog signal back to a more reasonable level. This action occurs during the course of the burst.

The 'fine' method of offset uses an 8-bit R2R ladder DAC driven by the X drive lines to create an offset in the amplifier stage. The DAC is driven after the burst has ceased and the charge accumulated, so there is no conflict in this dual-use of the X lines.

Figure 1-6 Circuit Block Diagram



Short sample gate dwell times after the X edges are used to limit the effect of moisture spreading from key to key by taking advantage of the RC filter-like nature of continuous films; the shorter the dwell time, the less time that the charge has to travel through the impedance of the film (Section 3.13). This effect is completely independent of the frequency of burst repetition, intra-burst pulse spacing, or X drive pulse width.

Burst mode operation permits reduced power consumption and reduces RF emissions, while permitting excellent response time.

1.3 Matrix Configuration

The matrix scanning configuration is shown in Figure 1-5. The ‘X’ drives are sequentially pulsed in groupings of bursts; an 8:1 analog mux acts as the sample switch for all ‘Y’ lines. At the intersection of each ‘X’ and ‘Y’ line in the matrix itself, where a key is desired, should be an interdigitated electrode set similar to those shown in Figure 1-2. The outermost electrode or the key border should always be connected to an ‘X’ drive; flooding the area around keys with X fill to a width of up to 10mm can help in suppressing moisture films further. Consult Quantum for application assistance on key design.

1.4 Communications

The device uses two variants of SPI communications, Slave-only and Master-Slave. Over this interface is a command and data transfer structure designed for high levels of flexibility using minimal numbers of bytes. For more information see Sections 4 and 5.

Device variations: Refer to Section 3.1 for differences between the parts covered by this datasheet.

2 Signal Processing

The devices calibrate and process all signals using a number of algorithms specifically designed to provide for high survivability in the face of adverse environmental challenges. The QT60xx5 provides a large number of processing options which can be user-selected to implement very flexible, robust keypanel solutions.

2.1 Negative Threshold

See also command ^A, page 23

The negative threshold value is established relative to a key’s signal reference value. The threshold is used to determine key touch when crossed by a negative-going signal swing after having been filtered by the detection integrator (Section 2.6). Larger absolute values of threshold desensitize keys since the signal must travel farther in order to cross the threshold level. Conversely, lower thresholds make keys more sensitive.

As Cx and Cs drift, the reference point drift-compensates for these changes at a user-settable rate (Section 2.4); the threshold level is recomputed whenever the reference point moves, and thus it also is drift compensated.

The negative threshold is programmed on a per-key basis using the setup process described in Section 5.

2.2 Positive Threshold

See also command ^B, page 23

The positive threshold is used to provide a mechanism for recalibration of the reference point when a key's signal moves abruptly to the positive. These transitions are described more fully in Section 2.7.

Positive threshold levels are programmed in using the setup process described in Section 5 on a per-key basis.

2.3 Hysteresis

See also command ^C and ^D, page 24

Refer to Figure 2-1. QT60xx5 ICs employ programmable hysteresis levels of 12.5%, 25%, or 50% of the delta between the reference and threshold levels. There are different hysteresis settings for positive and negative thresholds which can be set by the user. The percentage refers to the distance between the reference level and the threshold at which the detection will drop out. A percentage of 12.5% is less hysteresis than 25%, and the 12.5% hysteresis point is closer to the threshold level than to the reference level.

The hysteresis levels are set for all keys only; it is not possible to set the hysteresis differently from key to key on either the positive or negative hysteresis levels.

2.4 Drift Compensation

*See also command*s *^H, ^I, page 25*

Signal levels can drift because of changes in Cx and Cs over time. It is crucial that such drift be compensated, else false detections, non- detections, and sensitivity shifts will follow. The QT60xx5 can compensate for drift using two setups, ^H and ^I.

Drift compensation (Figure 2-1) is performed by making the reference level track the raw signal at a slow rate, but only while there is no detection in effect. The rate of adjustment must be performed slowly, otherwise legitimate detections could be ignored. The devices drift compensate using a slew-rate limited change to the reference level; the threshold and hysteresis values are slaved to this reference.

When a finger is sensed, the signal falls since the human body acts to absorb charge from the cross-coupling between X and Y lines. An isolated, untouched foreign object (a coin, or a water film) will cause the signal to rise very slightly due to the enhanced coupling thus created. These effects are contrary to the way most capacitive sensors operate.

Once a finger is sensed, the drift compensation mechanism ceases since the signal is legitimately detecting an object. Drift compensation only works when the key signal in question has not crossed the negative threshold level (Section 2.1).

The drift compensation mechanism can be made asymmetric if desired; the drift-compensation can be made to occur in one direction faster than it does in the other simply by setting ^H and ^I to different settings.

Specifically, drift compensation should be set to compensate faster for increasing signals than for decreasing signals. Decreasing signals should not be compensated quickly, since an approaching finger could be compensated for partially or entirely before even touching the touch pad. However, an obstruction over the sense pad, for which the sensor has already made full allowance for, could suddenly be removed leaving the sensor with an artificially suppressed reference level and thus become insensitive to touch. In this latter case, the sensor should compensate for the object's removal by raising the reference level relatively quickly.

The drift compensation rate can be set for each key individually, and can also be disabled completely if desired on a per-key basis.

Drift compensation and the detection time-outs (Section 2.5) work together to provide for robust, adaptive sensing. The time-outs provide abrupt changes in reference location depending on the duration of the signal 'event'.

Drift compensation can result in reference levels that are at the boundaries of the 8-bit signal window. When this occurs, saturation is reached and the drift compensation process stops. One of two error flags is set when the signal approaches either end of the signal window; it is up to the host controller to read these flags and induce a full recalibration via a recalibration command at that time (Section 2.10 and command ‘b’, page 28)

2.5 Detection Recalibration Delay

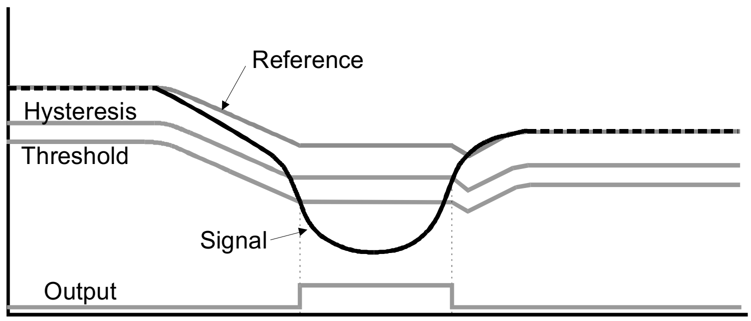
See also command ^L, page 25

If a foreign object contacts a key the key's signal may change enough in the negative direction, the same as a normal touch, to create an unintended detection. When this happens it is usually desirable to cause the key to be recalibrated in order to restore its function after a time delay of some seconds.

The Negative Recal Delay timer monitors this detection duration; if a detection event exceeds the timer's setting, the key will be fast-recalibrated within its current 8-bit window. This form of recalibration is simply one of setting Reference = Signal, and does not affect Offset or Cz state; as a result this form of recalibration requires only one burst spacing interval

t

Figure 2-1 Thresholds and Drift Compensation



o accomplish. Only a full recalibration via a reset or a recalibration command will perform a complete recalibration involving both the R2R Offset and Cz capacitors (Section 2.10).

After a fast recalibration has taken place, the affected key will once again function normally even if it is still being contacted by the foreign object. This feature is set on a per-key basis using setup ^L. It can be disabled if desired by setting this parameter to zero, so that it will not recalibrate automatically.

2.6 Detect Integrator

See also command ^J, page 25

To suppress false detections caused by spurious events like electrical noise, the QT60xx5 incorporates a 'detection integrator' counter that increments with each detection sample until a user-defined limit is reached, at which point a detection is confirmed. If no detection is sensed on any of the samples prior to the final count, the counter is reset immediately to zero, forcing the process to restart.

When an active key is released, the counter must count down to zero before the key state is set to 'off'. Setting a key's detection integrator target value to zero disables that key although the bursts for that key continue normally.

The detection integrator is extremely effective at reducing false detections at the expense of slower reaction times. In some applications a slow reaction time is desirable; the detection integrator can be used to intentionally slow down touch response in order to require the user to touch longer to operate the key.

There are 16 possible values for this function.

2.7 Positive Recalibration Delay

See also command ^K, page 25

A recalibration can occur automatically if the signal swings more positive than the positive threshold level. This condition can occur if there is positive drift but insufficient positive drift compensation, or if the reference moved negative due to a recalibration, and thereafter the signal returned to normal.

As an example of the latter, if a foreign object or a finger contacts a key for period longer than the Negative Recal Delay, the key is by recalibrated to a new lower reference level. Then, when the condition causing the negative swing ceases to exist (e.g. the object is removed) the signal can suddenly swing back positive to near its normal reference.

It is almost always desirable in these cases to cause the key to recalibrate to the new signal level so as to restore normal touch operation. The device accomplishes this by simply setting Reference = Signal.

The time required to detect this condition before recalibrating is governed by the Positive Recalibration Delay command. In order for this feature to operate, the signal must rise through the positive threshold level (Section 2.2) for the proscribed user-set interval determined by ^K.

After the Positive Recal Delay interval has expired and the fast-recalibration has taken place, the affected key will once again function normally. This interval can be set on a per-key basis; it can also be disabled by setting ^K to zero.

2.8 Reference Guardbanding

See also commands ^N, ^O, page 26; L, page 27

QT60xx5 devices provide for a method of self-checking that allows the host device to ascertain whether one or more key reference levels are 'out of spec'. This feature can be used to determine if an X or Y line has broken, the matrix panel has delaminated from the control panel, or there is a circuit fault.

Guardbanding alerts the host controller when the reference level of a key falls outside of acceptable absolute levels. The guardband is expressed in percent of absolute reference from the reference level of each individual key. The normal reference levels can be locked into internal eeprom via the Lock command 'L' during production; deviations in references that fall outside the guardbands centered on these reference levels are then reported as errors.

Positive excursion guarding is treated separately from negative excursion guarding. The possible negative settings are from 1% to 99% of absolute signal reference in steps of 1% as set by command ^O. Positive excursions can run from 10% to 1,000% in steps of 10% as set by command ^N. A setting of 0 disables the corresponding guardband direction.

Since the circuit uses a segmented ADC approach with a 'coarse' (based on Cz states) and 'fine' (based on R2R ladder drive) offsets, the determination of percentage reference deviation from 'normal' presents a problem. The contributions of the Cz caps and the R2R ladder must be factored into the determination in order to make an accurate assessment of the error band. There are three commands which set coefficients used to convert the Cz and DAC offset values to 'absolute signal' values, according to the following equation, for each key:

TotalRef(k) = (C1 x nCz) + (C2 x Offset) + SigRef

Where -
TotalRef(k) is the equivalent absolute reference for key 'k';
C1 is a global constant set by commands ^T and ^U;
C2 is a global constant set by command ^V;
nCz is the number of Cz caps switched in for key 'k';
Offset is the noted value of the R2R DAC for key 'k';
SigRef is the noted current 'window reference' for key 'k'.

The percent deviations are computed in relation to TotalRef(k) on a per-key basis at the time the 'L' command is executed. Once the L command has recorded all values of TotalRef into eeprom, the part will compare the actual running reference level of each key to its corresponding TotalRef value to see if it falls outside the guardbands specified by global parameters ^N and ^O.

Values which correspond to the reference circuit of Figure 3-1 are:

C1 = 1513; ^T value = 0x05, ^U value = 0xE9
C2 = 8; ^V value = 0x08

Guardbanding tests should not be confused with Reference Boundary errors (Section 2.11). Guardbanding can report errors that occur even if the signal is properly centered in the ADC window, while Reference Boundary error reporting cannot. Guardband tests do however require that the key being checked be first fully recalibrated in order to allow the Cz and DAC offset values to alter.

