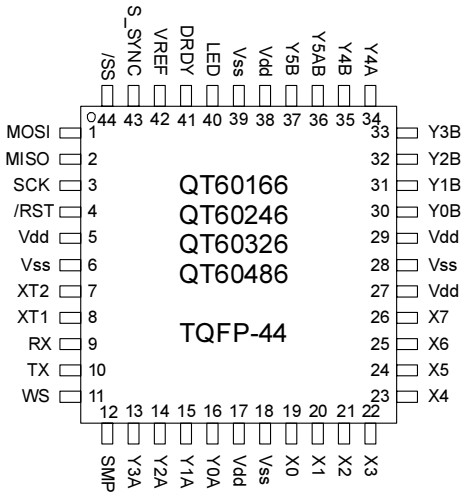
QUANTUM RESEARCH GROUP

ADVANCE INFORMATION

QProx™ QT60xx6
16, 24, 32, 48 KEY QMATRIX™ ICs

- **Advanced second generation QMatrix controller**
- **Keys individually adjustable for sensitivity, response time, and many other critical parameters**
- **Panel thicknesses to 50mm through any dielectric**
- **16, 24, 32 or 48 touch key versions**
- **100% autocal for life - no adjustments required**
- **SPI Slave or Master/Slave interface to a host controller**
- **UART serial interface to a host controller**
- **Sleep mode with wake pin**
- **Adjacent key suppression feature**
- **Synchronous noise suppression pin**
- **Spread-spectrum modulation: high noise immunity**
- **Mix and match key sizes & shapes in one panel**
- **Low overhead communications protocol**
- **FMEA compliant design features**
- **Negligible external component count**
- **Extremely low cost per key**
- **44-pin TQFP package**



APPLICATIONS -

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

These digital charge-transfer ("QT") QMatrix™ ICs are designed to detect human touch on up 48 keys when used with a scanned, passive X-Y matrix. They will project touch 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 or UART port, for example via Quantum's QmBtn program, or from a host microcontroller. Key setups are stored in an onboard eeprom and do not need to be reloaded with each powerup.

These devices 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. It can 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 device contains Quantum-pioneered adaptive auto self-calibration, drift compensation, and digital filtering algorithms that make the sensing function robust and survivable.

The parts can scan matrix touch keys over LCD panels or other displays when used with clear ITO electrodes arranged in a matrix. They do 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 a resonator and a few passive parts, all of which can fit into a 6.5 sq cm footprint (1 sq inch). Control and data transfer is via either a SPI or UART port, which is autodetected.

These devices makes 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 older methods, it does not require one IC per key.

AVAILABLE OPTIONS

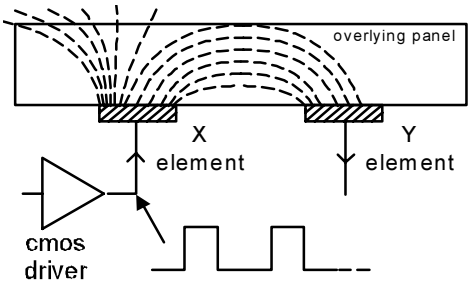
T_A	# Keys	Part Number
-40°C to +105°C	16	QT60166-AS
-40°C to +105°C	24	QT60246-AS
-40°C to +105°C	32	QT60326-AS
-40°C to +105°C	48	QT60486-AS

Copyright © 2003 QRG Ltd

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 5 square centimeters of single-sided PCB area.

Figure 1-1 Field flow between X and Y elements



QMatrix parts employ transverse charge-transfer ('QT') sensing, a technology that senses changes in electrical charge forced across an electrode by a digital edge (Figure 1-1).

QMatrix devices allow for a wide range of key sizes and shapes to be mixed together in a single touch panel.

The devices use both UART and SPI interfaces 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 operation of the IC as well as read back key status and signal levels in real time.

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

1.1 Part differences

Versions of the device are capable of a maximum of 16, 24, 32, and 48 keys.

The QT60xx6 devices are identical to one another in all respects, except that each device is capable of only the number of keys specified for each device. These keys can be located anywhere within the electrical grid of 8 X and 6 Y scan lines. Unused keys are always pared from the burst sequence in order to optimize timing performance.

Even with a given part type, such as QT60486, a lesser number of enabled keys will cause any unused acquisition burst timeslots to be pared. Thus, if only 40 keys are actually enabled, only 40 timeslots are used for scanning.

2 Hardware

2.1 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 up to 64 times in a burst whose length is determined by the Setups parameter BL, which can be set on a per-key basis. A burst is completed entirely before the next key is sampled; at the end of each burst the resulting signal is converted to digital form and processed. The burst length directly impacts key gain; each key can have a unique burst length in order to allow tailoring of key sensitivity on a key by key basis.

2.2 Oscillator

The oscillator can use either a quartz crystal or a ceramic resonator. In either case, the XT1 and XT2 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 XT1 in which case, XT2 should be left unconnected.

The frequency of oscillation should be 16MHz +/-1% for accurate UART transmission timing.

2.3 Sample Capacitors

The charge sampler capacitors on the Y pins should be the values shown. They can be X7R ceramic type. The value of these capacitors is non-critical and can vary from 3.3nF to 10nF; 4.7nF is acceptable in most cases. Heavy Cx load capacitances may necessitate the use of larger Cs capacitors.

The Cs capacitor values have no effect on conversion gain.

Unused Y lines should have a 1nF dummy capacitor connected as shown.

2.4 Sample Resistors

There are 6 sample resistors (Rs) used to perform single-slope ADC conversion of the acquired charge on each Cs capacitor. These resistors are directly linked with acquisition gain. Larger values of Rs will proportionately increase signal gain. Values of Rs can range from 220KΩ to 1MΩ. 220KΩ is a reasonable typical value for most purposes.

Larger values for Rs will also increase conversion time and may reduce the fastest possible key sampling rate, which can impact response time especially with larger numbers of enabled keys.

2.5 Signal Levels

Using Quantum's QmBtn™ software it is easy to observe the absolute level of signal received by the sensor on each key. The signal values should normally be in the range from 250 to 750 counts with properly designed key shapes (see appropriate Quantum app note on matrix key design).

QmBtn software is available free of charge on Quantum's website.

The signal swing from the smallest finger touch should preferably exceed 10 counts, with 15 being a reasonable target. The signal threshold setting (NTHR) should be set to a value guaranteed to be less than the signal swing caused by the smallest touch.

Increasing the burst length (BL) parameter will increase the signal strengths as will increasing the Rs values.

2.6 Matrix Series Resistors

The X and Y matrix scan lines should use series 1K resistors or higher. X drive lines require them in most cases to reduce edge rates and thus RF emissions. Y lines need them to reduce EMC susceptibility problems and in some cases, ESD effects.

1K is a good starting point, but in fact the value can be much higher in most cases. The end limit is reached when the signal level and hence key sensitivity is clearly being affected by the resistance. Too high a value on the X lines will limit the charge coupling across the key. Too high a value on the Y lines will reduce the amount of charge captured by the sampling capacitor.

End limits can vary depending on key geometry and stray capacitance, but often are found to be in the region of 20K ~ 50K ohms.

2.7 Key Design & Layouts

Keys can be constructed out of a variety of materials including flex circuits, FR4, and even inexpensive single-sided CEM-1. It is best to place the chip near the keys on the same PCB so as to reduce trace lengths, thereby reducing the chances for EMC problems.

Please refer to the latest Quantum application note on how to create PCB layouts for keys.

2.8 Startup / Calibration Times

The devices require 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 cal 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.

2.9 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 Force Reset command, 0x04 is also provided which generates an equivalent hardware reset.

If an external reset is not used, this pin may be connected to Vdd.

