Atmel AT42QT2640

64-key 8-slider/wheel QMatrix FMEA IEC/EN/UL60730 Touch Sensor IC

PRELIMINARY DATASHEET

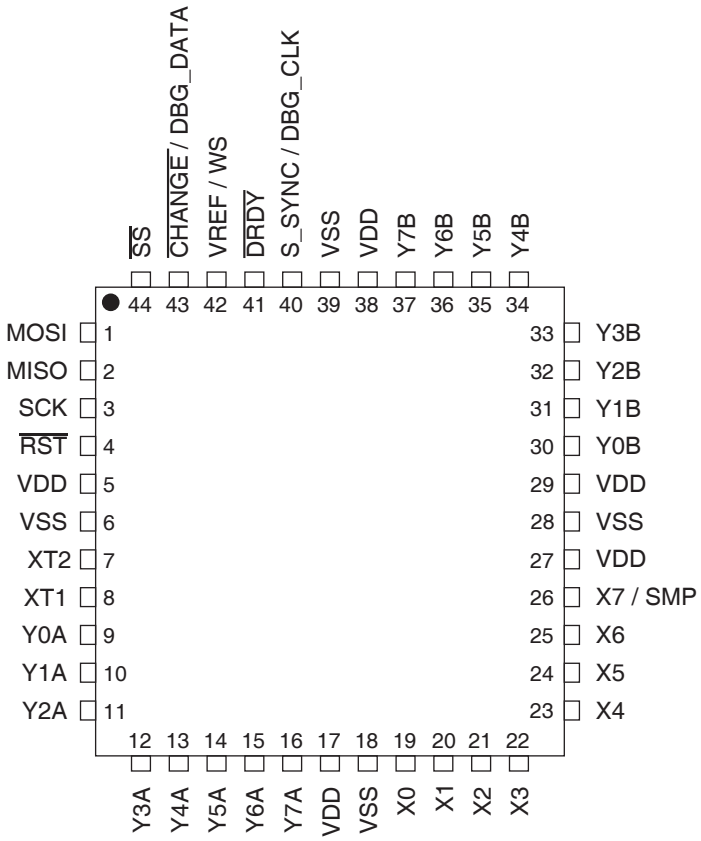
Features

- Number of channels:
  - Up to 64 keys
  - Up to eight sliders/wheels, each with 8-bit resolution
- Technology:
  - Patented charge-transfer (transverse mode), with frequency hopping
- Key outline sizes:
  - 5 mm × 5 mm or larger (panel thickness dependent); widely different sizes and shapes possible
- Key spacings:
  - 6 mm or wider, center to center (panel thickness dependent)
- Electrode design:
  - Two-part electrode shapes (drive-receive); wide variety of possible layouts
- Layers required:
  - One layer (with jumpers), two layers (no jumpers)
- Electrode materials:
  - PCB, FPCB, silver or carbon on film, ITO on film
- Panel materials:
  - Plastic, glass, composites, painted surfaces (low particle density metallic paints possible)
- Adjacent Metal:
  - Compatible with grounded metal immediately next to keys
- Panel thickness:
  - Up to 50 mm glass, 20 mm plastic (key size dependent)
- Key sensitivity:
  - Individually settable over serial interface
- Interfaces:
  - SPI slave (4 MHz max. clock)
  - CHANGE status indication pin
  - Debug output
- Signal processing:
  - Self-calibration, auto drift compensation, noise filtering, Adjacent Key Suppression® (AKS®)
- FMEA compliant design features

- IEC/EN/UL60730 compliant design features
  - UL approval
  - VDE compliance
  - For use in both class B and class C safety-critical products
- Detects and Reports Key Failure
- Power:
  - 4.75 V to 5.25 V
- Package:
  - 44-pin 10 × 10 mm TQFP RoHS compliant