If a key is outside of a limit, either of bits 2 and 3 of command 'e' will be set for that key. The error will also appear as an error in a bitfield reported with command 'E'.

There is no mechanism by which keys will automatically recalibrate if the reference drifts past a guardband boundary.

2.9 Adjacent Key Suppression

See also command ^P, page 26

QT60xx5 devices incorporate adjacent key suppression that can be selected on a per-key basis. This feature permits the suppression of multiple key presses based on relative signal strength. This feature assists in solving the problem of surface water 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 also affect an adjacent key. This feature acts to suppress the unintended key(s).

Adjacent key suppression works for keys across the entire panel and is not restricted to physically adjacent keys. The device has no knowledge of which keys are adjacent. When enabled for a key, adjacent key suppression causes detections on that key to be suppressed if any other key in the panel has a more negative signal deviation from its reference, even if the other key does not have adjacent key suppression enabled.

This feature does not account for varying key gains (burst length) but ignores the actual negative detection threshold setting for the key. If keys in a panel have different sizes, it may be necessary to reduce the gains of larger keys relative to smaller ones to equalize the effects of adjacent key suppression. The 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 (Section 3.13), creating a more robust sensing method.

2.10 Full Recalibration

See also command ‘b’, page 28

The devices fully recalibrate on powerup, after a hard reset, a soft reset or after a recalibrate ‘b’ command using an algorithm that seeks out the optimal level of R2R offset and Cz cancellation on a per-key basis. After powerup or a reset the matrix is scanned key by key and appropriate calibrations are set for each in accordance with user-defined setup information. Since the circuit can tolerate a very wide signal range, it is capable of adapting to a wide mix of key sizes and shapes having widely varying Cx coupling capacitances.

If a false calibration occurs due to a key touch or foreign object on the keys during powerup, the affected key will recalibrate again when the object is removed depending on the settings of Positive Threshold and Positive Recal Delay (Sections 2.2 and 2.7).

Full recalibration is distinct from fast-recalibration, wherein only the Reference level is quickly adjusted. Full recalibration requires 26 burst cycles to complete whereas fast recalibration requires only one cycle (Section 2.5). The time required for recalibration is dependent on the burst spacing setting ^G (Section 3.8).

Individual keys or groups of keys can be recalibrated with a single command depending on the current command scope. The time required to recalibrate many keys is not multiplicative; the cal process for multiple keys runs in parallel.

2.11 Boundary Error Reporting

See also commands ‘e’, page 22; ^N, page 26

Unlike guardband error reporting, boundary error reporting only works within the active ADC signal window segment in which the key's signal resides. Complex factoring of Cz and Offset are not required for these tests, and the tests do not require that the key be recalibrated to see the error condition.

Drift compensation can cause a key's reference level to move near to the border of the ADC's 8-bit signal window; this may make a key inoperable if the reference pegs near zero, depriving the signal of the ability to move further negative when a key is touched. Normally the reference level should be reasonably centered within the ADC's current range, i.e. at a level of about 128 decimal / 0x80 hex.

The truth logic for reference level drift error reporting is:

e/b2 = Reference > 191
e/b3 = Reference < 64

where e/b2 is command 'e' bit 2, and e/b3 is command 'e' bit 3. If either bit is set, the key should be recalibrated using command 'b'. Note that guardbanding errors (Section 2.8) also use these same bits for error reporting, but guardbanding does not usually affect these bits until after a recalibration.

Each Reference Boundary error will also appear as an error in a bitfield reported from command 'E'.

There is no mechanism by which keys can be made to automatically recalibrate if the reference drifts past a window boundary.

2.12 Device Status & Reporting

See also commands ‘7’, page 21; ‘e’, page 22; ‘E’, page *22*; ‘k’, page 22, ‘K’, page 23

The device can report on the general device status or specific key states including touches and error conditions, depending on the command used.

Usually it is most efficient to periodically request the general device status using command ‘7’ first, as the response to this command is a single byte which reports back on behalf of all keys. ‘7’ indicates if there are any keys detecting, calibrating, or in error.

If command ‘7’ reports a condition requiring further investigation, the host device can then use commands ‘e’, ‘E’, ‘k’ or ‘K’ to provide further details of the event(s) in progress. This hierarchical approach provides for a concise information flow using minimal data transfers and low host software overhead.

Bit 4 reports if there is a discrepancy between the eeprom and the Flash rom backup of the eeprom in case of data corruption. It is also set whenever a Setup parameter has changed but not yet been copied into Flash. See Section 4.6.

3 Circuit Operation

A reference QT60xx5 circuit is shown in Figure 3-1.

3.1 Part Differences

QT60xx5 parts use identical circuits and operate in identical manner in all respects, except that only the QT60645 can acquire 64 keys.

The QT60325 and QT60485 can only acquire 32 and 48 keys respectively, but both still use an 8x8 matrix; any 32 or 48 keys in the matrix can be used. Unused keys must be disabled by setting their burst length to zero via command ^F. These two parts have the higher order keys disabled (key numbers 32 and up, and 48 and up respectively). Any of these keys can be enabled by first disabling undesired lower order keys so that the limit is never exceeded during the setup process.

3.2 Matrix Scan Sequence

The circuit operates by scanning each key sequentially, key by key. Key scanning begins with location X=0 / Y=0. X axis keys are known as *rows* while Y axis keys are referred to as *columns*. Keys are scanned sequentially by row, for example the sequence Y0X0 Y0X1 .... Y0X3, Y1X0 Y1X1... etc.

Each key is sampled from 1 to 64 times in a burst whose length is determined by command ^F. A burst is completed entirely before the next key is sampled; at the end of each burst the resulting analog signal is converted to digital by an 8-bit ADC inside the part. The length of the burst directly impacts on the gain of the key; each key can have a unique burst length in order to allow tailoring of key sensitivity on a key by key basis.

3.3 Signal Path

Refer to Figures 1-4, 3-1, and 3-2. Further descriptions can be found in Section 0.

Charge gate. Only one X row is pulsed during a burst. Charge is coupled across a key's Cx capacitance from the X row to all Y columns. A particular key is chosen by gating the charge from a single Y column into a charge integrator. The gate is an 8:1 analog mux whose path is selected by lines YS0, YS1, and YS2; the gate is enabled by a pulse from the PLD. The charge integrator is described below.

Dwell time. The gate must be switched closed just prior to the rising edge of X and must be reopened just after X has finished rising, in order to capture the charge driven across key capacitance Cx. The delay time from the rise of X to the opening of the gate is known as the Y-sample *dwell time*. Dwell time duration has a dramatic effect on the suppression of signals due to moisture films as described in Section 3.13.

Charge neutralization. When X falls again, the charge across Cx must be neutralized. Without neutralization Cx charge would be sampled one time only and not again during operation. To accomplish this, the PLD always clamps all Y lines to ground except during the rise of X for the key being scanned.

Charge integrator. The first opamp is configured as an integrator with a reset switch; capacitor Cs performs the charge integration function. Capacitor Ca acts to absorb charge momentarily before the opamp can react to absorb the charge across Cs; the value of Ca is not critical. A P-channel JFET resets Cs between bursts. The opamp's output swings negative, and as a consequence a negative power supply is required.

Charge cancellation. Two Cz capacitors are used to cancel charge across Cs in stepwise fashion in order to increase signal range. These capacitors can switch during the course of a burst to reduce the final output of the amplifier chain, preventing early signal saturation due to large keys (high Cx) and/or long burst lengths. The Cz's are normally driven to +5V when not in use; switching them to ground causes a step subtraction of charge from the integrator.

Signal amplification; offset. At the burst's end the integrator output is amplified by an opamp circuit which includes an offset from an R2R ladder DAC driven off the X drive lines. The offset from the DAC repositions the output of the amplifier chain to coincide as closely as possible with the center span of the ADC, which can convert voltages between Vss and Vdd.

Burst timing. Figure 3-2 relates to a particular key being addressed by an X row line and gate control lines YSn. At the end of the burst, the X pins are used to drive the external R2R ladder network which generates an offset to the amplifier chain. It is during this period that the amplifier stabilizes and the signal is sampled by the QT60xx5.

Signal gain. Gain is directly controlled by burst length, amplifier gain, and the value of Cs. Burst length can be adjusted on a key by key basis whereas Av and Cs are fixed for all keys. See Section 3.6. The detection threshold setting also factors directly into key sensitivity.

3.4 'X' Electrode Drives

The 'X' lines are directly connected to the matrix without buffering. The positive edges of these signals are used to create the transient field flows used to scan the keys. Only one X line is actively driving the matrix for scanning purposes at a time, and it will pulse for a 'burst length' for each key as determined by the 'Burst Length' Setups parameter (see command ^F, page 24 and Section 3.6).

A 22V10 type CMOS PLD is used in part to create the short gate dwell times necessary to enhance moisture suppression and to control the transfer switch. The PLD takes as its input the logical OR of all 'X' lines, combines the result with the 'YCn' switching signals, and with the aid of an RC time constant creates the required dwell time enable on the 'Y' switch enable (QS3251 pin 7). The code for the PLD can be found in Section 6.

3.4.1 RFI From X Lines

X drive lines will radiate a small amount of RFI. This can be attenuated if required by using series resistor in-line with each X trace; the resistor should be placed near to the QT60xx5. Typical values can range from 100 to 500 ohms. Excessive amounts of R will cause a counterproductive drop in signal strength. RC networks can also be used as shown in Figure 4-4.

Inserted resistors in the X lines also have the positive effect of limiting ESD transient currents (Section 3.22).

3.4.2 Noise Coupling Into X lines

External noise, sometimes caused by ground bounce due to injected line noise, can couple into the X lines and cause signal interference in extreme cases. Such noise can be readily suppressed by the use of series resistors as