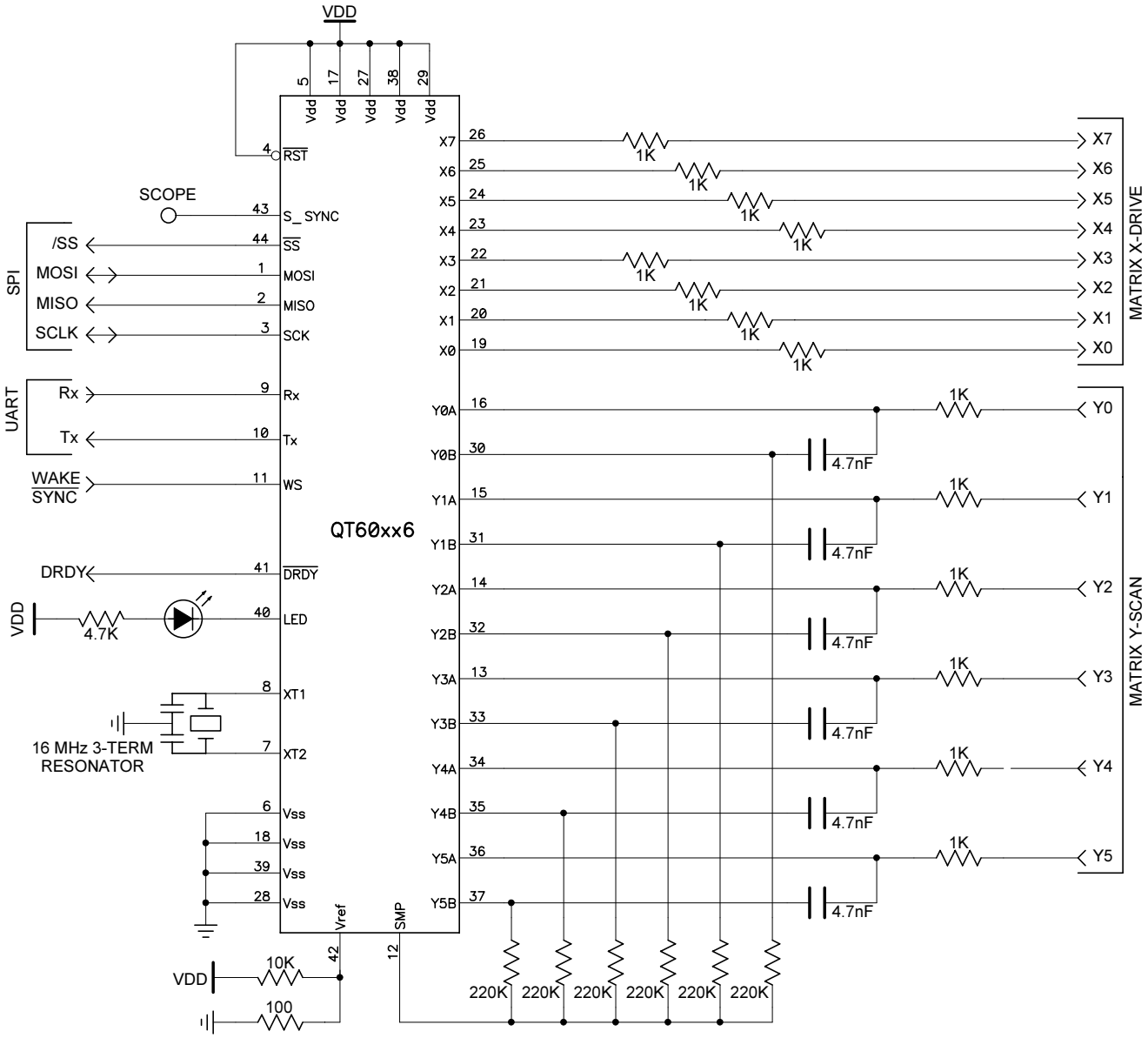
2.1 Wiring

Table 2.1 - Pin Listing

Applies to all devices

Pin	Function	I/O	Comments	If Unused, Connect To..
1	MOSI	I/O	SPI data input	Vss or Vdd
2	MISO	O	SPI data output	Leave open
3	SCK	I/O	SPI clock input	Vss or Vdd
4	/RST	I	Reset low	Vdd
5	Vdd	P	Power, +5V	-
6	Vss	P	Supply ground	-
7	XT2	O	16 MHz 3-terminal resonator	Leave open
8	XT1	I		-
9	Rx	I	UART receive data input	Vdd
10	Tx	O	UART transmit data; use 10K ~ 50K pullup	Leave open
11	WS	I	Wake-up from sleep input / sync input	Vss or Vdd
12	SMP	I/O	Sample output. Also - When forced high before reset, induces 'factory defaults' into all setups.	-
13	Y3A	I	Y line connection	Use 1nF dummy Cs
14	Y2A	I	Y line connection	Use 1nF dummy Cs
15	Y1A	I	Y line connection	Use 1nF dummy Cs
16	Y0A	I	Y line connection	Use 1nF dummy Cs
17	Vdd	P	Power, +5V	-
18	Vss	P	Supply ground	-
19	X0	O	X matrix drive line	Leave open
20	X1	O	X matrix drive line	Leave open
21	X2	O	X matrix drive line	Leave open
22	X3	O	X matrix drive line	Leave open
23	X4	O	X matrix drive line	Leave open
24	X5	O	X matrix drive line	Leave open
25	X6	O	X matrix drive line	Leave open
26	X7	O	X matrix drive line	Leave open
27	Vdd	P	Power, +5V	-
28	Vss	P	Supply ground	-
29	Vdd	P	Power, +5V	-
30	Y0B	I	Y line connection	Use 1nF dummy Cs
31	Y1B	I	Y line connection	Use 1nF dummy Cs
32	Y2B	I	Y line connection	Use 1nF dummy Cs
33	Y3B	I	Y line connection	Use 1nF dummy Cs
34	Y4A	I	Y line connection	Use 1nF dummy Cs
35	Y4B	I	Y line connection	Use 1nF dummy Cs
36	Y5A	I	Y line connection	Use 1nF dummy Cs
37	Y5B	I	Y line connection	Use 1nF dummy Cs
38	Vdd	P	Power, +5V	-
39	Vss	P	Supply ground	-
40	LED	O	Status output / LED indicator drive	Leave open
41	DRDY	O	1= Comms ready; use 10K ~ 50K pullup	Leave open
42	Vref	I	0.05V nominal +/-10% via external divider	-
43	S_Sync	O	Scope Sync: Synchronization test signal	Leave open
44	/SS	I	SPI slave select	Vdd

Figure 2.1 Wiring Diagram



Note: Use either UART or SPI comm port but not both. Device autodetects communication type depending on which one first receives a command. See Section Table 2.1 for connections when pins are unused.

3 Serial Communications

These devices can use either SPI or UART communications modes; it cannot use both at the same time. The mode selected depends on which mode is used first to communicate with the part.

The host device always initiates communications sequences; the QT is incapable of chattering data back to the host. This is intentional for FMEA purposes so that the host always has total control over the communications with the QT60xx6.

A command from the host always ends in a response of some kind from the QT. Some transmission types from the host or the QT employ a CRC check byte to provide for robust communications.

A DRDY line is provided that handshakes transmissions. Generally this is needed by the host from the QT to ensure that transmissions are not sent when the QT is busy or has not yet processed a prior command. In UART mode this line is bi-directional, and the QT can use it to suspend transmissions back to the host if the host is busy.

3.1 DRDY Line

Serial communications is controlled by the DRDY line, which is an output from the QT60xx6 to the host. When DRDY is high, the host is permitted to send data. This works in both UART and SPI modes. After a byte is received DRDY will always go low even if only for a few microseconds; during this period the host should not send data. Therefore, after each byte transmission the host should first check that DRDY is high again.

The host should sequence transmissions as follows:

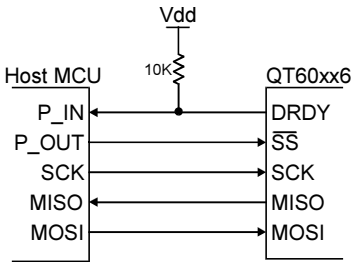
1. Check to see if DRDY is high; if not, wait
2. If DRDY is high: send a byte to QT
3. Wait 100µs or longer (time T2)
4. Wait until DRDY is high (it may already be high)
5. Send next command or null byte to QT

DRDY is an open-drain output which must be pulled high by an external resistor, from 10K ~ 50K ohms in either UART or SPI mode.