1. Pinout and Schematic

1.1 Pinout Configuration



1.2 Pin Descriptions

Table 1-1. Pin Listing

Pin	Name	Type	Comments	If Unused, connect To...
1	MOSI	I	SPI data input	–
2	MISO	O	SPI data output	–
3	SCK	I	SPI clock input	–
4	$\overline{\text{RST}}$	I	Reset low; has internal 30 kΩ – 60 kΩ pull-up resistor. This pin should be controlled by the host.	Vdd
5	VDD	P	Power	–
6	VSS	P	Ground	–
7	XT2	O	Ceramic resonator or crystal,16 MHz	–
8	XT1	I		–
9	Y0A	I/O	Y line connection	Leave open
10	Y1A	I/O	Y line connection	Leave open
11	Y2A	I/O	Y line connection	Leave open
12	Y3A	I/O	Y line connection	Leave open
13	Y4A	I/O	Y line connection	Leave open
14	Y5A	I/O	Y line connection	Leave open
15	Y6A	I/O	Y line connection	Leave open
16	Y7A	I/O	Y line connection	Leave open
17	VDD	P	Power	–
18	VSS	P	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 / SMP	O	X matrix drive line / Sample output	Leave open / –
27	VDD	P	Power	–
28	VSS	P	Ground	–
29	VDD	P	Power	–

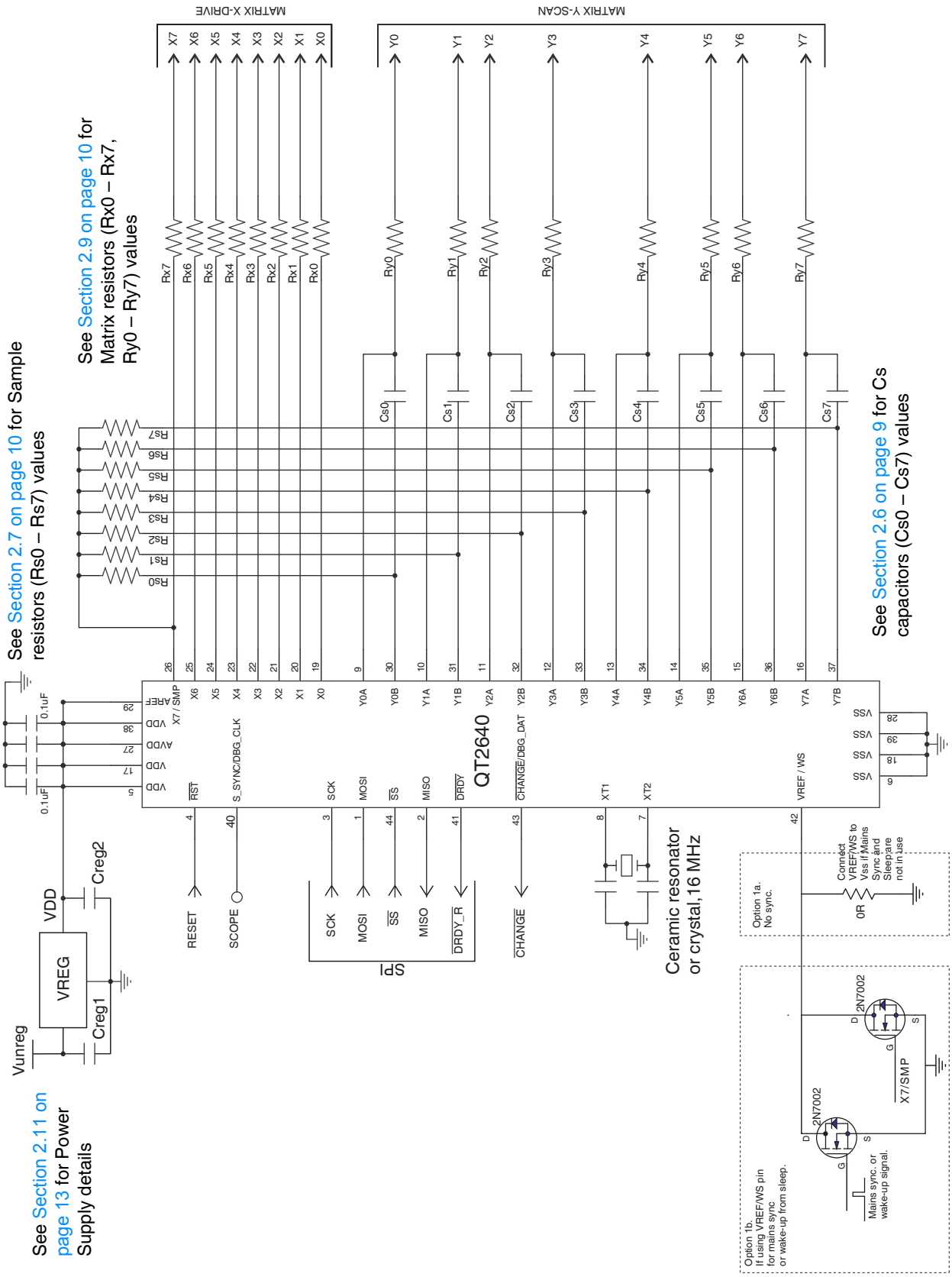
Table 1-1. Pin Listing (Continued)