Figure 3-1 Recommended Circuit Diagram

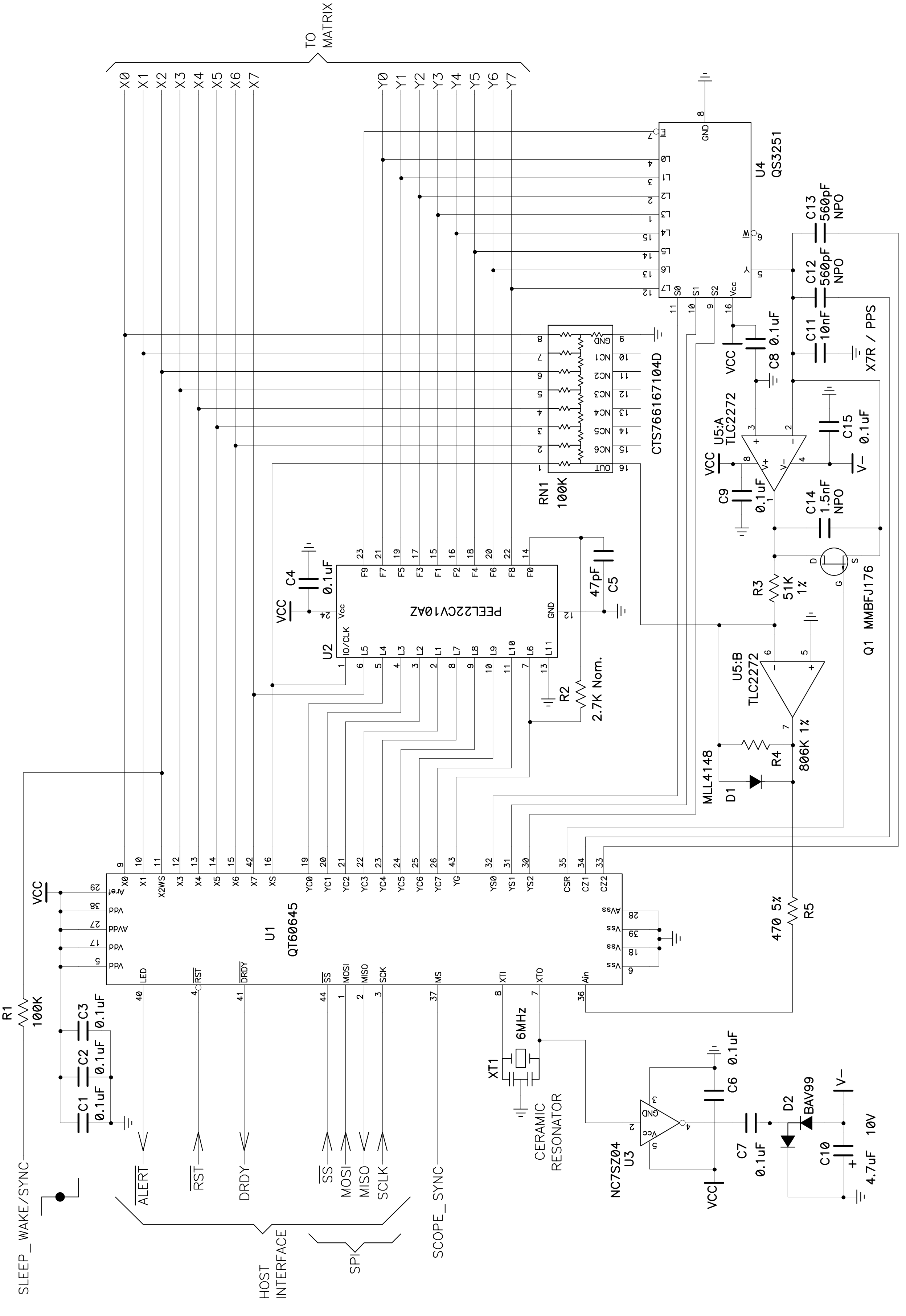
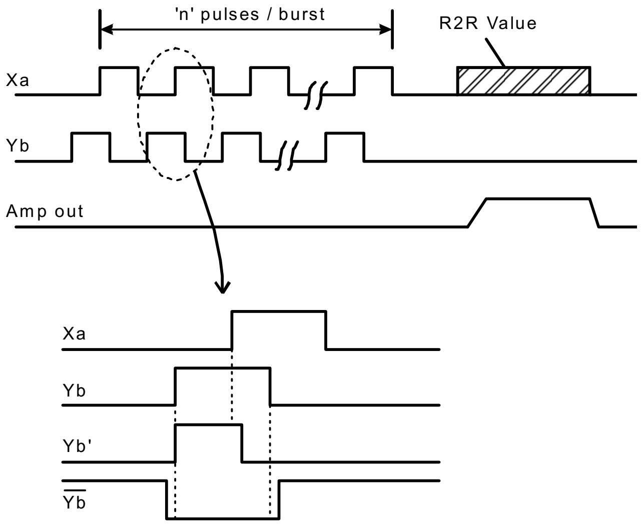


Figure 3-2 Relationship of X and Y signals



described above. Adding a small capacitor to the matrix line on the QT60xx5 side of the R, for example 100pF to ground near the QT60xx5, will greatly help to reduce such effects.

3.5 'Y' Gate Drives

There are 8 'Y' gate drives (YC0..YC7) which are active-high; only one of these lines is used during a burst for a particular key. These lines are used to control the PLD to ground all unselected 'Y' lines, making them low impedance. The selected 'Y' line in the matrix remains unclamped by the PLD during the rising edge of the 'X' drive line, during the time that the coupled charge from a single key is fed to the charge integrator via the 8:1 analog mux.

There are also 3 Y-encoded lines YS0..YS2 which select the correct switch to actuate in the analog mux for the desired 'Y' line. Line 'YG' from the controller acts to trigger the PLD's pulse generation circuit, whose pulse width following the rise of an 'X' line is dependent on an RC time constant. This pulse, 'YE', drives the enable pin of the QS3251 mux low (switch on) just before a positive-going 'X' drive pulse, and high again (switch off) just after the 'X' drive pulse. The time from the rising edge of an 'X' signal to the rising edge of 'YE' is referred to as the dwell time, and this parameter has a direct effect on the ability of the circuit to suppress moisture films (see Sections 3.9 and 3.13).

After the 'YE' pulse has ceased, the controller and circuit act to ground all 'Y' lines via the PLD just before the 'X' drive signal goes low; this restores the charge across the matrix keys to a null state, making them ready for another sample.

3.5.1 RFI From Y Lines

Y lines are 'virtual grounds' and do not radiate a significant amount of RFI; in fact, they act as 'sinks' for RFI emitted by the X lines since they are virtual grounds. Series-R in the Y lines is not required for RFI suppression, and in fact series-R can introduce cross-talk among keys.

3.5.2 Noise Coupling Into Y Lines

External noise, sometimes caused by ground bounce due to injected line noise, can couple into the Y lines and cause signal interference in extreme cases. Such noise can be readily suppressed by adding a 100pF capacitor from each Y line to a ground plane near the QT60xx5.

3.6 Burst Length & Sensitivity

See also Command ^F, page 24

The signal gain in volts / pF of Cx for each key is controlled by circuit parameters as well as the burst length.

The burst length is simply the number of times the charge-transfer ('QT') process is performed on a given key. Each QT process is simply the pulsing of an X line once, with a corresponding Y line enabled to capture the resulting charge passed through the key's capacitance Cx.

QT60xx5 devices use a finite number of QT cycles which are executed in a short burst. There can be from 1 to 64 QT cycles in a burst, in accordance with the list of permissible values shown in conjunction with command ^F, page 24. If burst length is set to zero, the burst is disabled but its time slot in the scanning sequence of all keys is preserved so as to maintain timing.

Increasing burst length directly affects key sensitivity. This occurs because the accumulation of charge in the charge integrator is directly linked to the burst length. The burst length of each key can be set individually, allowing for direct digital control over the signal gains of each key individually.

Apparent touch sensitivity is also controlled by the Negative Threshold setting (Section 2.1. Burst length and negative threshold interact; normally burst lengths should be kept as short as possible to limit RF emissions, but the threshold setting should normally be kept above a setting of 6 to limit false detections. The detection integrator can also prevent false detections at the expense of slower reaction time (Section 2.6).

3.7 Intra-Burst Spacing

See also Command ^M, page 26

The time between X drive pulses during a burst is the intra-burst pulse spacing. This timing has no noticeable effect on performance of the circuit, but can have an impact on the nature of RF spectral emissions from the matrix panel. The setting of this function can be from 2μs through 10μs, loosely corresponding to fundamental emission frequencies from 500kHz and 100kHz respectively.

Longer spacings require more time to execute and can limit the operational settings of burst length and/or burst spacing (Section 5.7).

The intra-burst QT spacing has no effect on sensitivity or water film suppression and is not particularly important to the sensing function other than described above.

3.8 Burst Spacing

See also Command ^G, page 24

The interval of time from the start of one burst to the start of the next is known as the *burst spacing*. This is an alterable parameter which affects all keys.

Shorter spacings result in faster response time, but due to increasing timing restrictions at shorter spacings burst lengths are restricted, limiting the amount of gain that can be obtained; see Section 5.7. Conversely longer spacings permit higher burst lengths but slow down response time.

Spacings from 250µs to 2ms are available.

3.9 PLD Circuit and Charge Sampler

The PLD should be a CMOS 22V10 type having no internal pullup or bus-keeper resistors in order to limit leakage current. ICT's PEEL22CV10AZ is a good example of a device that works well, and code for this part can be found in Section 6.

The PLD performs two functions: Y line clamping and transfer switch gating.

The PLD acts as a Y-line clamp to ground for each Y line that is not being sensed. The charge integrator should only receive charge starting just before an X line goes high, to a point just after the transition (the X-Y 'dwell time'). At all other times the Y line can be simply clamped to ground. It is also an essential function of the PLD to restore a neutral charge across the Cx of each key during the negative transition of X lines; without a charge restoration function the charge-transfer principle would cease to function after a single X pulse, and multi-pulse bursts would not be possible.

The PLD also acts to generate a pulse that sets the dwell time for the QS3251 8:1 charge sampler switch. A simple RC network plus PLD logic controls the QS3251 gate pin E' starting from when line YG becomes active to a time after X7 or XS transition high. XS is the logical-OR of X0..X6; X8 and XS are OR'd together in the PLD so that any single X line can trigger the timing network.

The X-Y dwell time can be measured with an oscilloscope by timing the interval from XS or X8 to output F9 of the 22V10. A dwell time of 70ns - 90ns works very well to suppress the effects of moisture films on the surface of the control panel. Longer times are acceptable if surface moisture is not normally present. R2 and/or C5 in Figure 3-1 can be adjusted to obtain the desired dwell time.

While the QS3251 is gated by the signal on its E' pin from the PLD, the actual switch being controlled is determined by the YS0, YS1, YS2 lines from the QT60xx5. The timing of these three signals is non-critical.

3.10 Opamps

The amplifier chain should be configured as shown in Figure 3-1. The opamps should have a GBW product of at least 2MHz, have rail-rail CMOS outputs, and be able to operate from split rail supplies. To eliminate leakage current issues the amplifier should be a JFET or CMOS input type only. A TI TLC2272 type opamp is a good example of the type of device which should be employed.

The first opamp stage is configured as a charge integrator, whose output ranges within 0V to -2.5V. A p-jfet controlled from the QT60xx5 is used as a reset switch for the integration capacitor C14. Since most opamps are not fast enough to integrate the nanosecond duration transient charge pulses coming from the Y lines and the switched Cz capacitors, a large, non-critical capacitor C11 is used to temporarily store transient charge until the opamp can assimilate it over the following microseconds.

The second stage opamp must invert the first opamp output in order to provide a positive-going signal to Ain of the QT60xx5. This stage is also used to facilitate the introduction of offset from the R2R network (Section 3.12).

The second stage must be clamped with a diode as shown so that negative excursions of the amplifier do not under-drive the Ain pin of the device. An output resistor further limits possible Ain currents. If this clamping is not included, there can be high drain currents from Ain which can lead to total device latchup requiring that power to be cycled to restore operation.

3.11 Sample Capacitors

Charge sampler capacitor Cs (C14 in Figure 3-1) should be a 1.5nF NPO or X7R 5% type for thermal stability reasons. The two Cz capacitors C12 and C13 should be 560pF NPO ceramic 5%. The transient charge absorber C11 should be a 100nF X7R type, 10%.

More information on how the Cs and Cz capacitors function is described in Section .

The values of capacitance should not be altered from the schematic shown; value changes can cause acquisition gaps to occur which can result in keys that cannot calibrate for unknown reasons.

3.12 R2R Resistor Ladder

The R2R resistor ladder network (RN1 in Figure 3-1) should have a listed value of 100K ohms and a precision of 7 bits or better. The ladder directly connects to the summing junction of the second opamp; it is used to offset the analog signal down with increasing binary value. The R2R offset drive value is determined individually for each key during recalibration by an algorithm which seeks to center the analog signal Ain at 2.5 volts. The binary value only changes when a key is recalibrated or after powerup during the normal startup calibration cycle; drift compensation does not change R2R drive.

The R2R is driven by the matrix X lines; since the analog signal is only read after the completion of each burst, the dual-use of X lines does not pose a conflict.

The rated resistance of an R2R ladder is also its Thevenin equivalent resistance which affects the scaling of the offset injected into the amplifier, in terms of mV/bit. The scaling of offset injection also affects the crossover points for the switching of each Cz capacitor. If during the calibration cycle the R2R network is found to not provide enough offset to bring the signal to the midpoint of the ADC's range, a Cz capacitor is switched in to create an additional offset.

If the R2R drive value and Cz values are not properly matched, the circuit may not be able to converge on all calibration points, i.e. there will be acquisition 'holes'. This will happen if the Cz cancellation voltage step is too large with respect to the amount of full-scale influence of the R2R ladder on the analog offset. It is not recommended that the reference circuit shown in Figure 3-1 be altered without a specific reason.