3.2 SPI Communications

SPI mode is selected if the host sends data over the SPI lines first. There is no other configuration required to make the device operate in SPI mode. Once SPI is selected after a powerup, the device cannot switch to UART mode unless the device is reset.

Figure 3-1 SPI Connections



SPI communications operates in slave mode only, and obeys DRDY control signaling. The clocking is as follows:

Clock idle:	High
Clock shift out edge:	Falling
Clock data in edge:	Rising
Max clock rate:	4MHz

SPI mode requires 5 signals to operate:

MOSI - Master out / Slave in data pin; used as an input for data from the host (master). This pin should be connected to the MOSI (DO) pin of the host device.

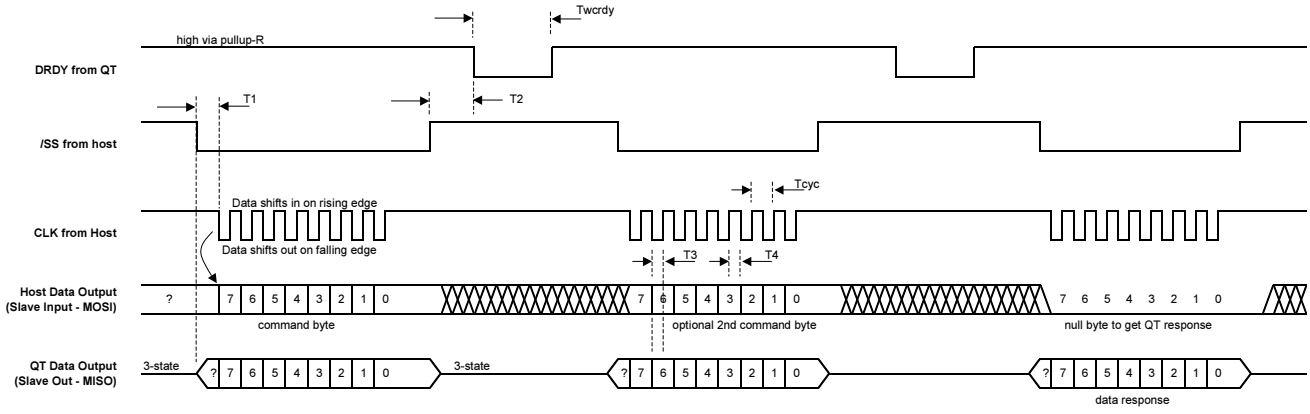
MISO - Master in / Slave out data pin; used as an output for data to the host. This pin should be connected to the MISO (DI) pin of the host.

SCK - SPI clock - input only clock pin from host. The host must shift out data on the falling edge of SCK; the QT60xx6 clocks data in on the rising edge of SCK. The QT60xx6 likewise shifts data out on the rising edge back to the host. **Important note:** SCK must idle high; SCK should never float.

/SS - Slave select - input only; acts as a framing signal to the sensor from the host. /SS must be low before and during reception of data from the host. It must not go high again until the SCK line has returned high; /SS must idle high.

DRDY - Data Ready - active-high - indicates to the host that the QT is ready to send or receive data. This pin idles high. DRDY should be pulled high with a 10K to 100K pullup resistor. In SPI mode this pin is an output only.

Figure 3-2 SPI Slave-Only Mode Timing



The MISO pin on the QT floats in 3-state mode between bytes when /SS is high. This facilitates multiple devices on one SPI bus.

Null Bytes: When the QT responds to a command with one or more response bytes, the host can issue a new command to the QT instead of a null in the last shift operation.

New commands attempted during intermediate byte transfers are ignored, and null bytes should always be used in these cases.

3.3 UART Communications

See also SR setup parameter, page 16.

UART mode is selected if the host sends data over the UART lines first. There is no other configuration required to make the device operate in UART mode. Once UART is selected after a power-up, the device cannot switch to SPI mode unless the device is reset.

UART mode communications functions in the same basic way as SPI communications. The Baud rate is adjusted by means of setup parameter 'SR' (pages 16, 0). Once a new Baud rate has been set, the device must be reset for the new rate to take effect.

The major difference with SPI mode is that the UART mode is asynchronous and so the host does not clock the QT. No framing /SS or clock signal is required, simplifying the interface greatly. Return data is sent from the QT back to the host when the data is ready.

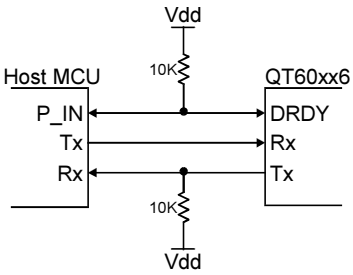
Multi-drop capability: The QT60xx6 in UART mode floats Tx within 10µs after each transmitted byte. The host's Rx pin can thus be shared with other similar UART based peripherals.

Wake operation: The device can be put into sleep mode with a serial command, 0x16 (page 9) and then be waked with a dummy null byte from the host, if the Rx and WS pins are connected together.

Rx - Receive async data. This pin is an input only.

Tx - Transmit async data. Drives out when transmitting but floats within 10µs of the end of the stop bit, to allow bussing with several similar parts. Tx should idle high, and must be pulled high with a 10K ~ 20K resistor to Vdd at all times in UART mode. Tx is push-pull when transmitting data.

Figure 3-3 UART Connections



UART transmission parameters are:

Baud rate:	9600 ~ 115,200
Start bits:	1
Data bits:	8
Parity:	None
Stop bits:	1

DRDY in UART mode: Section 3.1 applies.

DRDY is bi-directional in UART mode. DRDY can be pulled down by either the QT or the host (wire-AND), so that either device can be inhibited from sending data until the other is ready. The host should obey this control line or transmission errors can occur. The host should grant a 10µs grace period after clamping DRDY low in which it can still accept a transmission.

As explained in Section 3.1, DRDY is not clamped low immediately after the QT receives a byte; there can be up to a 100µs delay from the end of the stop bit before DRDY goes low. Sampling of DRDY by the host should occur 100µs after the byte has been fully sent; if DRDY is already high at this point, or becomes high, then it is clear to send.

Null Bytes: Unlike SPI mode, there is no reason to send null bytes to the QT in UART mode.

4 Control Commands

Refer to Section 5.1, page 11 for further details.

The devices feature a set of commands which are used for control and status reporting. The host device has to send the command to the QT60xx6 and await a response.

SPI mode: While waiting the host should delay for 100 µs from the end of the command, then start to check if DRDY is or goes high. If it is high, then the host master can clock out the resulting byte(s).

UART mode: After the command is sent, the QT will send back the response usually starting within 100µs. The host can clamp DRDY low (wire-AND logic) to inhibit a response if the host is not able to receive the transmission.

4.1 Null Command - 0x00

Used primarily to shift back data from the QT in SPI mode. Since the host device is always the master in SPI mode, and data is clocked in both directions, the Null command is required frequently to act as a placeholder where the desire is to only get data back from the QT, not to send it.

In SPI communications, when the QT60xx6 responds to a command with one or more response bytes, the host can issue a new command instead of a null on the last byte shift operation.

New commands during intermediate byte shift-out operations are ignored, and null bytes should always be used.

4.2 Enter Setups Mode - 0x01

This command is used to initiate the Setups block transfer from Host to QT.

The command must be repeated 2x within 100ms or the command will fail; the repeating command must be sequential without any intervening command. After the 2nd 0x01 from the host, the QT will reply with the character 0xFE. In SPI mode this character must be shifted out by sending a null (0x00) from the host. This command suspends normal sensing starting from the first 0x01. A failure of the command will cause a timeout.

Each byte in the block must arrive at the QT no later than 100ms after the previous one or a timeout will occur.

Any timeout will cause the device to cancel the block load and go back to normal operation.

If no response comes back, the command was not received and the device should preferably be reset from the host by hardware reset just in case there are any other problems.

If 0xFE is received by the host, then the host should begin to transmit the block of Setups to the QT. DRDY handshakes the data. The delay between bytes can be as short as 10us but the host can make it longer than this if required, but no more than 100ms. The last two bytes the host should send is the CRC for the block of data only.