Pin	Name	Type	Comments	If Unused, connect To...
30	Y0B	I/O	Y line connection	Leave open
31	Y1B	I/O	Y line connection	Leave open
32	Y2B	I/O	Y line connection	Leave open
33	Y3B	I/O	Y line connection	Leave open
34	Y4B	I/O	Y line connection	Leave open
35	Y5B	I/O	Y line connection	Leave open
36	Y6B	I/O	Y line connection	Leave open
37	Y7B	I/O	Y line connection	Leave open
38	VDD	P	Power	–
39	VSS	P	Ground	–
4O	S_SYNC / DBG_CLK	O	Scope Synchronization output or Debug Clock	Leave open
41	$\overline{\text{DRDY}}$	OD	This pin MUST be used. 1 = comms ready; need a 100 µs grace period before checking. Open-drain with internal 20 kΩ – 50 kΩ pull-up resistor	–
42	VREF/ WS	I	Connect to Vss unless using sleep or mains sync / Wake-up from sleep input and/or sync input	– VSS
43	$\overline{\text{CHANGE}}$ / DBG_DATA	OD / O	Key touch change, active low. Has internal 20 kΩ – 50 kΩ pull-up resistor. / Debug Data	Leave open
44	$\overline{\text{SS}}$	I	SPI slave select; has internal 20 kΩ – 50 kΩ pull-up resistor	–

I	Input only	O	Output only, push-pullI/O	Input/output
OD	Open drain output	P	Ground or power	

1.3 Schematic

Figure 1-1. Typical Circuit



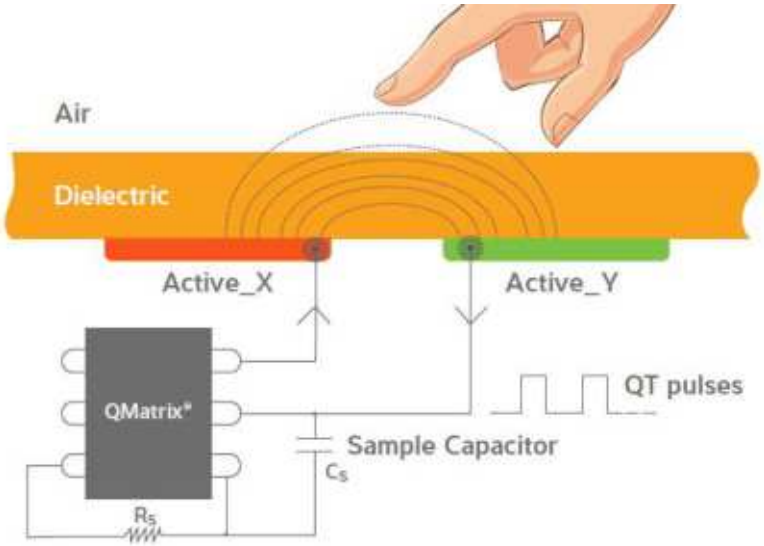
2. Hardware and Functional

2.1 Introduction

The AT42QT2640 (QT2640) is a digital burst-mode sensor, designed specifically for QMatrix layout touch controls; it includes 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 a few square centimeters of single-sided PCB area. CEM-1 and FR1 punched, single-sided materials can be used for the lowest possible cost. The PCB rear can be mounted flush on the back of a glass or plastic panel using a conventional adhesive, such as 3M VHB two-sided adhesive acrylic film.

The QT2640 employs QMatrix transverse charge-transfer (QT™) sensing – a technology that senses changes in electrical charge forced across two electrode elements by a pulse edge (see Figure 2-1).