3.13 Water Film Suppression

Water films on the user surface can cause problems with false detection under certain conditions. Water films on their own will not normally cause false detections. The most common problem occurs when surface water bridges over 2 or more keys, and a user touches one of the keys and the water film causing an adjacent key to also trigger. Essentially, the water film transports the touch contact to adjacent keys.

The recommended circuit suppresses water coupling by reducing the sample dwell time. Short dwell times reduce the amount of charge collected via resistive water films, i.e. they suppress charge from areas adjacent to the scanned key. This effect has nothing to do with the frequency of the burst itself.

To accomplish this, a CMOS PLD is configured with a simple timing circuit to shorten the dwell time of the Y gate as described in Section 3.9. The RC of this circuit should be adjusted to provide a timing dwell delay of around 75ns +/-20%. Shorter dwell times can be used, but these can cause excess suppression of human touch as well. If series resistors are used in line with the X and Y matrix lines for noise and ESD suppression (Section 3.22), excessively short dwell times can seriously affect signal gain.

Source code for one type of recommended 22V10 can be found in Section 6. The 22V10 should have conventional CMOS I/O structures without 'bus-keepers' or pullup resistors in order to work optimally.

Mechanical measures can also be used to suppress key cross-coupling, for example one can use raised plastic barriers between keys, or placing keys in shallow wells to lengthen the electrical path from key to key.

3.14 Reset Input

The RST' pin can be used to reset the device to simulate a power down cycle, in order to bring the part up into a known state should communications with the part be lost. The pin is active low, and a low pulse lasting at least 10µs must be applied to this pin to cause a reset.

To provide for proper operation during power transitions the devices have an internal brown-out detector set to 4 volts.

A reset command, 'r', is also provided which generates an equivalent hardware reset (page 28).

3.15 Oscillator

The oscillator can use either a quartz crystal or a ceramic resonator. In either case, the XTI and XTO must both be loaded with 22pF capacitors to ground. 3-terminal resonators having onboard ceramic capacitors are commonly available and are recommended. An external TTL-compatible frequency source can also be connected to XTI; XTO should be left unconnected.

The frequency of oscillation should be 6MHz +/-2%.

3.16 Startup / Calibration Times

The QT60xx5 requires initialization times as follows:

1. From very first powerup to ability to communicate:
   2,000ms (One time event to initialize all of eeprom)
2. Normal cold start to ability to communicate:
   70ms (Normal initialization from any reset)
3. Calibration time per key vs. burst spacings:
   spacing = 250µs: 425ms
   spacing = 300µs: 510ms
   spacing = 400µs: 680ms
   spacing = 500µs: 850ms
   spacing = 1ms: 1,700ms
   spacing = 2ms: 3,400ms

   To the above, add 2,000ms or 70ms from (1) or (2) for the total elapsed time from reset to ability to report key detections.

Keys that cannot calibrate for some reason require 5 calibration cycles before they report as errors. However, the device can report back during this interval that the key(s) affected are still in calibration via status function bits.

3.17 Sleep_Wake / Noise Sync

The Sleep_wake and Noise sync features depend on the use of pin X2WS as an input. To prevent interference with scan line X2 during acquisitions, a resistor equal to the rating of the R2R ladder (i.e. 100K) must be used in series. The Sleep and Sync features can be used simultaneously; the part can be put into Sleep mode, but awakened by a noise sync signal which is gated in at the time desired.

Sleep mode: *See also command 'Z', page 28.*

The device can be put into an ultra low-power sleep mode using the 'Z' command. When this command is received, the Sleep line must be placed immediately thereafter into a logic-high state. The part will complete an ongoing burst before entering Sleep. The part can be awakened by a low transition on the X2WS pin lasting at least 5µs. One convenient way to wake the part is to connect pin X2WS to MOSI via the 100K resistor, and have the host send a null command to the device. The part will wake and the null command will not be processed. The MOSI line in turn requires a pullup resistor to prevent the line from floating low and causing an unintentional wake from sleep.

During Sleep the oscillator is shut down, and the part hibernates with microamp levels of current drain. When the part wakes, the part resumes normal functionality from the point where it left off. It will not recalibrate keys or engage in other unwarranted behavior.

Just before going to sleep the part will respond with a response of 'Z'. In slave-only SPI mode (see Section 4.3), the SS line must be floated high by the host as soon as it receives this response; if SS does not float high, sleep will fail and the device will instead completely reset after about 2 seconds. Upon waking the part will issue another 'Z' byte back to the host.

Noise sync: *See also command ^W, page 29*.

External fields can cause interference leading to false detections or sensitivity shifts. The strongest external fields usually come from AC power. RF noise sources are heavily suppressed by the low impedance nature of the QT circuitry itself.

External noise only becomes a problem if the noise is uncorrelated with signal sampling; uncorrelated noise can cause aliasing and beat effects in the key signals. To suppress this problem the devices feature a noise sync input which allows bursts to synchronize to the noise source. This same input can also be used to wake the part from a low-power Sleep state.

The device's bursts can be synchronized to an external source of repetitive electrical signal, such as 50Hz or 60Hz, or possibly a video display vertical sync line, using the Sleep_wake / Noise sync line. The noise sync operating mode is set by command ^W. This feature allows dominant external noise signals to be heavily suppressed, since the system and the noise become synchronized and no longer beat or alias with respect to each other. The sync occurs only at the burst for key 0 (X0Y0); the device waits for the sync signal for up to 100ms after the end of a preceding full matrix scan (after key #63), then when a negative sync edge is received, the matrix is scanned in its entirety again.

The sync signal drive should be a buffered logic signal, or perhaps a diode-clamped signal, but never a raw AC signal from the mains.

Since Noise sync is highly effective yet simple and inexpensive to implement, it is strongly advised to take advantage of it anywhere there is a possibility of encountering electric fields. Quantum's QmBtn software can show signal noise caused by nearby AC electric fields and will hence assist in determining the need to make use of this feature.

If the sync feature is enabled but no sync signal exists, the sensor will continue to operate but with a delay of 100ms from the end of one scan to the start of the next, and hence will have a slow response time.

3.18 LED / Alert Output

Pin 40 is designed to drive a low-current LED, 5mA maximum, in an active-low configuration. The LED will glow brightly (i.e. pin 40 will be solid low) during calibration of one or more keys, for example at startup. When a key is detected, the LED will pulse low for the duration of the burst during which the key is being sensed, i.e. with a very low duty cycle. Each additional key being detected will also create a low pulse for that key's burst. During all other times, the LED pin will be off (high).

This pin can be used to alert the host that there is key activity, in order to further limit the amount of communication between the device and the host. The LED / Alert line should ideally be connected to an interrupt pin on the host that can detect a negative edge, following which the host can proceed to poll the device for key activations.

This pin also pulls low if there is a key error of any kind.

Note that in sleep mode if the LED was on prior to sleep, it will remain on during sleep.

3.19 CSR Drive Polarity

See also Command ^S, page 29

The polarity of the Cs integrator capacitor reset drive can be set for active high or active low operation using command ^S. In the reference circuit show in Figure 3-1, the JFET will reset Cs when the drive signal is low, so ^S should be set to '0'.

This feature allows for operation with other types of semiconductor switches or circuit topologies which may require a different Cs reset control polarity.

3.20 Oscilloscope Sync

See also Command ^R, page 29

The 'MS' pin 37 can output an oscilloscope sync signal which is a positive pulse that brackets the burst of a selected key. This feature is controlled by the ^R command. More than one burst can output a sync pulse, for example if the scope of the command when set is a row or column, or is all keys. The ^R command is volatile and does not survive a reset or power down.

This feature is invaluable for diagnostics; without it, observing signals clearly on an oscilloscope for a particular burst is nearly impossible.

This pin is also used as a SPI mode select pin. In order to prevent a shorted output when the oscilloscope sync is enabled, the MS pin should only be connected to ground or Vdd via a ≥10K resistor.

This function is supported in QmBtn PC software via a checkbox.

3.21 Power Supply and PCB Layout

Vdd should be 5.0 volts +/- 5%. This can be provided by a common 78L05 3-terminal regulator. LDO type regulators are usually fine but can suffer from poor transient load response which may cause erratic signal behavior.

If the power supply is shared with another electronic system, care should be taken to assure that the supply is free of digital spikes, sags, and surges which can adversely affect the circuit. The devices can track slow changes in Vcc depending on the settings of drift compensation, but signals can be adversely affected by rapid voltage steps and impulse noise on the supply rail.

Supply bypass capacitors of 0.1uF to a ground plane should be used near every supply pin of every active component in the circuit.

Vee is a negative supply which can range from -3V to -5V. It does not need to be regulated but should be well filtered and free from externally induced fluctuations. The circuit of Figure 3-1 shows a simple, inexpensive charge-pump which is driven from resonator pin XTO to generate Vee.

The current requirements of the circuit are approximately 20mA / Vdd, 4mA / Vee when running.

PCB layout: The PCB layout should incorporate a ground plane under the entire circuit; this is possible even with 2-layer boards. The ground plane should be broken up as little as possible. Internal nodes of the circuit can be quite sensitive to external noise and the circuit should be kept away from stray magnetic and electric fields, for example those emanating from mains power components such as transformers and power capacitors. If proximity to such components is unavoidable, an electrostatic shield should be considered. The Sync feature (Section 3.17) can be invaluable in reducing these types of noise sources, but only up to a point.

Sample layout artwork is available from Quantum on request.

3.22 ESD / Noise Considerations

In general the QT60xx5 will be well protected from static discharge during use by the overlying panel. However, even with a dielectric panel transients currents can still flow into scan lines via induction or in extreme cases, dielectric breakdown. Porous or cracked materials may allow a spark to tunnel through the panel. In all cases, testing is required to reveal any potential problems. The devices have diode protected pins which can absorb and protect the device from most induced discharges, up to 5mA.

The X lines are not usually at risk during operation, since they are low-resistance output drives. The YCn lines are not directly connected to the matrix and so are not at risk. However the PLD and the QS3251 are connected to the Y lines and may require additional ESD protection.

Diode clamps can be used on the X and Y matrix lines if desired. The diodes should be high speed / high current types such as BAV99 dual diodes, connected from Vdd to Vss with the diode junction connected to the matrix pin. Diode arrays can also be used.

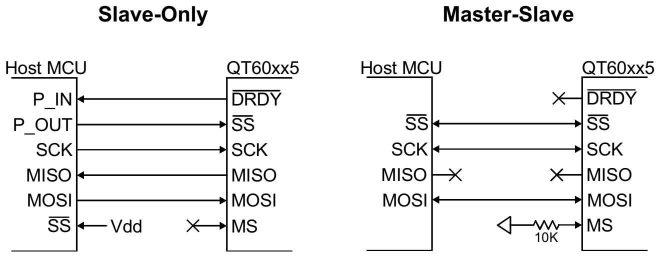
Capacitors placed on the X and Y matrix lines can also help to a limited degree by absorbing ESD transients and lowering induced voltages. Values up to 100pF can be used without causing circuit problems.

The circuit can be further protected by inserting series resistors into the X and/or Y lines to limit peak transient current. Values up to 500 ohms can be used in most cases, but if the dwell time is short this resistance can cause a reduction in gain. RC networks as shown in Figures 4-4 and 4-5 can provide enhanced protection against ESD while also limiting the effects of external EMI should this be a problem.

External field interference can occur in some cases; these problems are highly dependent on the interfering frequency and the manner of coupling into the circuit. PCB layout (Section 3.21) and external wiring should be carefully designed to reduce the probability of these effects occurring.

SPI data noise: In some applications it is necessary to have the host MCU at a distance from the sensor, perhaps with the interface coupled via ribbon cable. The SPI link is particularly vulnerable to noise injection on these lines; corrupted or false commands can be induced from transients on the power supply or ground wiring. Bypass capacitors and series resistors can be used to prevent these effects as shown in Figures 4-4 and 4-5.