After the block transfer the QT will check the CRC and respond with 0x00 if there was an error. Regardless, it will program the internal eeprom. If the CRC was correct it will reply with a second 0xFE after the eeprom was programmed.

If there was an error in the block transfer the device will restore the last known good Setups from Flash memory the next time the device is reset. However until that point, the device will attempt to operate using the new Setups block even if it is corrupt.

At the end of the full block load sequence, the device restarts sensing without recalibration.

4.3 Cal All - 0x03

This command must be repeated 2x within 100ms or the command will fail; the repeating command must be sequential without any intervening command.

After the 2nd 0x03 from the host, the QT will reply with the character 0xFC. Shortly thereafter the device will recalibrate all keys and restart operation.

If no 0xFC comes back, the command was not properly received and the device should preferably be reset.

The host can monitor the progress of the recalibration by checking the status byte, using command 0x05.

4.4 Force Reset - 0x04

The command must be repeated 2x within 100ms or the command will fail; the repeating command must be sequential without any intervening command. After the 2nd 0x04, the QT will reply with the character 0xFB just prior to executing the reset operation.

The host can monitor the progress of the reset by checking the status byte for recalibration, using command 0x05.

4.5 Error Status - 0x05

This command returns the general error status code.

Bit 5: Set if there is a FMEA failure detected

Bit 6: Set of there is a communications failure. This can be reset by sending command 0x0f (last command command).

A CRC byte is appended to the response; this CRC folds in the command 0x05 itself initially.

4.6 Report 1st Key - 0x06

Reports the first or only key to be touched, plus indicates if there are yet other keys that are also touched.

The return bits are as follows:

BIT	Description
7	1= more than 1 key is active
6	1= any error condition is present
5	Key bit 5
4	Key bit 4
3	Key bit 3
2	Key bit 2
1	Key bit 1
0	Key bit 0

Bits 0..5 encode for the first detected key in range 0..47.

If 2 or more keys in detection, bit 7 is set and the host should interrogate the part via the 0x07 command to read out all the key detections. This one command should be the dominant interrogation command in the host interface; further commands can be issued if the response to 0x06 warrants it.

A CRC byte is appended to the response; this CRC folds in the command 0x06 itself initially.

4.7 Report Detections for All Keys - 0x07
Returns six bytes which indicate all keys in detection if any, as a bitfield. The first byte returned is the MSByte. Key 0 reports in LSByte bit 0.

A CRC byte is appended to the response; this CRC folds in the command 0x07 itself initially.

4.8 Report Signals for All Keys - 0x08
Returns the raw signal values for all keys. Each value is a 16-bit number, and there are 48 words returned. No CRC is appended to the return, so the data should not be considered secure. The high byte of key 0 is returned first.

4.9 Report References for All Keys - 0x09
Returns the reference values for all keys. Each value is a 16-bit number, and there are 48 words returned. No CRC is appended to the return, so the data should not be considered secure. The high byte of key 0 is returned first.

4.10 Report Deltas for All Keys - 0x0a
Returns the delta signal values with respect to the reference levels for all keys. Each value is an 8-bit signed number, and there are 48 bytes returned. No CRC is appended to the return, so the data should not be considered secure. The byte for key 0 is returned first.

If the delta value exceeds the range -127 ... +128, the result is truncated.

4.11 Report Error Flags for All Keys - 0x0b
Returns six bytes which show error flags as a bitfield for all keys. The first byte returned is the MSByte. Key 0 reports in LSByte bit 0.

A CRC byte is appended to the response; this CRC folds in the command 0x0b itself initially.

4.12 Report FMEA Status - 0x0c
Returns one byte which shows the FMEA error status of the X and Y matrix scan lines, OR'd together in one result byte. Each bit in the byte represents the OR of one X and one Y scan line (except for the top two bits which are X only). A one in any bit position indicates an error in a corresponding scan line.

A CRC byte is appended to the response; this CRC folds in the command 0x0c itself initially.

4.13 Dump Setups Block - 0x0d
This command causes the device to dump the entire internal Setups block back to the host.

If the transfer is not paced faster than 100ms per byte the transfer will be aborted and the device will time out. This can happen if the host is also controlling DRDY.

During the transfer, sensing is halted. Sensing is resumed after the command has finished.

A 16-bit CRC is appended to the response; this CRC is the same as the Setups table CRC and is sent LSByte first.

4.14 Eeprom CRC - 0x0e
This command returns the 16-bit CRC calculated from the eeprom contents. The CRC is sent back LSByte first. The CRC sent back is the same CRC that is appended to the end of the Setups block.

No CRC is appended to the response.

4.15 Return Last Command - 0x0f
This command returns the last received command character, in 1's complement (inverted). If the command is repeated twice or more, it will return the inversion of 0x0f, 0xf0.

If a prior command was not valid or was corrupted, it will return the bad command as well.

No CRC is appended to the response.

4.16 Version - 0x10
This command returns the version number of the part as a value from 0..255.

A CRC byte is appended to the response; this CRC folds in the command 0x10 itself initially.

4.17 Internal Code - 0x11
This command returns an internal code word (2 bytes) of the part for factory diagnostic purposes.

A CRC byte is appended to the response; this CRC folds in the command 0x11 itself initially.

4.18 Internal Code - 0x12
This command returns an internal code word (2 bytes) of the part for factory diagnostic purposes.

No CRC is appended to the response.

4.19 Sleep - 0x16
The command must be repeated 2x within 100ms or the command will fail. After the 2nd 0x16 from the host, the device will reply with the character 0xe9 then sleep.

The device will then enter a lower power sleep mode until awakened by an edge or pulse on pin WS. When the device wakes, it will resume current operation in the state from which it exited and attempt to send a 0x01 code back to the host.

During Sleep the DRDY pin is held low, and released once the device awakes and is ready to return the 0x01 code.

The WS pin can be connected to Rx or /SS to provide a 'free' wakeup connection from the host controller. A dummy byte or /SS toggle can be sent to wake up the device.

4.20 Data Set for One Key - 0x4k
Returns the data set for key k, where k = {0..47}. This returns 5 bytes, in the sequence:

Signal (2 bytes)
Reference (2 bytes)
Normal Detect Integrator (1 byte)

Signal and Reference are returned LSByte first.

No CRC is appended.

4.21 Status for Key 'k' - 0x8k
Returns a bitfield for key 'k' where k is from {0..47}. The bitfield indicates as follows:

Bit 4: Set if key is enabled
Bit 3: Set if key is in detect
Bit 2: Set if the key's reference is less than LSL

A CRC byte is appended to the response; this CRC folds in the command 0x8k itself initially.

4.22 Cal Key ‘k’ - 0xck

This command must be repeated 2x within 100ms or the command will fail; the repeating command must be sequential without any intervening command.

This command functions the same as 0x03 CAL command except this command only affects one key ‘k’ where ‘k’ is from 0 to 47.

The chosen key ‘k’ is recalibrated in its native timeslot; normal running of the part is not interrupted and all other keys operate correctly throughout. This command is for use only during normal operation to try to recover a single key that has failed or is not calibrated correctly.

Returns the 1’s compliment of 0xck just before the key is recalibrated.