Figure 2-1. Field Flow Between X and Y Elements



The QT2640 allows a wide range of key sizes and shapes to be mixed together in a single touch panel, and is designed for use with up to 64 keys, and up to eight sliders and wheels, or a mixture of keys, sliders and wheels.

The QT2640 uses a memory mapped SPI interface to allow key data to be extracted and to permit individual key parameter setup. The 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 QT2640 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.

A Debug output interface is also supported, which can be used to monitor many operating variables during product development.

The QT2640 incorporates many tests and checks to enable a product to achieve FMEA and IEC/EN/UL60730 compliance. The results of some tests need to be checked by the host. To achieve a compliant design, the host must read back the test results and confirm their validity.

The QT2640 is able to scan the touch matrix twice as fast as previous generation devices; it can take twice the number of samples in a given time frame. This means the QT2640 is much better equipped to continue normal operation in the face of heavy noise.

See Appendix C. on page 69 for information on conducted noise immunity.

2.2 Key Numbers

The keys are numbered from 0 – 63. Table 2-1 shows the key numbering.

Table 2-1. Key Numbers

	X7	X6	X5	X4	X3	X2	X1	X0	
Y0	7	6	5	4	3	2	1	0	Key numbers
Y1	15	14	13	12	11	10	9	8	
Y2	23	22	21	20	19	18	17	16	
Y3	31	30	29	28	27	26	25	24	
Y4	39	38	37	36	35	34	33	32	
Y5	47	46	45	44	43	42	41	40	
Y6	55	54	53	52	51	50	49	48	
Y7	63	62	61	60	59	58	57	56	

2.3 Matrix Scan Sequence

Key scanning begins with location X = 0, Y = 0 (key 0). All keys on X0 are scanned first, then X1 and finishing with all keys on X7 (for example, the sequence X0Y0, X0Y1 – X0Y7, X1Y0 – X1Y7 … X7Y1 – X7Y6, X7Y7). Table 2-1 shows the key numbering.

All keys on the same X line are excited together in a burst of acquisition pulses whose length is determined by the setups parameter BL (see Section 5.5 on page 41); this can be set to a different value for each key. A burst is completed entirely before the next X line is excited. At the end of each burst the resulting signals, one for each Y line, are converted to digital form and processed. The burst length directly impacts key gain. Each key can have a different burst length in order to allow tailoring of key sensitivity. Although all keys on an entire X line are excited simultaneously, the charge is selectively captured at each Y line according to the burst length selected.

2.4 Enabling/Disabling Keys – Burst Removal

Unused keys are always removed from the computation sequence in order to optimize speed. If all keys are disabled on any given X, the entire X line is also removed from the burst sequence. If only two X lines have enabled keys, only two timeslots are used for scanning.

The NDIL parameter is used to enable and disable keys in the matrix. Setting NDIL = 0 for a key disables it (Section 5.3 on page 39). Keys that are disabled are eliminated from the scan sequence to save scan time and thus power. If all keys on an X line are disabled, the burst for the entire X line is removed from the scan sequence, further saving time and power. This has the consequence of affecting the scan rate of the entire matrix as well as the time required for initial matrix calibration. It does not affect the time required to calibrate an individual key once the matrix is initially calibrated after power-up or reset.

It is very important that only those keys that physically exist are enabled. All non-existent keys must be disabled (NDIL = 0) otherwise other keys in the matrix can incorrectly report their signal as zero.

2.5 Oscillator

The oscillator can use either a quartz crystal or a ceramic resonator. In all cases, XT1 and XT2 must both be loaded with low-value capacitors to ground. These capacitors should be in the range 12 pF to 22 pF. Follow the manufacturer's recommendations for the appropriate value within this range. Resonators and crystals requiring loading capacitors outside this range are unsuitable for operation with the QT2640.

A resistor of value 1MΩ is connected internally between XT1 and XT2.

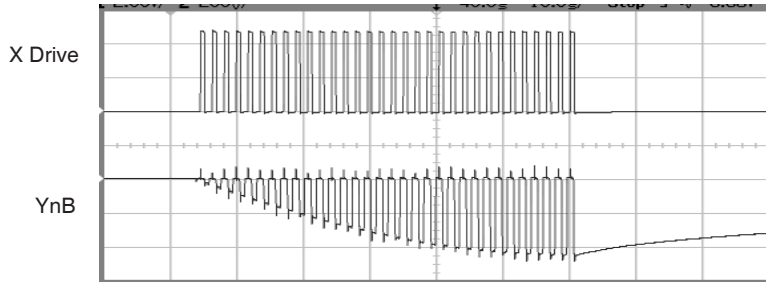
The frequency of oscillation should be 16 MHz ±2%.