Figure 4-1 SPI Connections



4 Serial Interface

QT60xx5 parts use an SPI serial interface to a host MCU. This port uses a protocol described in Section 5.

4.1 Serial Port specifications

QT60xx5's use an SPI synchronous serial interface with the following specifications at 6MHz oscillator frequency:

Max clock rate, Fck	1.5MHz
Data length	8 bits
Host command space, Tcm	≥ 50µs
Response delay to host, Tdr1	Table 4-1, also, Sec. 7
Drdy delay from response, Tdr2	1µs to 1ms
Multi-byte return spacing, Tdr3	15µs to 2ms

The host can clock the SPI at any rate up to and including the maximum clock rate Fck. The maximum clock rate of the part in Master mode is determined by Setup ^Q.

The part can operate in either master-slave mode or slave-only mode, and is thus compatible with virtually all SPI-capable microcontrollers.

The SPI interface should not be used over long distances due to problems with signal ringing and introduced noise etc. unless suitably buffered or filtered with RC networks as shown in Figures 4-4 and 4-5. Slower data rates with longer RC timeconstants will provide enhanced resistance to noise and ringing problems.

Conversion to asynchronous UART format can be accomplished by using a microcontroller with conversion firmware. Using such a conversion device the part can communicate with Quantum's QmBtn PC software. Consult Quantum for details.

4.2 Protocol Overview

The SPI protocol is based entirely on polled data transmission, that is, the part will not send data to the host of its own volition but will do so only in response to specific commands from the host.

Run-time data responses, such as key detection or error information, requires simple single-byte functions to evoke a response from the part.

Setup mode interactions mostly use 2-byte functions from the host to cause the part to alter its behavior; these functions also cause writes to the internal eeprom.

The concept of 'scope' is used to allow functions to operate on individual keys or groupings of keys. The scope of subsequent functions can be altered by short initial scope instructions.

See Section 5 for protocol details.

4.3 SPI Slave-Only Mode

Refer to Figures 4-1 and 4-2. Select Slave-only by floating Pin 37 (MS) or tying high via a ≥10K resistor. Pin 37 also functions as an oscilloscope sync output (Section 3.20) and so should never be tied directly to either supply rail. In Slave mode the host must always be in Master mode, as it controls all SPI activity including the clocking of the interface in both directions. Unlike hardware SPI slaves, QT60xx5's need processing time to respond to functions. DRDY' is used to let the host know when data is ready for collection; it indicates to the host when data is

Figure 4-2 SPI Slave-Only Mode Timing

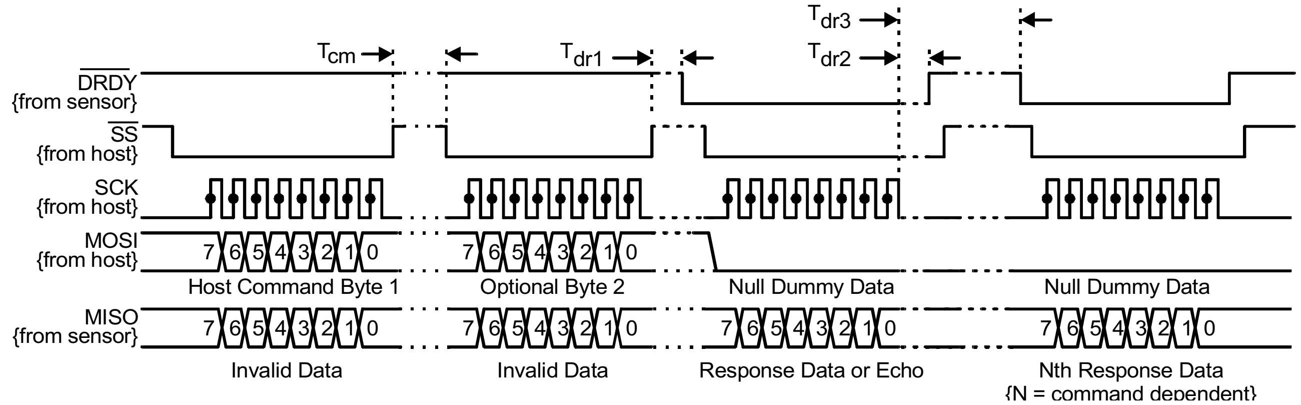
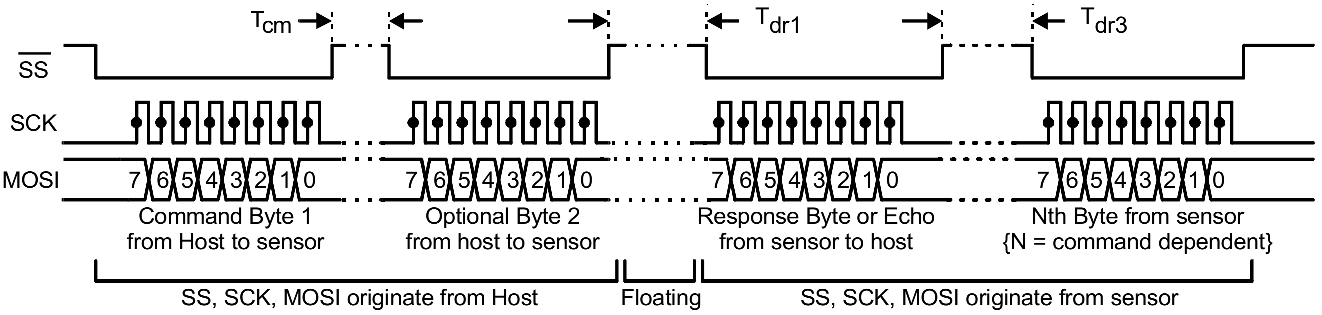


Figure 4-3 SPI Master/Slave Mode Timing


Tcm
Tdr1
Tdr3
SS
SCK
MOSI
7 6 5 4 3 2 1 0
Command Byte 1 from Host to sensor
Optional Byte 2 from host to sensor
Response Byte or Echo from sensor to host
Nth Byte from sensor {N = command dependent}
SS, SCK, MOSI originate from Host
Floating
SS, SCK, MOSI originate from sensor

ready in response to a command so that the host can clock over the data.

This mode requires 5 signals to operate:

MOSI - Master out / Slave in data pin; used as an input for data from the host at all times. This pin should be connected to the MOSI pin of the host device.

MISO - Master in / Slave out data pin; used as an output for data to the host at all times. This pin should be connected to the MISO pin of the host device.

SCK - SPI clock - input only clock pin from host. The host must shift out data on the falling edge of SCK; the QT60xx5 clocks data in on the rising edge of SCK. **Important note:** SCK must idle low just before and after SS' transitions either up or down, or the transmission will fail; between bytes SCK should always idle low. SCK should never float.

SS' - Slave select - input only; this pin acts as a framing signal to the sensor from the host. This line must go low just before and during reception of data from the host. It must not go high again until the SCK line has returned low; during data or echo response it must not go high until after the host has sensed that DRDY' has gone high from the device. This pin must idle high. The SS' pin has an internal pullup resistor inside.

DRDY' - Data Ready - active-low - indicates to the host that the part is ready to send data back subsequent to a command from the host. This pin idles high. The DRDY' pin has an internal pullup resistor inside.

Internal pullup resistors note: The internal pullup resistors can range from 35k to 120k ohms. If RC filtering is used on the SPI lines per Figure 4-4, this resistance may not be low enough to ensure adequate signal risetime and may need to be augmented with external 10k pullups.

The host must wait until DRDY' goes low before an SPI transfer to retrieve data. For multi-byte responses, the host must observe DRDY' to see when it goes high again after each data byte, then low again, before executing another

Table 4-1 Typical DRDY (Tdr1) Response Delays (Burst Length = 12)

	Burst Spacing					
Function Type	**250µs**	**300µs**	**400µs**	**500µs**	**1ms**	**2ms**
Setup - Put (affect 1 key)	10ms	10ms	10ms	10ms	10ms	10ms
Setup - Put (affect 8 keys)	40ms	40ms	40ms	40ms	40ms	40ms
Setup - Put (affect 64 keys)	300ms	300ms	300ms	300ms	300ms	300ms
Lock Reference Levels ('L') command	800ms	800ms	800ms	800ms	800ms	800ms
Calibrate command (all keys)	3ms	2.7ms	2.5ms	2.5ms	2ms	2ms
Get key errors (E), Get keys pushed (K)	1ms	1ms	800us	800us	800us	800us
All other commands	400us	300us	300us	300us	300us	300us

Figure 4-4 Filtering SPI Slave-Only Connections

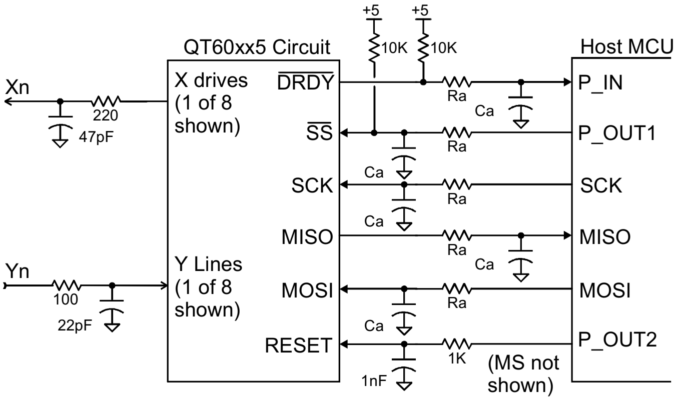
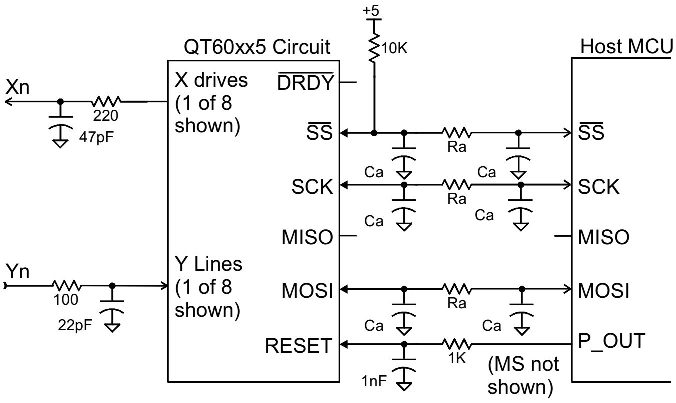


Figure 4-5 Filtering SPI Master-Slave Connections



Recommended Values of Ra & Ca for Figures 4-4 and 4-5

SPI Clock Rate	Ra	Ca
1.5MHz	680	100pF
375kHz	1,000	270pF
93.75kHz	2,200	470pF
46.875kHz	2,200	1nF

transfer to get the next data byte. The host should send null bytes (0x00) to retrieve data.

If the DRDY' line does not go low after a command, the command was not properly received or it was inappropriate. The delay to DRDY' low depends on how many bytes of data are being loaded into eeprom; Table 4-1. Absolute worst case delays are found in Section 7; these timings occur only rarely, for example if the device happens to be busy with adjacent key suppression calculations, which occurs only at the moment when a key is first detected.

A typical Slave-only function sequence is as follows:

1) The host pulls SS' low, then transfers a command to the sensor. The host then releases SS' to float high. DRDY' is unaffected in this step.

2) For 2-byte functions, (1) is repeated with a ≥50μs delay.

3) When the sensor has the command echo or requested data ready to send back to the host, it loads it into its SPI register and pulls DRDY' low.

4) The host detects that the sensor has pulled DRDY' low and in turn the host pulls SS' low.

5) The host obtains the byte from the sensor by transmitting a dummy byte (0x00) to the sensor.

6) The sensor releases DRDY' to float high.

7) After the host detects that DRDY' has floated high the host must allow SS' to also float high.