5.1 Summary table of commands

Hex	Name	Description	#/Cmd	# rtnd	Rtn range	CRC	Notes
0x00	Null command	Used to get data back in SPI mode	1	1	0..0xFF	-	Flushes pending data from QT; one required to extract each response byte.
0x01	Enter Setups mode	Enter Setups, stop sensing; followed by block load of binary Setups of length 'nn'. Command must be repeated 2x consecutively without any intervening command in 100ms to execute. Sensing auto-restarts.	2 + nn + 2	1+nn+ 1+1	0xFE + 0xFE or + 0x00 (err)	16	First 0xFE issued when ready to get data, second 0xFE issued when all loaded and burned; else timeout. If 2 commands not received in 100ms, times out and no response is issued. Part will timeout if each byte not received within 100ms of previous byte. If CRC failure, returns 0x00 instead of 0xFE Data block length is 'nn' + 2 (CRC-16). CRC is sent LSB first. A CRC of 0x0000 is also acceptable in which case CRC is not checked. The internal EEPROM will be programmed regardless of CRC health, but, if the CRC is bad, the EEPROM will not be marked 'valid config changes' and thus on reset the EEPROM will be restored from flash backup thus overwriting the desired (but corrupt) new setups..
0x03	CAL all	Force device to recalibrate all keys; re-enters RUN mode afterwards automatically; Command must be repeated 2x consecutively without any intervening command in 100ms to execute	2	1	0xFC	-	Returns 1's complement of command to acknowledge cmd once the cal has been scheduled. If 2 commands not received in 100ms, times out and no response is issued.
0x04	Force reset	Force device to reset. Command must be repeated 2x consecutively without any intervening command in 100ms to execute	2	2	0xFB	-	Returns 1's complement of command to acknowledge command prior to reset. If 2 commands not received in 100ms, times out and no response is issued.
0x05	Error status	Get general part status Bit 5 - set if FMEA failure Bit 6 - set if comms error. This bit can be reset by sending cmd 0x0F(last cmd).	1	2	0..0xFF	8	Last return byte is CRC-8 of cmmd + return data
0x06	Report 1st key	Get indication of first touched key + others	1	2	0..0xFF	8	Bit 7 indicates more than one touch, if set. Bit 6 is set if any of the following conditions prevail: calibrating, key(s) failed cal, sync fail, comms error, FMEA failure, EEPROM corrupt. Bits 5..0 indicate first key touched; Bits 5..0 = 0x3F if no touch. 2nd return byte is CRC-8 of cmmd + return data
0x07	Report all keys	Sends back all key detect status bits (bitfield)	1	7	0..0xFF 6 bytes	8	Last return byte is CRC-8 of cmmd + return data
0x08	Signals for all	Sends back all key signal levels	1	96	0..0xFFFF 48 words	-	Returns block data for all keys' signals
0x09	References for all	Sends back all key reference levels	1	96	0..0xFFFF 48 words	-	Returns block data for all keys' references

Hex	Name	Description	#/Cmd	# rtnd	Rtn range	CRC	Notes
0x0a	Deltas for all	Sends back all key delta signals from ref	1	48	0..0xFF 48 bytes	-	Returns block data for all keys' signal deltas from refs; Signed binary: range -127 .. +128. Truncated results (no wrap)
0x0b	Error flags for all	Error bit fields	1	7	0..0xFF 6 bytes	8	Last return byte is CRC-8 of cmmd + return data
0x0c	FMEA status	FMEA bitfield on X, Y lines	1	2	0..0xFF	8	Last return byte is CRC-8 of cmmd + return data
0x0d	Dump Setups	Returns Setups block area followed by CRC.. Scanning is halted and then auto-restarted after the cmd has completed.	1	nn+2	0..0xFF nn+2 bytes	16	Dump of fixed length 'nn' followed by CRC-16 CRC is same as CRC at end of Setups block load. CRC is sent to host LSB first. Part will timeout if each byte not transmitted within 100ms of previous byte. (This can happen if DRDY is driven by the host).
0x0e	Eeprom CRC	Get eeprom CRC	1	2	0..0xFFFF	16	CRC-16 only on Setups array section of eeprom CRC is same as CRC at end of Setups block load. CRC is Tx LSB first.
0x0F	Return last cmmd	Returns last command received	1	1	0..0xFF	-	Returns 1's compliment of last command even if bad
0x10	Version	Code version	1	2	0..0xFF	8	2nd byte is CRC-8 of cmmd + return data
0x11	Return internal code		1	3	0..0xFFFF	8	Returned internal code. 2nd byte is CRC-8 of cmmd + return data
0x12	Return internal code		1	2	0.0xFFFF		
0x16	Sleep	Enter sleep; Command must be repeated 2x consecutively without any intervening command in 100ms to execute.	2	1	0xE9 + 0x01	-	Returns 1's complement of command to acknowledge; wakes on INT, meanwhile sleeps in low power mode; 0x01 when restarted. If 2 commands not received in 100ms, times out and no response is issued. DRDY is held low while the part is asleep. DRDY is released high once awake and ready to return the 0x01.
0x4k	Data for 1 key	Get signal, ref, Norm DI for key k {0..47} Signal: 2 bytes; Ref: 2 bytes; Norm DI: 1 byte	1	5	0..FF Each byte	-	Diagnostic use only, not to be relied upon (no CRC). Signal and ref are Tx as 2 bytes, LSB first.
0x8k	Status for key 'k'	Get status byte for key 'k' {0..47}	1	2	0..FF	8	Second return byte is CRC-8 of cmmd + return data Bit 2 - set if Ref < lower signal limit Bit 3 - set if key detect Bit 4 - set if key enabled
0xck	CAL key 'k'	Force calibration of key # k where k= 0..47. Command must be repeated 2x consecutively without any intervening command in 100ms to execute	2	1	~0xck	-	Used in Run mode. Normal sensing of other keys not affected. CAL of 'k' only takes place in the key's normal timeslot. Returns the ones compliment of the cmd char, once the cal is scheduled.

5 Setups

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. They provide a large number of processing options which can be user-selected to implement very flexible, robust keypanel solutions.

User-defined Setups are employed to alter these algorithms to suit each application. These setups are loaded into the device in a block load over one of the serial interfaces. The Setups are stored in an onboard eeprom array. After a block load, the device should be reset to allow the new Setups block to be shadowed in internal Flash ROM and to allow all the new parameters to take effect.

Refer to Section 6.2, page 17 for a table of all Setups.

Block length issues: The setups block is 247 bytes long to accommodate 48 keys. This can be a burden on smaller host controllers with limited memory. In larger quantities the devices can be procured with the setups block preprogrammed from Quantum. If the application only requires a small number of keys (such as 16) then the setups table can be compressed in the host by filling large stretches of the Setups area with nulls.

Many setups employ lookup-table value translation. The Setups Block Summary on page 19 shows all translation values.

Default Values shown are factory defaults.

5.1 Negative Threshold - NTHR

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. 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; the threshold level is recomputed whenever the reference point moves, and thus it also is drift compensated.

The amount of NTHR required depends on the amount of signal swing that occurs when a key is touched. Thicker panels or smaller key geometries reduce 'key gain', ie signal swing from touch, thus requiring smaller NTHR values to detect touch.

The negative threshold is programmed on a per-key basis using the Setup process. See table, page 19.

Typical values: 3 to 8
(7 to 12 counts of threshold; 4 is internally added to NTHR to generate the threshold).

Default value: 6
(10 counts of threshold)

5.2 Positive Threshold - PTHR

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. This condition is not normal, and usually occurs only after a recalibration when an object is touching the key and is subsequently removed. The desire is normally to recover from these events quickly.

Positive threshold levels are programmed in using the Setup process on a per-key basis.

Typical values: 1 to 4
(5 to 8 counts of threshold; 4 is internally added to PTHR to generate the threshold)

Default value: 2
(6 counts of threshold)

5.3 Drift Compensation - NDRIFT, PDRIFT

Signals can drift because of changes in Cx and Cs over time and temperature. It is crucial that such drift be compensated, else false detections and sensitivity shifts can occur.

Drift compensation (Figure 5-1) 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.

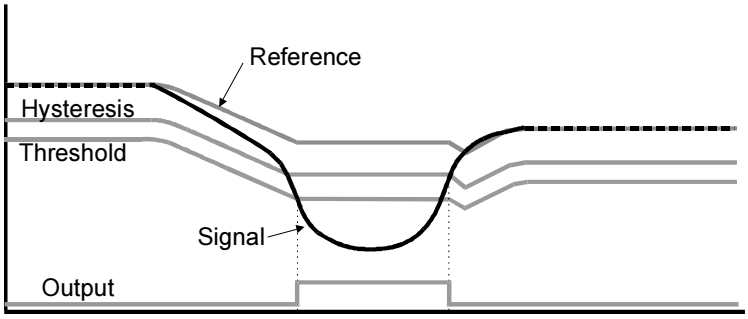
When a finger is sensed, the signal falls since the human body acts to absorb charge from the cross-coupling between X and Y lines. An isolated, untouched foreign object (a coin, or a water film) will cause the signal to rise very slightly due to an enhancement of coupling. This is contrary to the way 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 signal in question has not crossed the negative threshold level.