2.6 Sample Capacitor; Saturation Effects

The charge sampler capacitors on the Y pins (Cs0 – Cs7) should be NPO (preferred), X7R ceramics or PPS film; NPO offers the best stability. The value of these capacitors is not critical but 4.7 nF is recommended for most cases.

Cs voltage saturation is shown in Figure 2-2. This nonlinearity is caused by excessive voltage accumulation on Cs inducing conduction in the pin protection diodes. This badly saturated signal destroys key gain and introduces a strong thermal coefficient which can cause phantom detection.

Figure 2-2. VCs – Nonlinear During Burst (Burst too long, or Cs too small, or X-Y transcapacitance too large)



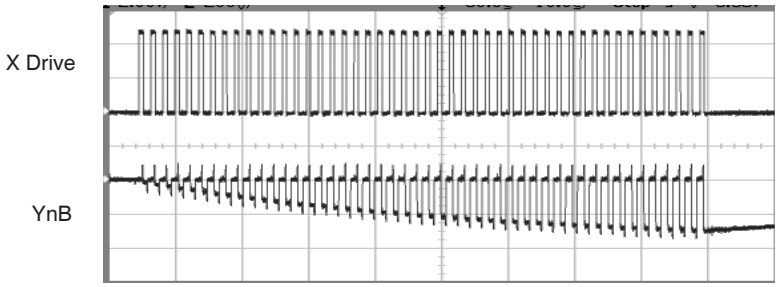
The cause of this is either from the burst length being too long, the Cs value being too small, or the X-Y transfer coupling being too large. Solutions include loosening up the interdigitation of key structures, greater separation of the X and Y lines on the PCB, increasing Cs, and decreasing the burst length.

Increasing Cs makes the part slower; decreasing burst length makes it less sensitive. A better PCB layout and a looser key structure (up to a point) have no negative effects.

Cs voltages should be observed on an oscilloscope with the matrix layer bonded to the panel material; if the Rs side of any Cs ramp is more negative than –0.25 V during any burst (not counting overshoot spikes which are probe artifacts), there is a potential saturation problem.

Figure 2-3 shows a defective waveform similar to that of Figure 2-2, but in this case the distortion is caused by excessive stray capacitance coupling from the Y line to AC ground; for example, from running too near and too far alongside a ground trace, ground plane, or other traces. The excess coupling causes the charge-transfer effect to dissipate a significant portion of the received charge from a key into the stray capacitance.

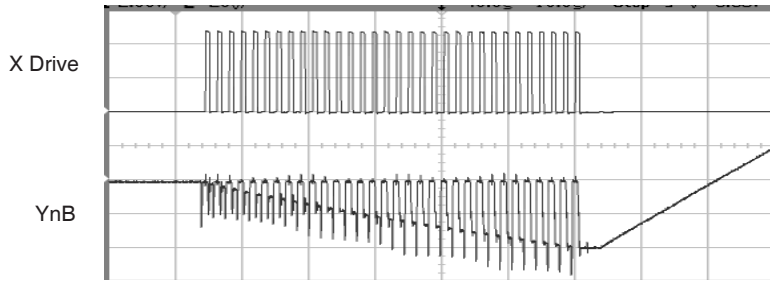
Figure 2-3. VCs – Poor Gain, Nonlinear During Burst (Excess capacitance from Y line to Gnd)



This phenomenon is more subtle; it can be best detected by increasing BL to a high count and watching what the waveform does as it descends towards and below –0.25 V. The waveform appears deceptively straight, but it slowly starts to flatten even before the –0.25 V level is reached.

A correct waveform is shown in Figure 2-4. Note that the bottom edge of the bottom trace is substantially straight (ignoring the downward spikes).

Figure 2-4. VCs – Correct



Unlike other QT circuits, the Cs capacitor values on QT2640 have no effect on conversion gain. However, they do affect conversion time.

Unused Y lines should be left open.