8) For multi-byte responses, steps (3) through (7) are repeated until the return data is completely sent.

Note that the host must release the SS' line in step (7) even between multiple byte responses because the QT60xx5 waits for the SS' line to return high before signaling that the next byte is ready for collection.

Note also that the host should check the DRDY' line and wait for it to go high before transmitting another byte. Until the DRDY' line is released the sensor is still processing a data return, even if the complete response data has been fully transferred; the sensor may still be busy when the host finishes the byte transfer and may not be able to digest a new command immediately.

4.4 SPI Master-Slave Mode

Refer to Figures 4-1 and 4-3. In Master-Slave mode the host and the sensor take turns being Master, with the host always leading off in Master mode during an exchange. The current Master always controls all 3 signal lines. The sensor takes a variable amount of time to respond to the host, depending on the nature of the function and its current and pending tasks. SPI Master/Slave mode is selected by tying Pin 37 (MS) low via a 10K resistor. ***Pin 37 also functions as an oscilloscope sync output (see Section 3.20 and command ^R, page 29) and so should never be tied directly to either supply rail.*** The host, like the sensor, must idle in slave mode when not sending a command.

Master/Slave requires 3 signals to operate:

MOSI - Master out / Slave in data pin - bidirectional - an input pin while the host is transmitting data; an output when the sensor is transmitting data. The MOSI of the host and slave should be tied together. The MISO lines are not used on either part and should be left open.

SCK - SPI clock - bidirectional - an input pin when receiving data; an output pin when sending. The host must shift out data on the falling edge of SCK; the QT60xx5 clocks data in on the rising edge of SCK. **Important note:** SCK from the host must be low before asserting SS' low or high at either end of a byte or the transmission will fail. SCK should idle low; if in doubt, a 10K pulldown resistor should be used. When the sensor returns data it becomes the Master; data is shifted out by it on the falling edge of SCK and should be clocked in by the host on the rising edge.

SS' - Slave select - bidirectional framing control. When the sensor is in slave mode, this pin accepts the SS' control signal from the host. In either data direction, SS' must go low before and any during data transfer; it should not go high again until SCK has returned low at the end of a byte. In Master mode the sensor asserts control over this line, to make the host a slave and to frame the data. This line must idle high; the part includes an internal pullup resistor and should be floated during idle times.

Internal pullup resistor note: The internal pullup resistor on SS' can range from 35k to 120k ohms. If RC filtering is used on the SPI lines per Figure 4-5, this pullup resistance may not be low enough to ensure adequate signal risetime and may need to be augmented with external 10k pullups.

A command may consist of one or two bytes with a ≥50µs delay between command bytes. At the end of a full command, the Master must go into Slave mode to await a response from the sensor.

The sensor may take some time to process the host command and respond. When it does so, it asserts SS' low and begins clocking its data. For multi-byte responses, the bytes will be sent at intervals which may be somewhat irregular depending on the request and the processing load of the sensor. The host must be prepared to accept the sensor data as it comes or there can be a data overrun in the host. *If the data returns too quickly for the host to accept it, the SPI clock rate should be lowered*.

A typical Master-Slave function sequence is as follows:

1) Host enters Master mode. The sensor is already in Slave mode.
2) The host pulls SS' low, then transfers one byte of command to the sensor via MOSI, then releases SS' to float high again.
3) For 2-byte functions, (2) is repeated with ≥50µs spacings between bytes.
4) The host immediately places its SPI port into Slave mode, floating SCK and MOSI'; SS' stays floating.
5) When the sensor has a command echo or data to send back, it puts its SPI register in Master mode, taking control over MOSI and SCK. SS' remains floating.
6) The sensor pulls SS' low, then clocks out its response byte to the host, then floats SS' high again.
7) The sensor repeats (6) as necessary for multiple byte responses.
8) The sensor returns to slave mode.

After the transmission sequence, the SPI lines float high or are left to float in an indeterminate state (MOSI) until the next transmission sequence is initiated by the host. *The host should wait for ≥1ms after a sequence before initiating another transmission sequence.*

4.5 Sensor Echo and Data Response

The devices respond to each and every valid command from the host with at least one return byte. In the case of functions that do not send data back to the host, the part returns the command itself as an echo, but only after the function has been processed to completion; this also holds for 2-byte functions where the second byte is an operand: in these cases the return byte is an echo of the command, not the operand.

Commands that return data do not send back a command echo. If desired, the command byte can be verified via the 'l' (lowercase L) function; see page 28.

The host should not transmit a new command until the last command has been processed and responded to completion, plus 1ms.

Commands that are not recognized are ignored, and the host should monitor for timeouts to detect these conditions. If this occurs a new command should not be sent until the specified timeout condition has expired.

The maximum timings shown in Table 4-1 and Section 7-5 are guaranteed provided that the part is operating within its burst timing limitations described in Section 5.7. If the burst timing is in violation, the response time to a command may take considerably longer.

4.6 Eeprom Corruption

The device stores its Setup data in an internal eeprom which can be readily altered via Put mode commands. Sometimes noise on Vdd, the SPI lines or Reset pin can cause eeprom corruption which can be difficult or inconvenient to correct.

The device should always be left in Get mode to prevent spurious commands from corrupting the eeprom. The Get command should ideally be repeated every second or so to ensure that if noise on the SPI lines causes a false Put mode command that it does not last long. Preferably, the 'l' command (lowercase 'L') should be used to verify that the Put command has succeeded.

Flash backup: The part backs up the entire eeprom array into onboard Flash rom after one or more Setup write commands have been issued and the part is then reset. During normal operation the part constantly compares the Flash area with the eeprom array to ensure the two sections match. If an eeprom error is detected, the device sets an error flag (bit 4) in the general device status byte (Command '7', page 21) which can be read by the host device. The LED output also becomes active. If the bit 4 error flag is set, the host should immediately induce a device reset.

Bit 4 is also set if an intentional write has been made to eeprom, but not yet copied into Flash via the reset process. It is perfectly acceptable to continue altering any number of Setup parameters prior to doing the reset, ignoring this bit.

During power up or after a reset, the device compares the Flash area with eeprom, and if there is a discrepancy the eeprom is refreshed from Flash, unless an intentional write was detected in which case the Flash is updated from the eeprom. As intentional writes in Put mode should only occur during manufacture, it is normally safe to assume that eeprom changes during normal run mode are errors.

The host can also periodically test the checksum of the eeprom as a backup mechanism to the bit 4 error flag.

The uppercase 'L' command, Lock Reference Levels, also writes data to eeprom, and this data also has the potential to become corrupted. This data is also backed up in Flash so that it can be recovered, and an error in this data will also set bit 4 and also alter the checksum. Also, the 'L' command only operates if the device is in Put mode as a further protection.

Flash rom has a limit of 1,000 write cycles, so copy-to-Flash should not be used often.

5 Commands & Functions

The command structure is designed to minimize control and data traffic. All repetitive data and status commands from the host are single-byte, and most commands result in single-byte device responses. Behavioral setup commands involve multiple bytes but these are infrequently used.

Special 'scope' commands exist to restrict subsequent commands to a specific key or range of keys. This control structure permits most matrix keys, which are usually identical in shape and size, to be programmed 'in bulk' using a 'global' scope command, followed by a scope restriction to specific key(s), followed by more key programming, to prevent the need for tedious key-by-key programming across an entire matrix.

There are four types of commands:

Direction - Determine whether subsequent commands are used to get data from or put data to the part;

Scope - Restrict the range of effect of subsequent commands to a specific set of keys;

Status - Cause the part to respond with key information, such as detections, signals, error codes, and the like;

Setup - Modify functionality such as burst length, threshold levels, drift compensation characteristics, etc.

Supervisory - Special functions such as diagnostics, calibration, etc. which affect the part as a whole.

All command types can be intermixed. Even during normal device operation it is possible to use Setup and Supervisory functions to alter key behavior on the fly. There is no special 'setup mode'.

Get/Put, Scope, and many Supervisory functions are volatile and do not persist after a power down or reset cycle. Some Supervisory commands require that the part be reset in order for the new settings to take effect.

Note that the Setup functions write to eeprom and require extra time for a response back to the host. Also note that as with all eeprom memories there is a recommended lifetime limit to the number of writes; this limit is 100,000 cycles.

Command functions are summarized in Section 5-6

It is highly advised to test the device checksum (command '6') or individual key settings or the general device status ('7') once Setups have been programmed into the part, each time the part is powered up and periodically while running.

The part backs up all eeprom locations into Flash, from which data is restored automatically following a reset if eeprom corruption is detected. The part should also be reset after any Put command(s) in order to force the copy of eeprom data into Flash. See Section 4.6.

5.1 Direction Commands

Setup commands can be used to either send control information to the part for programming into its internal eeprom, or to extract the current setting of this information. The same Setup function can do either. To accomplish this the device relies on direction control via the Get and Put commands. In Get mode, a Setup command will return information. In Put mode, the behavior of the device is altered, and often a second operand byte must be sent.

The powerup or reset default mode is Get. The current Get/Put mode persists until countermanded by a different Get/Put command or until the device is reset or powered off.

It is advisable to use Put mode only when actually writing Setups to the device, which will happen infrequently; the part should normally be left in Get mode. Get mode acts as a lock to prevent accidental changes to the internal eeprom.

Multiple direction commands of the same type (g, g, g, g ...) are harmless and can be used to insure that the part does not accidentally enter Put mode for a prolonged period, for example due to noise glitches on the SPI lines. The 'g' command can be repeated every few seconds.

g 0x67 - GET COMMAND

	Scope	Bytes / Cmd	2nd Byte Range	Returns
Put	n/a	1	n/a	0x67
Get	n/a	n/a	n/a	n/a

Lowercase 'G'. The 'g' command causes the device to treat all subsequent Setup commands as 'Gets'; after, when a Setup command is received from the host the part will respond by sending back the current status of that Setup parameter.

The 'g' command is always single-byte and echoes back itself.

p 0x70 - PUT COMMAND

	Scope	Bytes / Cmd	2nd Byte Range	Returns
Put	n/a	1	n/a	0x70
Get	n/a	n/a	n/a	n/a

Lower case 'p'. The 'p' command causes the device to treat all subsequent Setup commands as 'Puts'; after, when a 2-byte Setup command is received from the host the part will respond by programming in the desired parameter for the key(s) which are affected.

The 'p' command is always single-byte and echoes back itself.

5.2 Scope Commands

The host should always set the scope parameter when initializing the part during normal operation as well as during setup. Scope commands are persistent and apply to all subsequent functions that are affected by scope, until a different scope command is issued. On powerup or after reset the device defaults to scope = 'all keys'.

Many functions only address one key regardless of the current scope; in these cases the key being addressed is always the lowest ranked key in the current scope; for example, the single key X3Y0 when scope is X=3 Y=all.

Key numbering convention: The numbering of keys goes by row then column. For example, the key in row X=3, column Y=1 (X3Y1) is key 11. The formula for conversion of an X-Y location to a key number is:

key number = X_row + (Y_column x 8)

where row and column numbers are per Figure 1-6. Keys are acquired in this same burst sequence, i.e. X0Y0, X1Y0, X2Y0 etc.

s 0x73 - Specific Key Scope

	Scope	*Bytes / Cmd*	*2nd Byte Range*	*Returns*
Put	n/a	2	0x00..0x3F	0x73
Get	n/a	n/a	n/a	n/a

Lowercase 'S'. Targets a specific individual key for all further functions that are affected by scope. The second byte must contain a binary key number from 0..63.

S 0x53 - All Keys Scope

	Scope	*Bytes / Cmd*	*2nd Byte Range*	*Returns*
Put	n/a	1	n/a	0x53
Get	n/a	n/a	n/a	n/a

Uppercase 'S'. Addresses all keys in the matrix for all further functions that can target a group of keys. Functions that can only address a single key when scope is set to 'S' will address key X0Y0 (key = 0).

x 0x78 - Row Keys Scope

	Scope	*Bytes / Cmd*	*2nd Byte Range*	*Returns*
Put	n/a	2	0x00..0x07	0x78
Get	n/a	n/a	n/a	n/a

Lowercase 'X'. Targets keys in a specific row for functions that can address key groups. The second byte must contain a row number from 0..7.