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 changing the NDRIFT and PDRIFT Setups parameters. This can be done on a per-key basis.

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

Figure 5-1 Thresholds and Drift Compensation



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.

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

NDRIFT Typical values: 9 to 11
(2 to 3.3 seconds per count of drift compensation)

NDRIFT Default value: 10
(2.5s / count of drift compensation)

PDRIFT Typical values: 3 to 5
(0.4 to 0.8 seconds per count of drift compensation; translation via LUT, page 19)

PDRIFT Default value: 4
(0.6s / count of drift compensation)

5.4 Detect Integrators - NDIL, FDIL

To suppress false detections caused by spurious events like electrical noise, the device incorporates a 'detection integrator' or DI counter mechanism that acts to confirm a detection by consensus (all detections in sequence must agree). The DI mechanism counts sequential detections of a key that appears to be touched, after each burst for the key. For a key to be declared touched, the DI mechanism must count to completion without even one detection failure.

The DI mechanism uses two counters. The first is the ‘fast DI’ counter FDIL. When a key’s signal is first noted to be below the negative threshold, the key enters ‘fast burst’ mode. In this mode the burst is rapidly repeated for up to the specified limit count of the fast DI counter. Each key has its own counter and its own specified fast-DI limit (FDIL), which can range from 1 to 15. When fast-burst is entered the QT device locks onto the key and repeats the acquire burst until the fast-DI counter reaches FDIL, or, the detection fails beforehand. After this the device resumes normal keyscanning and goes on to the next key.

The ‘Normal DI’ counter counts the number of times the fast-DI counter reached its FDIL value. The Normal DI counter can only increment once per complete scan of all keys. Only when the Normal DI counter reaches NDIL does the key become formally ‘active’.

The net effect of this is that the sensor can rapidly lock onto and confirm a detection with many confirmations, while still scanning other keys. The ratio of ‘fast’ to ‘normal’ counts is completely user-settable via the Setups process. The total number of required confirmations is equal to FDIL times NDIL.

If FDIL = 5 and NDIL = 2, the total detection confirmations required is 10, even though the device only scanned through all keys only twice.

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

If FDIL = 1, the device functions conventionally; each channel acquires only once in rotation, and the normal detect integrator counter (NDIL) operates to confirm a detection. Fast-DI is in essence not operational.

If FDIL $\geq$ 2, then the fast-DI counter also operates in addition to the NDIL counter.

If Signal $\leq$ NThr: The fast-DI counter is incremented towards FDIL due to touch.

If Signal >NThr then the fast-DI counter is cleared due to lack of touch.

NDIL Typical values: 2, 319

NDIL Default value: 2

FDIL Typical values: 4 to 6

FDIL Default value: 5

5.5 Negative Recal Delay - NRD

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

The Negative Recal Delay timer monitors such detections; if a detection event exceeds the timer's setting, the key will be automatically recalibrated. After a 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 the NRD setup parameter.

NRD can be disabled by setting it to zero (infinite timeout) in which case the key will never auto-recalibrate during a continuous detection (but the host could still command it).

NRD is set using one byte per key, which can range in value from 0..255. NRD is expressed in 0.5s increments. Thus if NRD =120, the timeout value will actually be 60 seconds.

NRD Typical values: 20 to 60 (10 to 30 seconds)

NRD Default value: 20 (10 seconds)

5.6 Positive Recalibration Delay - PRD

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 NRD auto-recalibration, and thereafter the signal rapidly returned to normal (positive excursion).

As an example of the latter, if a foreign object or a finger contacts a key for period longer than the Negative Recal Delay (NRD), 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 quickly so as to restore normal touch operation. The time required to do this is governed by PRD. In order for this to work, the signal must rise through the positive threshold level PTHR continuously for the PRD period.

After the PRD interval has expired and the auto-recalibration has taken place, the affected key will once again function normally. PRD is set on a per-key basis.

PRD Typical values: 5 to 8 (0.7s to 2.0s)

PRD Default value: 6 (1 second)

5.7 Burst Length - BL

The signal gain 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.

QT60xx6 devices use a fixed number of QT cycles which are executed in burst mode. There can be up to 64 QT cycles in a burst, in accordance with the list of permitted values shown in Section 6.5.

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 level (NTHR). Burst length and NTHR interact; normally burst lengths should be kept as short as possible to limit RF emissions, but NTHR should be kept above 6 to reduce false detections due to external noise. The detection integrator mechanism also helps to prevent false detections.

BL Typical values: 2, 3 (48, 64 pulses / burst)

BL Default value: 2 (48 pulses / burst)

5.8 Adjacent Key Suppression - AKS

These devices incorporate adjacent key suppression ('AKS' - patent pending) that can be selected on a per-key basis. AKS permits the suppression of multiple key presses based on relative signal strength. This feature assists in solving the problem of surface moisture which can bridge a key touch to an adjacent key, causing multiple key presses. This feature is also useful for panels with tightly spaced keys, where a fingertip might inadvertently activate an adjacent key.

AKS works for keys that are AKS-enabled anywhere in the matrix and is not restricted to physically adjacent keys; the device has no knowledge of which keys are actually physically adjacent. When enabled for a key, adjacent key suppression causes detections on that key to be suppressed if any other AKS-enabled key in the panel has a more negative signal deviation from its reference.

This feature does not account for varying key gains (burst length) but ignores the actual negative detection threshold setting for the key. If AKS-enabled 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 AKS. The signal threshold of the larger keys can be altered to compensate for this without causing problems with key suppression.

Adjacent key suppression works to augment the natural moisture suppression of narrow gated transfer switches creating a more robust sensing method.

AKS Default value: 0 (Off)

5.9 Oscilloscope Sync - SSYNC

Pin 43 (S_Sync) can output a positive pulse oscilloscope sync that brackets the burst of a selected key. More than one burst can output a sync pulse as determined by the Setups parameter SSYNC for each key.

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

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

SSYNC Default value: 0 (Off)

5.10 Negative Hysteresis - NHYST

The devices employ programmable hysteresis levels of 6.25%, 12.5%, 25%, or 50%. The hysteresis is a percentage of the distance from the threshold level back towards the reference, and defines the point at which a touch detection will drop out. A 12.5% hysteresis point is closer to the threshold level than to the signal reference level.

Hysteresis prevents chatter and works to make key detection more robust. Hysteresis is used only once the key has been declared to be in detection, in order to determined when the key should drop out.

Excessively large amounts of hysteresis can result in 'sticking key' that do not release after touch, especially when signal levels are small. Low amounts of hysteresis can cause key chatter due to low level signal noise or minor amounts of finger motion.

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

NHYST Typical values: 0, 1 (6.25%, 12.5%).

NHYST Default value: 1 (12.5%)

5.11 Dwell Time - DWELL

The Dwell parameter in Setups causes the acquisition pulses to have differing charge capture durations. Generally, shorter durations provide for enhanced surface moisture suppression, while longer durations are usually more compatible with EMC requirements. Longer dwell times permit the use of larger series resistors in the X and Y lines to suppress RFI effects, without compromising key gain.

This parameter lets the designer trade off one requirement for with the other.

DWELL Typical value: 1 (187.5ns)

DWELL Default value: 1 (187.5ns)

5.12 Mains Sync - MSYNC

The MSync feature uses the WS pin. The Sleep and Sync features can be used simultaneously; the part can be put into Sleep mode, but awakened by a mains sync signal at the desired time.

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

Noise such as from 50Hz or 60Hz fields becomes a problem if it is uncorrelated with acquisition signal sampling; uncorrelated noise can cause aliasing effects in the key signals. To suppress this problem the WS input 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 noise sync operating mode is set by parameter MSYNC in Setups.

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, 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.

MSYNC Default value: 0 (Off)

5.13 Burst Spacing - BS

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. The burst spacing can be viewed as a scheduled timeslot in which a burst occurs. This approach results in an orderly and predictable sequencing of key scanning with predictable response times.