2.7 Sample Resistors

The sample resistors (Rs0 – Rs7) are used to perform single-slope analog-to-digital (ADC) conversion of the acquired charge on each Cs capacitor. These resistors directly control acquisition gain; larger values of Rs proportionately increase signal gain. Values of Rs can range from 220 kΩ to 4.7 MΩ. A value of 470 kΩ is typical for most purposes.

Unused Y lines do not require an Rs resistor.

2.8 Signal Levels

Using Atmel'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 of 250 to 750 counts with properly designed key shapes (see the *Touch Sensors Design Guide*, available on Atmel's website www.atmel.com). However, long adjacent runs of X and Y lines can also artificially boost the signal values, and induce signal saturation: this is to be avoided. The X-to-Y coupling should come mostly from intra-key electrode coupling, not from stray X-to-Y trace coupling.

QmBtn software is available free of charge on the Atmel 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 increases the signal strengths as will increasing the sampling resistor (Rs) values.

2.9 Matrix Series Resistors

The X and Y matrix scan lines should use series resistors (Rx0 – Rx7 and Ry0 – Ry7 respectively) for improved EMC performance (Figure 1-1 on page 6).

X drive lines require Rx in most cases to reduce edge rates and thus reduce RF emissions. Values range from 1 kΩ to 100 kΩ, typically 1 kΩ.

Y lines need Ry to reduce EMC susceptibility problems and in some extreme cases, ESD. Values range from 1 kΩ to 100 kΩ, typically 1 kΩ. Y resistors act to reduce noise susceptibility problems by forming a natural low-pass filter with the Cs capacitors.

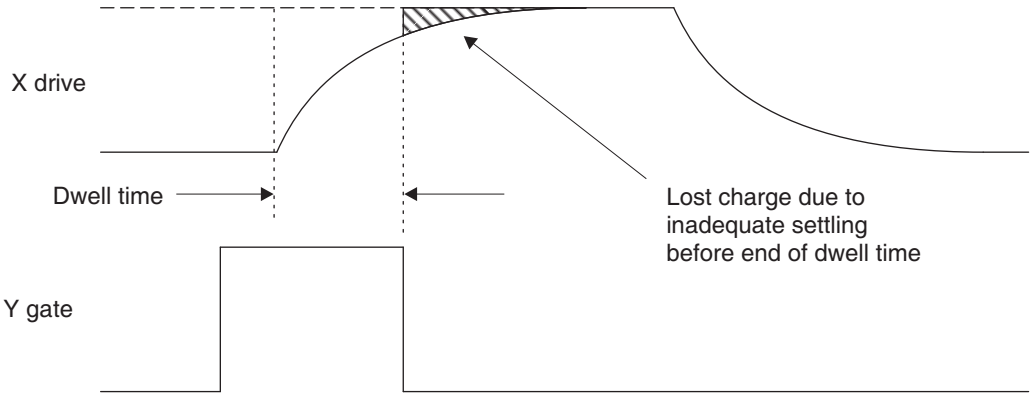
It is essential that the Rx and Ry resistors and Cs capacitors be placed very close to the chip. Placing these parts more than a few millimeters away opens the circuit up to high frequency interference problems (above 20 MHz) as the trace lengths between the components and the chip start to act as RF antennas.

The upper limits of Rx and Ry are reached when the signal level and hence key sensitivity are clearly reduced. The limits of Rx and Ry depend on key geometry and stray capacitance, and thus an oscilloscope is required to determine optimum values of both.

Dwell time is the duration in which charge coupled from X to Y is captured (Figure 2-5 on page 11). Increasing the dwell time increases the signal levels lost to higher values of Rx and Ry, as shown in Figure 2-5. Too short a dwell time causes charge to be 'lost', if there is too much rising edge roll-off. Lengthening the dwell time causes this lost charge to be recaptured, thereby restoring key sensitivity. In the QT2640 dwell time is adjustable (see Section 5.8 on page 46).

Dwell time problems can also be solved by either reducing the stray capacitance on the X line(s) (by a layout change – for example, by reducing X line exposure to nearby ground planes or traces) or the Rx resistor needs to be reduced in value (or a combination of both approaches).