Functions that can only address a single key when scope is set using 'x' will address key XnY0 where n= 2nd byte.

y 0x79 - Column Keys Scope

	Scope	*Bytes / Cmd*	*2nd Byte Range*	*Returns*
Put	n/a	2	0x00..0x07	0x79
Get	n/a	n/a	n/a	n/a

Lowercase 'Y'. Targets keys in a specific column for functions that can address key groups. The second byte is a binary column number from 0..7.

Functions that can only address a single key when scope is set using 'y' will address key X0Yn where n= 2nd byte.

5.3 Status Commands

Status commands cause the sensor to report back information related to keys and their signals.

It is not necessary to set the part to Get mode with these commands, although it is advised to leave the part in Get mode as a normal precaution (see Section 5.1)

0 0x30 - Signal for Single Key

	Scope	*Bytes / Cmd*	*# Bytes Rtnd*	*Returns*
Put	n/a	n/a	n/a	n/a
Get	1	1	1	0x00..0xFF

Numeric '0'. Returns the signal level in 8-bit unsigned binary for one key whose location is determined by scope. Note that the signal level is inverted: decreasing values correspond to more touch due to the physics of key detection described in Section 1.1.

1 0x31 - Delta Signal for Single Key

	Scope	*Bytes / Cmd*	*# Bytes Rtnd*	*Returns*
Put	n/a	n/a	n/a	n/a
Get	1	1	1	0x00..0xFF

Numeric '1'. Returns the value {Reference - Signal} in unsigned 8-bit binary for one key whose location is determined by scope. If Signal > Reference, the result is truncated to zero.

Increasing amounts of this value correspond to increasing amounts of touch as the sign of signal is inverted (see 0x30 above).

2 0x32 - Reference Value

	Scope	*Bytes / Cmd*	*# Bytes Rtnd*	*Returns*
Put	n/a	n/a	n/a	n/a
Get	1	1	1	0x00..0xFF

Numeric '2'. Returns the Reference value in unsigned 8-bit binary for one key whose location is determined by scope.

3 0x33 - R2R Offset

	Scope	*Bytes / Cmd*	*# Bytes Rtnd*	*Returns*
Put	n/a	n/a	n/a	n/a
Get	1	1	1	0x00..0xFF

Section , p. 4

Numeric '3'. Returns the R2R offset value in unsigned 8-bit binary for one key whose location is determined by scope. This function is useful primarily for circuit diagnostics or for an independent determination of proper circuit operation.

4 0x34 - Cz State

	Scope	*Bytes / Cmd*	*# Bytes Rtnd*	*Returns*
Put	n/a	n/a	n/a	n/a
Get	1	1	1	0x00..0x02

Section , p. 4

Numeric '4'. Returns the Cz state for one key whose location is determined by scope. This function is useful primarily for circuit diagnostics or for an independent determination of circuit operation after calibration. A higher value indicates more Cz cancellation is being applied to compensate for Cx; a value of 2 indicates both Cz caps are being switched in.

5 0x35 - Detection Integrator Counts

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	1	1	1	0x00..0xFF

Numeric '5'. Returns the Detection Integrator counter value for one key whose location is determined by scope. This function is useful primarily for circuit diagnostics.

6 0x36 - Eeprom Checksum

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	n/a	1	1	0x00..0xFF

Numeric '6'. Returns the entire eeprom checksum. This function is useful primarily for diagnostics and should periodically be used to check for valid eeprom contents.

The checksum should be computed when the entire device's settings, including the locked reference levels ('L' command) are known. The host can then periodically test the checksum to validate eeprom integrity. If needed, the eeprom can then be reprogrammed by the host or the device can be reset to allow the eeprom to be updated from Flash rom (see Section 4.6).

The checksum is a simple 8-bit carry fold-back type. Changes to multiple Setups can generate identical checksums. Changes to one location only will always produce a different checksum. An identical change to 2, 4, 8, 16, 32 or 64 keys is more prone to generating an identical checksum. A unique checksum can be obtained again by altering any Setup for another key (i.e. an unused key) to be different.

Note that the general status byte returned by the ‘7’ command contains a bit that is set if there is an error in eeprom data; this feature operates independently of the checksum command.

There is no put version of the command.

7 0x37 - General Device Status

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	n/a	1	1	0x00..0x0F

Section 2.12, p. 8

Numeric '7'. Returns the part's general status byte which is a 4-bit pattern as follows:

Bit 0: 1= one or more keys are in detection
Bit 1: 1= one or more keys are recalibrating
Bit 2: 1= one or more keys are reporting errors
Bit 3: 1= sync fail; the part is not synchronized to an external source (if in that mode; see Section 3.17).
Bit 4: 1 = Eeprom / Flash contents discrepancy

Higher bits report as 0's and are not used.

This command can be used as a general 1-byte status response; if one or more bits are set, the host can take further relevant action to narrow down the specific issue, such as which key is being touched or in error, via other commands.

<SP> 0x20 - Signal Levels for Group

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	8, 64	1	8 or 64	0x00..0xFF

Space character. Same function as 0x30 above except returns a group response of 8 bytes (if Scope = row or column selected) or 64 bytes (if Scope = entire matrix selected). If no group scope has been selected, returns data for all keys (64 bytes).

! 0x21 - Delta Signals for Group

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	8, 64	1	8 or 64	0x00..0xFF

Exclamation character. Same function as 0x31 above except returns a group response for 8 or 64 keys depending on current scope. If no group scope has been selected, returns data for all keys (64 bytes).

" 0x22 - Reference Levels for Group

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	8, 64	1	8 or 64	0x00..0xFF

Double quote character. Same function as 0x32 above except returns a group response of 8 or 64 bytes depending on current scope. If no group scope has been selected, returns 64 bytes.

0x23 - R2R Offset for Group

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	8, 64	1	8 or 64	0x00..0xFF

Hash character. Same function as 0x33 above except returns a group response of 8 bytes (Scope = row or column selected) or 64 bytes (Scope = entire matrix selected). If no group scope has been selected, returns 64 bytes.

$ 0x24 - Charge Cancellation for Group

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	8, 64	1	8 or 64	0x00..0x03

Dollar character. Same function as 0x34 above except returns a group response of 8 bytes (Scope = row or column selected) or 64 bytes (Scope = entire matrix selected). If no group scope has been selected, returns 64 bytes.

% 0x25 - Detect Integrator Counts for Group

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	8, 64	1	8 or 64	0x00..0xFF

Percent character. Same function as 0x35 above except returns a group response of 8 bytes (Scope = row or column) or 64 bytes (Scope = entire matrix). If no group scope has been selected, returns 64 bytes.

e 0x65 - Error Code for Selected Key

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	1	1	1	0x00..0x0F

Section 2.12, p. 8

Lowercase 'E'. Returns the error byte for a selected key defined by the 's' command. A 4-bit pattern is returned:

b7	b6	b5	b4	b3	b2	b1	b0
u	u	u	u	L	H	R	F

F: 1= failed last full recalibration attempt
R: 1= key is in process of full recalibration
H: 1= key reference is high (above normal bounds)
L: 1= key reference is low (below normal bounds)
u: undefined

Refer also to Section 2.10.

F, Bit 0 is set if it failed to calibrate properly during a forced recalibration. The sensor will automatically make 5 sequential attempts at recalibration before setting this flag.

R, Bit 1 is set if the key is in the process of a full recalibration. When set, bits 2 and 3 are immediately cleared.

H, Bit 2 when set indicates either:
- the reference has drifted above decimal 191, or,
- the total absolute reference level has become higher than the upper window boundary described in Section 2.11 and as defined by Command ^N after a forced recalibration.

L, Bit 3 when set indicates either:
- the reference has drifted below decimal 64, or,
- the total absolute reference level has become lower than the lower boundary described in Section 2.11, as defined by Command ^O after a forced recalibration.

Bits 2 and 3, if caused by drift compensation, would indicate that the key should be recalibrated by the host. If H and L appear immediately after a full recalibration, and boundary checking is enabled, it means that the key is probably defective.

E 0x45 - Error Codes for Group

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	1, 8, 64	1	1 or 8	0x00..0xFF

Section 2.12, p. 8

Uppercase 'E'. Returns general error codes for a range of keys defined by scope. Returns either 1 or 8 bytes depending on whether a single key, row, column, or entire matrix are selected.

The bitfields for a single key are the same as for 'e' above.

The bitfields for a single row (X) are:

b7	b6	b5	b4	b3	b2	b1	b0
Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0

The bitfields for a single column (Y) are:

b7	b6	b5	b4	b3	b2	b1	b0
X7	X6	X5	X4	X3	X2	X1	X0

The bitfields for a global response are:

	b7	b6	b5	b4	b3	b2	b1	b0
byte1	X7Y0 7	X6Y0 6	X5Y0 5	X4Y0 4	X3Y0 3	X2Y0 2	X1Y0 1	X0Y0 0
byte2	X7Y1 15	X6Y1 14	X5Y1 13	X4Y1 12	X3Y1 11	X2Y1 10	X1Y1 9	X0Y1 8
byte3	X7Y2 23	X6Y2 22	X5Y2 21	X4Y2 20	X3Y2 19	X2Y2 18	X1Y2 17	X0Y2 16
byte4	X7Y3 31	X6Y3 30	X5Y3 29	X4Y3 28	X3Y3 27	X2Y3 26	X1Y3 25	X0Y3 24
byte5	X7Y4 39	X6Y4 38	X5Y4 37	X4Y4 36	X3Y4 35	X2Y4 34	X1Y4 33	X0Y4 32
byte6	X7Y5 47	X6Y5 46	X5Y5 45	X4Y5 44	X3Y5 43	X2Y5 42	X1Y5 41	X0Y5 40
byte7	X7Y6 55	X6Y6 54	X5Y6 53	X4Y6 52	X3Y6 51	X2Y6 50	X1Y6 49	X0Y6 48
byte8	X7Y7 63	X6Y7 62	X5Y7 61	X4Y7 60	X3Y7 59	X2Y7 58	X1Y7 57	X0Y7 56

Byte 1 is the first returned byte in the sequence.

In all the above examples a '1' in a bit position indicates that there is some type of error associated with the key. The use of the 'e' command (or 'E' with scope set to a specific key) will specify the nature of the error.

k 0x6B - Reporting of First Touched Key

	Scope	Bytes / Cmd	#Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	n/a	1	1	0x00..0xFF

Section 2.12, p. 8

Lowercase 'K'. Returns a byte that indicates which if any key has been touched. The byte is structured as follows:

b7	b6	b5	b4	b3	b2	b1	b0
m	-	k5	k4	k3	k2	k1	k0

Bits are used as follows:
m - if '1', indicates that yet another key is active
k0..k5 - indicates the key number of a first detected key, in the range 0..63 (0x00..0x3F).

If a reported key drops out while other keys are active, 'k' will report one of the other active keys, but there is no rule for which of the next keys gets reported in k0..k5.

If the byte returned has a value of 255 (0xFF), then no key has been detected.

K 0x4B - Key Touch Reporting for Group

	Scope	Bytes / Cmd	# Bytes Rtnd	Returns
Put	n/a	n/a	n/a	n/a
Get	1, 8, 64	1	1 or 8	0x00..0xFF

Section 2.12, p. 8

Uppercase 'K'. Returns 1 or 8 bytes depending on the current scope. The byte(s) returned contain a bit pattern which indicates touched keys. A scope of a single key, row or column will return one byte. A scope of all keys will return 8 bytes. If scope is one key, only the LSB is used to report.