Shorter spacings result in a faster response time to touch; longer spacings permit higher burst lengths and longer conversion times but slow down response time.

An automatic setting is also available that performs a 'best fit' timeslot determination for each key's acquisition burst. The fit is determined on power-up each time and is fixed thereafter until reset again.

Standard BS settings from 500µs to 3ms are available.

BS Default value: 0 (Automatic)

5.14 Serial Rate - SR

The possible Baud rates are shown in Section 6.5. The rate chosen by this parameter only affects UART mode. SPI mode is slave-only and can clock at any rate from DC up to 4Mhz.

The Baud rate can be adjusted to one of 5 values from 9600 to 115.2K baud.

SR Default value: 0 (9600 Baud)

5.15 Lower Signal Limit - LSL

This Setup determines the lowest acceptable value of signal level for all keys. If any key's reference level falls below this value, the device declares an error condition.

Testing is required to ensure that there are adequate margins in this determination. Key size, shape, panel material, burst length, and dwell time all factor into the detected signal levels.

LSL Default value: 100

5.16 LED / Alert Output - LED

Refer to Section 6.3 for details.

Pin 40 is designed to drive a low-current LED or to be used as a status and error signalling mechanism for the host controller, primarily for FMEA purposes.

One use for this pin is to alert the host that there is key activity, in order to limit the amount of communication between the device and the host. The LED pin 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.

The table in Section 6.3 shows the possible internal conditions that can cause the LED pin to go active. In addition the LED pin can be made active high or active low. The various items in the table are logical-OR'd together. The LED pin can even be used as a watchdog for the host, to reset it should the host fail to send regular transmissions to the QT (bit 0 of LSL byte).

Note that the LED state will be preserved during sleep.

LED Default value: 0x6c
(see Section 6.3 for details)

5.17 Host CRC - HCRC

The setups block terminates with a 16-bit CRC, HCRC, of the entire block. The formulae for calculating this CRC and the 8-bit CRC also used in the device are shown in Section 8.

6.2 Setups Block Table

Block data is sent from the host to the QT in a block of hex data. The block can only be loaded in Setups mode following two sequential 0x01 commands.

Refer also to Section 6.5, page 19 for further details, and all of Section 5.

Item #	Byte	Parameter	Symbol	Bytes	Valid range	Bits	Key Scope	Default Value	Description	Page
1	0	Neg thresh Pos Thresh	NTHR PTHR	48	NTHR = 0..15 PTHR = 0..15	4 4	1 1	6 2	Lower nibble = Neg Threshold - take operand and add 4 to get value Upper nibble = Pos Threshold - take operand and add 4 to get value	13 13
2	48	Neg Drift Comp Pos Drift Comp	NDRIFT PDRIFT	48	NDRIFT = 0..15 PDRIFT = 0..15	4 4	1 1	10 4	Lower nibble = Neg Drift comp - Via LUT Upper nibble = Pos Drift comp - Via LUT	13
3	96	Normal DI Limit Fast DI Limit	NDIL FDIL	48	NDIL = 0..15 FDIL = 0..15	4 4	1 1	2 5	Lower nibble = Normal DI Limit, values same as operand (0 = disabled burst) Upper nibble = Fast DI Limit, values same as operand (0 does not work)	14
4	144	Neg recal delay	NRD	48	0..255	8	1	20	Range is in 0.5 sec increments; 0 = infinite; default = 10s (operand = 20) Range is 0.5...127s	14
5	192	Pos recal delay Burst Length AKS Scope Sync	PRD BL AKS SSYNC	48	PRD = 0..15 BL = 0..3 AKS = 0, 1 SSYNC = 0, 1	4 2 1 1	1 1 1 1	6 2 0 0	Lower nibble = PRD, via LUT, default = 6 (1 sec) Bits 5, 4: = BL, via LUT, default = 48 (setting =2) Bit 6 = AKS, 1 - enabled Bit 7 = Scope sync, 1 = enabled	14 15 15 15
6	193	Neg Hysteresis Dwell Time Mains Sync	NHYST DWELL MSYNC	1	NHYST = 0..3 DWELL = 0..2 MSYNC = 0, 1	4 2 1	48 48 48	1 1 0	Lower nibble = Neg hysteresis, all keys; default = 12.5% Bits 5, 4 = Dwell time, 3 values via LUT, default = 187.5ns Bits 6 = Mains sync, pos edge sensitive, 1 = enabled; default = 0 (off)	15 15 15
7	194	Burst spacing Serial rate	BS SR	1	BS = 0..11 SR = 0..4	4 4	48 device	0 0	Lower nibble = burst spacing; default = 0 (automatic) Upper nibble = serial rate via LUT - 9600, 19.2K, 38.4K, 57.6K, 115.2K (UART)	16 16
8	196	Lower signal Limit	LSL	2	0..2048	16	48	100	Lower limit of acceptable signal; below this value, declares error	16
9	197	LED Function	LED	1	0..255	8	device	0x6c	Controls what the LED does; see table, below.	16
10	199	Host CRC	HCRC	2	0..65K	16	device	-	CRC-16 of above setups	16
		Block length		**247**						

CRC Note: *A CRC calculator for Windows is available free of charge from Quantum Research on request.*

6.3 LED Function Control Byte Bits

See also page 16. The LED pin can be used to indicate a variety of things in combination. The LED control byte controls which states make the LED pin active. The active state can also be set either high or low by changing bit 7 in this byte. One purpose for these functions is to provide an FMEA-compliant mechanism for fault detection via an alternative path to the serial comms path. Another is to provide an interrupt signal to a host controller to reduce the amount of required comms traffic. Another reason is to simply light an LED on a sensing fault, a keypress, or a comms failure for diagnostic purposes.

Bit	1 =	0 =	Default
7	LED pin is active high polarity	LED pin is active low polarity	0
6	Active on any key error: (cal, cal failed, low sig)	Key errors have no effect	1
5	Active on any keypress	Not active on any keypress	1
4	Active while in sleep	Inactive on Sleep	0
3	Active on eeprom error	Inactive on eeprom error	1
2	Active on Mains sync error	Inactive on Mains sync error	1
1	Active on comms error: LED pin is set active on error, inactive again when 'get last cmd' is called, or part is reset. Error is unrecognised command.	Inactive on comms error	0
0	Host reset. Active if no host comms within any 2s period. Host reset pulse length is 150ms. The host watchdog is not enabled until the first valid cmd is received.	Communications unmonitored	0

6.4 Key Mapping

Several commands return bitfields related to keys. For example, command 0x07 (report all keys) returns 6 bytes containing flag bits, one per key, to indicate which keys are reporting touches. The following table shows the byte and bit order of the keys. The table contains the key number that is returned.

The key number is related to the X and Y scan lines which address each particular key. Each byte in the return stream represents one set of keys along a Y line, ie up to 8 keys. Thus, key 0 is at location X0,Y0 and key 29 is at location X5,Y3. .

		Bit (X line)							
		7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Byte (Y line)	**0**	7	6	5	4	3	2	1	0
	1	15	14	13	12	11	10	9	8
	2	23	22	21	20	19	18	17	16
	3	31	30	29	28	27	26	25	24
	4	39	38	37	36	35	34	33	32
	5	47	46	45	44	43	42	41	40

Note: Byte 0 is returned first.

6.5 Setups Block Summary

Typical values: For most touch applications, use the values shown in the outlined cells. Bold text items indicate default settings.