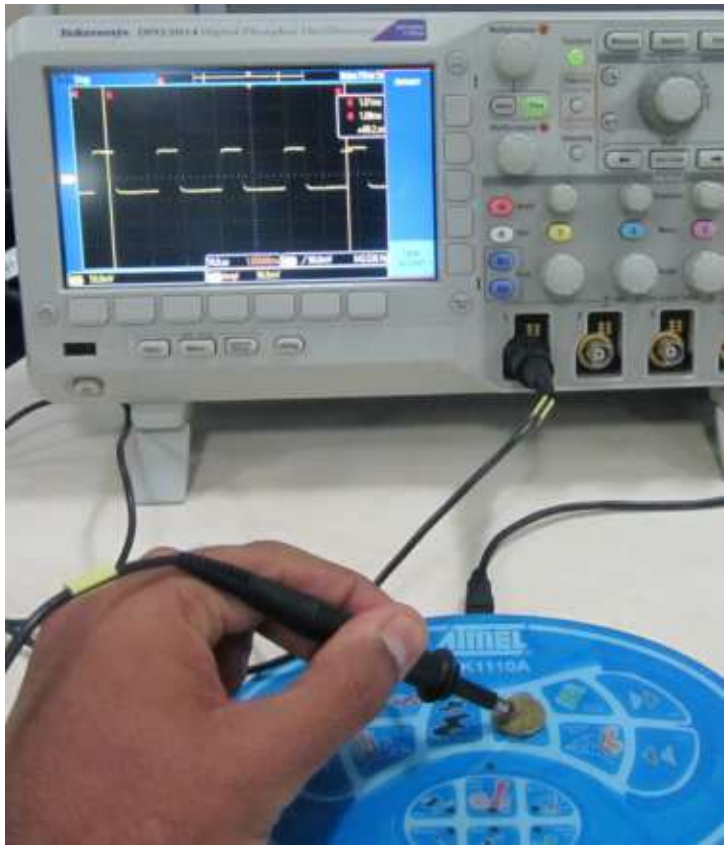
Figure 2-5. Drive Pulse Roll-off and Dwell Time



Note: The Dwell time is a minimum of ~125 ns – see Section 5.8 on page 46

One way to determine X line settling time is to monitor the fields using a patch of metal foil or a small coin over the key (see Figure 2-6). Only one key along a particular X line needs to be observed, as each of the keys along a particular X line are identical. The dwell time should exceed the observed 95% settling of the X-pulse by 25% or more.

Figure 2-6. Probing X-Drive Waveforms With a Coin



2.10 PCB Layout, Construction

2.10.1 Overview

It is best to place the chip near the touch keys on the same PCB so as to reduce X and Y trace lengths, thereby reducing the chances for EMC problems. Long connection traces act as RF antennas. The Y (receive) lines are much more susceptible to noise pickup than the X (drive) lines.

Even more importantly, all signal related discrete parts (resistors and capacitors) should be very close to the body of the chip. Wiring between the chip and the various resistors and capacitors should be as short and direct as possible to suppress noise pickup.

Ground planes and traces should NOT be used around the keys and the Y lines from the keys. Ground areas, traces, and other adjacent signal conductors that act as AC ground (such as Vdd) absorb the received key signals and reduce signal-to-noise ratio (SNR) and thus are counterproductive. Ground planes around keys also make water film effects worse.

Ground planes, if used, should be placed under or around the QT2640 chip itself and the associated resistors and capacitors in the circuit, under or around the power supply, and back to a connector, but nowhere else.

2.10.2 LED Traces and Other Switching Signals

Digital switching signals near the Y lines induce transients into the acquired signals, deteriorating the SNR performance of the QT2640. Such signals should be routed away from the Y lines, or the design should be such that these lines are not switched during the course of signal acquisition (bursts).

LED terminals which are multiplexed or switched into a floating state and which are within or physically very near a key structure (even if on another nearby PCB) should be bypassed to either Vss or Vdd with at least a 10 nF capacitor of any type, to suppress capacitive coupling effects which can induce false signal shifts. LED terminals which are constantly connected to Vss or Vdd do not need further bypassing.

2.10.3 PCB Cleanliness

Modern no-clean flux is generally compatible with capacitive sensing circuits.



CAUTION: If a PCB is reworked in any way, it is highly likely that the behavior of no-clean flux will change. This can mean that the flux changes from an inert material to one that can absorb moisture and dramatically affect capacitive measurements due to additional leakage currents. If so, the circuit can become erratic and exhibit poor environmental stability.