The bitfields for a single key are:

b7	b6	b5	b4	b3	b2	b1	b0
-	-	-	-	-	-	-	key

The bitfields for a single row (scope is X) are:

b7	b6	b5	b4	b3	b2	b1	b0
Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0

The bitfields for a single column (scope is Y) are:

b7	b6	b5	b4	b3	b2	b1	b0
X7	X6	X5	X4	X3	X2	X1	X0

The bitfields for a global report are:

	b7	b6	b5	b4	b3	b2	b1	b0
byte1	X7Y0 7	X6Y0 6	X5Y0 5	X4Y0 4	X3Y0 3	X2Y0 2	X1Y0 1	X0Y0 0
byte2	X7Y1 15	X6Y1 14	X5Y1 13	X4Y1 12	X3Y1 11	X2Y1 10	X1Y1 9	X0Y1 8
byte3	X7Y2 23	X6Y2 22	X5Y2 21	X4Y2 20	X3Y2 19	X2Y2 18	X1Y2 17	X0Y2 16
byte4	X7Y3 31	X6Y3 30	X5Y3 29	X4Y3 28	X3Y3 27	X2Y3 26	X1Y3 25	X0Y3 24
byte5	X7Y4 39	X6Y4 38	X5Y4 37	X4Y4 36	X3Y4 35	X2Y4 34	X1Y4 33	X0Y4 32
byte6	X7Y5 47	X6Y5 46	X5Y5 45	X4Y5 44	X3Y5 43	X2Y5 42	X1Y5 41	X0Y5 40
byte7	X7Y6 55	X6Y6 54	X5Y6 53	X4Y6 52	X3Y6 51	X2Y6 50	X1Y6 49	X0Y6 48
byte8	X7Y7 63	X6Y7 62	X5Y7 61	X4Y7 60	X3Y7 59	X2Y7 58	X1Y7 57	X0Y7 56

Byte 1 is the first returned byte in the sequence.

In all the above examples a '1' in a bit position indicates that the key is touched; a '0' indicates no touch.

5.4 Setup Commands

Setup functions are those that alter the behavior a key or a group of keys. The setups are programmed into eeprom locations in the part and ordinarily do not need to be reprogrammed once set. However it is possible to change a setup while the device is in normal operation without interrupting the sensing function of the part.

Setup functions alter the internal eeprom, and this requires a much longer time to complete than other commands; see Table 4-1.

Setup 'put' commands become effective immediately after the echo response of the command byte unless otherwise noted; some setups require that the key(s) being altered be recalibrated with the 'b' command before they take effect.

^A 0x01 - Negative Detect Threshold

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	1, 8, 64	2	0x04..0x40	0x01
Get	1	1	n/a	0x04..0x40

Section 2.1, p. 5

Ctrl-A. In Put mode, the command followed by a setting is programmed into eeprom for the key(s) affected by scope.

1, 8, or 64 keys may be affected. Valid decimal values are:

4	5	6	7	8	10	12	15
17	20	25	30	35	45	55	64

Values other than the above will be ignored.

In Get mode, the command will return a single byte. If scope is a row or column, ^a will return the lowest numbered key, i.e. X0 or Y0. If scope is the entire matrix, X0Y0 is returned.

This setup controls key sensitivity by setting the counts of signal delta needed to cause a detect. Higher = less sensitive. Numbers should be 6 or greater under most conditions to reduce the probability of noise detection. Numbers greater than 20 indicate that the burst length is probably too high. This setup interacts with Burst Length (^F).

^B 0x02 - Positive Detect Threshold

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	1, 8, 64	2	0x04..0x40	0x02
Get	1	1	n/a	0x04..0x40

Section 2.2, p. 6

Ctrl-B. In Put mode, the command followed by a setting is programmed into eeprom for the key(s) affected by scope. 1, 8, or 64 keys may be affected. Valid decimal values are:

4	5	6	7	8	10	12	15
17	20	25	30	35	45	55	64

Values other than the above are ignored.

In Get mode, the command will return a single byte. If scope is a row or column, ^B will return the lowest numbered key, i.e. X0 or Y0. If scope is the entire matrix, X0Y0 is returned.

This setup controls the ability of a key to recalibrate quickly should the signal transition positive quickly, as when a touch is prolonged enough to cause a recalibration, and when the key is then 'untouched'. This condition can also be caused by a foreign object being removed from a key. The value should normally be set between 6 and 10 counts. If the value is very high, the key will still recover by means of the drift compensation process, albeit more slowly.

^C 0x03 - Negative Threshold Hysteresis

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	64	2	0x01..0x03	0x03
Get	64	1	n/a	0x01..0x03

Section 2.3, p. 6

Ctrl-C. In Put mode, the command followed by a setting is programmed into eeprom for all keys only. The value should be from 0 to 3, representing hysteresis as follows:

0: 50%
1: 25%
2: 12.5%
3: 0% (no hysteresis)

Values other than the above will be ignored.

The percentage is the distance from the threshold level to the reference level. The hysteresis level is always closer to the threshold point than to the reference point. 25% is a reasonable value under most conditions.

As this parameter is common to all keys, Put and Get operations send or return only one byte.

^D 0x04 - Positive Threshold Hysteresis

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	64	2	0x01..0x03	0x04
Get	64	1	n/a	0x01..0x03

Section 2.3, p. 6

Ctrl-D. Identical in operation to ^C above except this applies to positive 'detections' used to recalibrate the sensor (see ^B above for details). Uses same hysteresis values as ^C above.

^E 0x05 - Dwell Time in Machine Cycles

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	64	2	0x01..0x0a	0x05
Get	64	1	n/a	0x01..0x0a

Section , p. 4; Section 3.3, p. 9; Section 3.13, p. 12

Ctrl-E. Governs the delay from the rise of an X drive to the termination of Y transfer gating ('dwell time'). See also Section 3.3.

One machine cycle is one oscillator clock cycle, i.e. 167ns with a 6MHz resonator. This feature allows for test and experimentation regarding water film suppression on the user surface of the matrix. Normally it should be set to 1 (one) cycle. There are no particular advantages to using longer X-Y delays, as the actual dwell time in the reference circuit is normally governed by the PLD. Longer dwell times can unintentionally increase the intra-burst pulse spacing.

As this parameter is common to all keys, Put and Get operations send or return only one byte.

^F 0x06 - Burst Length

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	1, 8, 64	2	0x00..0x40	0x06
Get	1	1	n/a	0x00..0x40

Section 3.6, p. 11

Ctrl-F. In Put mode the command sets the burst length of one or more keys, according to the current scope. Valid decimal values are:

0	1	2	3	4	5	7	10
12	15	20	25	30	40	50	64

Values other than the above will be ignored.

In Get mode the value of only a single byte is returned. If scope is a row or column, ^F will return the lowest numbered key, i.e. X0 or Y0. If scope is the entire matrix, the Burst Length for X0Y0 is returned.

^F sets the length of the acquisition burst on a key by key basis. This setting is directly proportional to signal gain. This setup interacts with Negative and Positive Threshold (^A and ^B). Increasing ^F can allow for higher threshold levels and more robust signals, at the expense of increased radiated emissions and reduced Cx load capacity.

Special condition: If the value for ^F for a key is set to zero the burst disabled and the key will not function; the key will report back with an error code. The timing for the 'phantom burst' will be preserved so that overall key scan timing will remain unchanged.

^G 0x07 - Burst Spacing

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	64	2	0x00..0x05	0x07
Get	64	1	n/a	0x00..0x05

Section 3.8, p. 11

Ctrl-G. In Put mode, sets the spacing between successive acquire bursts for the entire matrix.

The second byte indicates the spacing to be set according to the following values:

0: 250μs
1: 300μs
2: 400μs
3: 500μs
4: 1000μs
5: 2000μs

Values higher than the above will be truncated to 2000μs.

Longer delay times equate to slower acquisitions. At lower delay times (faster rep rates) there can be conflicts with long burst lengths which will prevent proper operation; see Section 5.7.

The time required to scan the entire keymatrix once is the above delay multiplied by 64 regardless of the number of keys actually used or the part model number.

Burst spacing also affects recalibration time; see Section 2.10

The scope for this function is always 'all keys'.

^H 0x08 - Negative Drift Compensation Rate

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	1, 8, 64	2	0x01..0x64	0x08
Get	1	1	n/a	0x01..0x64

Section 2.4, p. 6

Ctrl-H. In Put mode, sets the rate of drift compensation used during periods of non-detection, in the negative signal direction.

The second byte must be one of the following valid values (shown in decimal):

1	2	3	4	6	8	10	12
15	20	25	33	45	60	75	100

Values other than the above will be ignored.

These numbers correspond to the amount of drift compensation applied, in 100ms/count of reference change, for signals which are negative with respect to the reference level, i.e. in the same direction as legitimate detections. Higher numbers equate to slower drift compensation. Overcompensation (too fast) can result in the suppression of legitimate detections. Under-compensation can result in inadequate compensation for rapid environmental changes. Values of 15 to 45 (1.5 to 4.5 secs/count) are considered normal under most conditions.

Drift compensation does not occur while the signal has passed below the ^A threshold level or subsequently remained below the negative hysteresis level.

In Get mode the function returns a single value only; if scope is set to row, column, or all, only the value for the lowest ranking key in the group will be returned.

The scope in Put mode can be one key, a row or column, or all keys.

^I 0x09 - Positive Drift Compensation Rate

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	1, 8, 64	2	0x01..0x64	0x09
Get	1	1	n/a	0x01..0x64

Section 2.4, p. 6

Ctrl-I. Same as ^H above in all respects, except operates only when the signal is positive with respect to the reference level, i.e. in an abnormal direction. It is usually desirable to set this rate much faster than for ^H, i.e. to a lower number. Valid decimal values are:

1	2	3	4	6	8	10	12
15	20	25	33	45	60	75	100

Values other than the above will be ignored.

Values of 4 to 10 (0.4 to 1.0 secs/count) are considered suitable for most systems.

Positive drift compensation does not occur while the signal has exceeded the positive threshold level (see Setup ^B) or subsequently remained above the positive hysteresis level.

^J 0x0A - Detect Integrator Limit

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	1, 8, 64	2	0x00..0xFF	0x0A
Get	1	1	n/a	0x00..0xFF

Section 2.6, p. 7

Ctrl-J. In Put mode, sets the detect integrator limit for one or more keys according to scope.

The unit of measure is a burst, i.e. a setting of 5 means that a detection must be sensed 5 bursts in sequence. A burst for a key occurs once every complete matrix scan. Thus, if the burst spacing is 500us, the response time will be:

5 x 500us x 64 = 160ms

The second byte must be one of the following values (shown in decimal):

0	1	2	3	5	7	10	15
20	32	45	60	90	123	175	255

Values other than the above will be ignored.

In Get mode the function returns a single value only; if scope is set to row, column, or all, only the value for the lowest ranking key in the group will be returned.

This setup can be used as a noise filter, or as a mechanism to intentionally slow down key reaction time in order to require a long user touch.

Special condition: If the value for ^J is set to zero the key is disabled, but the burst for the key is still generated.

^K 0x0B - Positive Recalibration Delay

	Scope	Bytes / Cmd	Byte 2 Range	Returns
Put	1, 8, 64	2	0x00..0xFF	0x0B
Get	1	1	n/a	0x00..0xFF

Section 2.7, p. 7

Ctrl-K. In Put mode, sets the delay until recalibration, timed from when the signal first crosses the positive threshold.

The second byte controls the delay in 100ms increments, and must be one of the following valid values:

0	1	2	3	5	7	10	15
20	32	45	60	90	123	175	255

Values other than the above will be ignored. As an example, a value of 60 will cause a 6-second delay.

In Get mode the function returns a single value only; if scope is set to row, column, or all, only the value for the lowest ranking key in the group will be returned.

Special condition: If ^K is set to zero this feature is disabled and the key will never auto-recalibrate on positive transitions; however drift compensation will still operate.