Index Number	Parameter															
	NTHR counts	PTHR counts	NDRIFT secs	PDRIFT secs	NDIL counts	FDIL counts	NRD secs	PRD secs	BL pulses	AKS	Scope Sync	NHYST	DWELL	MSYNC	Burst spacing	UART Rate
Scope	Per key	Per key	Per key	Per key	Per key	Per key	Per key	Per key	Per key	Per key	Per key	Global	Global	Global	Global	Global
0	4	4	0.1	0.1	Key off	unused	0 (Infinite)	0 (Infinite)	16	**- Off -**	**- Off -**	6.25%	125ns	**- Off -**	**- Auto -**	**-9,600-**
1	5	5	0.2	0.2	1	1	0.5 .. 127s	0.1	32	On	On	**-12.5%-**	**-187.5ns-**	On	500µs	19,200
2	6	**- 6 -**	0.3	0.3	**- 2 -**	2	Default= 10s	0.2	**- 48 -**			25%	312.5ns		750µs	38,400
3	7	7	0.4	0.4	3	3		0.3	64			50%			1,000µs	57,600
4	8	8	0.6	**- 0.6 -**	4	4		0.5							1,250µs	115,200
5	9	9	0.8	0.8	5	**- 5 -**		0.7							1,500µs	
6	**- 10 -**	10	1	1	6	6		**- 1 -**							1,750µs	
7	11	11	1.2	1.2	7	7		1.5							2,000µs	
8	12	12	1.5	1.5	8	8		2							2,250µs	
9	13	13	2	2	9	9		3.2							2,500µs	
10	14	14	**- 2.5 -**	2.5	10	10		4.5							2,750µs	
11	15	15	3.3	3.3	11	11		6							3,000µs	
12	16	16	4.5	4.5	12	12		9								
13	17	17	6	6	13	13		12.3								
14	18	18	7.5	7.5	14	14		17.5								
15	19	19	10	10	15	15		25								
Default	6 (10)	2 (6)	10 (2.5s)	4 (0.6s)	2 (2)	5 (5)	20 (10s)	6 (1s)	2 (48)	0 (off)	0 (off)	1 (12.5)	1 (187.5)	0 (off)	0 (auto)	0 (9600)

7 Specifications

7.1 Absolute Maximum Electrical Specifications

Operating temp. . . . . -40°C to +105°C
Storage temp. . . . . -55°C to +125°C
V_{DD}. . . . . -0.5 to +5.5V
Max continuous pin current, any control or drive pin. . . . . ±10mA
Short circuit duration to ground, any pin. . . . . infinite
Short circuit duration to V_{DD}, any pin. . . . . infinite
Voltage forced onto any pin. . . . . -0.6V to (Vdd + 0.6) Volts
Frequency of operation. 17MHz
Eeprom setups maximum writes. . . . . 100,000 write cycles

7.2 Recommended operating conditions

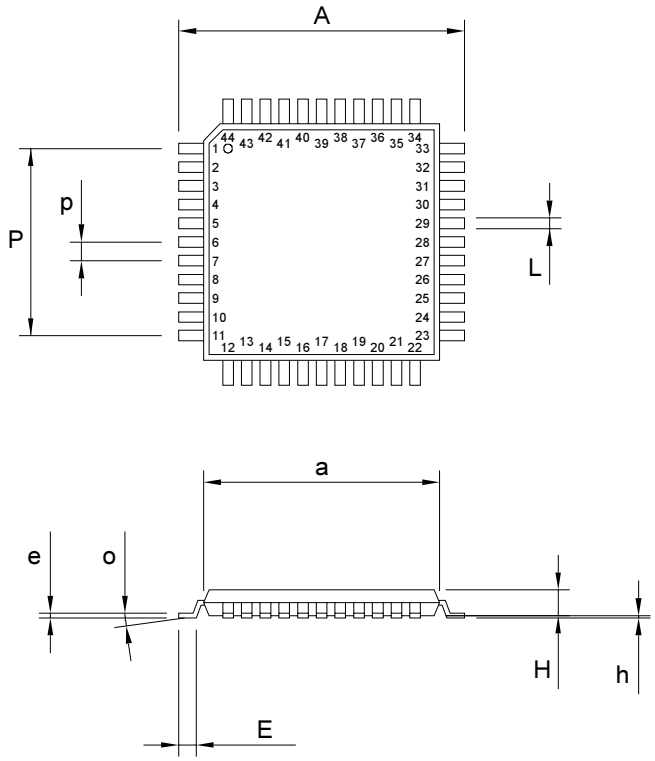
V_{DD}. . . . . +4.75 to 5.25V
Supply ripple+noise. . . . . 5mV p-p max
Cx transverse load capacitance per key. . . . . 0 to 20pF
Fosc oscillator frequency. . . . . 16MHz +/-2%

7.3 DC Specifications

Vdd = 5.0V, Cs = 4.7nF, Freq = 16MHz, Ta = recommended range, unless otherwise noted

Parameter	Description	Min	Typ	Max	Units	Notes
Iddr	Supply current, running			25	mA	Not including external components
Idds	Supply current, sleep		20		µA	Not including external components
Vil	Low input logic level			0.8	V	
Vhl	High input logic level	2.2			V	
Vol	Low output voltage			0.6	V	4mA sink
Voh	High output voltage	Vdd-0.7			V	1mA source
Iil	Input leakage current			±1	µA	
Ar	Acquisition resolution		9	11	bits	
Rp	Pullup resistors	35		120	kohms	DRDY, /SS pins

7.4 Mechanical Dimensions



Package Type: 44 Pin TQFP						
SYMBOL	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
a	9.90	10.10	SQ	0.386	0.394	SQ
A	11.75	12.21	SQ	0.458	0.478	SQ
e	0.09	0.20		0.003	0.008	
E	0.45	0.75		0.018	0.030	
h	0.05	0.15		0.002	0.006	
H	-	1.20		-	0.047	
L	0.30	0.45		0.012	0.018	
p	0.80	0.80	BSC	0.031	0.031	BSC
P	8.00	8.00	BSC	0.315	0.315	BSC
o	0	7		0	7	

7.5 Marking

T_A	TQFP Part Number	Keys	Marking
-40°C to +105°C	QT60166-AS	16	QT60166-A
-40°C to +105°C	QT60246-AS	24	QT60246-A
-40°C to +105°C	QT60326-AS	32	QT60326-A
-40°C to +105°C	QT60486-AS	48	QT60486-A

8 Appendix - CRC Algorithms

8.1 8-Bit CRC Software C Algorithm

```
// 8 bits crc calculation. Initial value is 0
// polynomial = X^8 + X^5 + X^4 + 1
// data is an 8 bit number;  crc is a 8 bit number

int eight_bit_crc(int crc, int data)
{  int   index;          // shift counter
   int   fb;

   index = 8;         // initialise the shift counter
   do
   {  fb = (crc ^ data) & 0x01;
      data >>= 1;
      crc >>= 1;

      If(fb)
      {  crc ^= 0x8c;
      }
   } while(--index);

      return crc;
}
```

8.2 16-Bit CRC Software C Algorithm

```
// 16 bits crc calculation. Initial value is 0
// polynomial = X^16 + X^12 + X^5 + 1
// data is an 8 bit number
// crc is a 16 bit number

long sixteen_bit_crc(long crc, int data)
{  int   index;          // shift counter
   short fb;

   crc ^= long(data) << 8;

   index = 8;
   do
   {  if(crc & 0x8000)
      {  crc= (crc << 1) ^ 0x1021;
      }
      else
      {  crc= crc << 1;
      }
   }while(--index);

   return crc;
}
```

A CRC calculator for Windows is available free of charge from Quantum Research.



Copyright © 2003 QRG Ltd. All rights reserved
Patented and patents pending

Corporate Headquarters
1 Mitchell Point
Ensign Way, Hamble SO31 4RF
Great Britain
Tel: +44 (0)23 8056 5600 Fax: +44 (0)23 8045 3939

admin@qprox.com
www.qprox.com

North America
651 Holiday Drive Bldg. 5 / 300
Pittsburgh, PA 15220 USA
Tel: 412-391-7367 Fax: 412-291-1015

The specifications set out in this document are subject to change without notice. All products sold and services supplied by QRG are subject to our Terms and Conditions of sale and supply of services which are available online at www.qprox.com and are supplied with every order acknowledgement. QProx, QTouch, QMatrix, QLevel, and QSlide are trademarks of QRG. QRG products are not suitable for medical (including lifesaving equipment), safety or mission critical applications or other similar purposes. Except as expressly set out in QRG's Terms and Conditions, no licenses to patents or other intellectual property of QRG (express or implied) are granted by QRG in connection with the sale of QRG products or provision of QRG services. QRG will not be liable for customer product design and customers are entirely responsible for their products and applications which incorporate QRG's products.