If a PCB is reworked in any way, clean it thoroughly to remove all traces of the flux residue around the capacitive sensor components. Dry it thoroughly before any further testing is conducted.

2.11 Power Supply Considerations

For Vdd information see Section 6.1 and Section 6.2 on page 59.

As the QT2640 uses the power supply as an analog reference, the power should be very clean and come from a separate regulator. A standard inexpensive Low Dropout (LDO) type regulator should be used; it should not also be used to power other loads such as relays or other high current devices. Load shifts on the output of the LDO can cause Vdd to fluctuate enough to cause false detection or sensitivity shifts.

Ceramic 0.1 µF bypass capacitors should be placed very close and routed with short traces to all power pins of the IC. There should be at least four such capacitors around the part.

2.12 Startup/Calibration Times

The QT2640 employs a rigorous initialization and self-check sequence for IEC/EN/UL60730 compliance. If the self-tests are passed, the last step in this sequence enables the serial communication interface. The communication interface is not enabled if a safety critical fault is detected during the startup sequence. The QT2640 requires initialization times as follows:

1. Normal reset to ability to communicate: 110 ms.
2. From very first power-up to ability to communicate:
   4.7 s (one time event to initialize all of EEPROM, or to recover EEPROM copy from Flash in the event of EEPROM corruption).
3. From power-up to ability to communicate:
   190 ms in the event the setups have been changed and the part needs to back up the EEPROM to Flash.

The QT2640 determines a reference level for each key by calibrating all the keys immediately after initialization. Each key is calibrated independently and in parallel with all other enabled keys. Calibration takes between 11 and 62 keyscan cycles; each cycle being made up of one sample from each enabled key. The QT2640 ends calibration for a key if its reference has converged with the signal DC level. The calibration time is shortest when the keys signals are stable, typically increasing with increasing noise levels to the maximum of 62 keyscan cycles.

An error is reported for each key where calibration continues for the maximum number of keyscan cycles and the key's reference does not appear to have converged with the signals DC level. Noise levels can vary from key to key such that some keys may take longer to calibrate than others. However, the QT2640 can report during this interval that the key(s) affected are still in calibration via the QT2640 status bits. Table 2-2 shows keyscan cycle times and calibration times per key versus dwell time and burst length for all 64 keys enabled. The values given assume the factory default settings except where noted.

Table 2-2. Keyscan Cycle and Calibration Times

Setups	Keyscan Cycle Time	Calibration Time (min)	Calibration Time (max)
BL = 0 (16 pulses) DWELL = 0 (125 ns) Rs = 470 kW Signal level = 200 counts	7 ms	77 ms (11 × 7)	434 ms (62 × 7)

2.13 Reset Input

Should communications with the QT2640 be lost, the $\overline{\text{RST}}$ pin can be used to reset the QT2640 to simulate a power-down cycle, in order to then bring the QT2640 up into a known state. The pin is active low, and a low pulse lasting at least 10 µs must be applied to this pin to cause a reset.

To provide for proper operation during power transitions the QT2640 has an internal brownout detector set to 4 V.

The reset pin has an internal 30 kΩ – 60 kΩ resistor. A 2.2 µF capacitor plus a diode to Vdd can be connected to this pin as a traditional reset circuit, but this is not necessary.

Where the QT2640 has detected a failure of one of the internal IEC/EN/UL60730 checks and has subsequently locked up in an infinite loop, only a power cycle or an external hardware reset can restore normal operation. It is strongly recommended that the host has control over the $\overline{\text{RST}}$ pin.

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

2.14 Detection Integrators

See also Section 5.3 on page 39.

The device features a detection integration mechanism, which acts to confirm a detection in a robust fashion. A per-key counter is incremented each time the key has exceeded its threshold and is decremented each time the key does not exceed its threshold. When this counter reaches a preset limit the key is finally declared to be touched.

For example, if the limit value is 10, then the device has to exceed its threshold and stay there for a minimum of 10 acquisitions before the key is declared to be touched.

The QT2640 uses a two-tier confirmation mechanism having two such counters for each key. These can be thought of as inner loop and outer loop confirmation counters. The inner counter is referred to as the fast DI; this acts to confirm a detection via rapid successive acquisition bursts, at the expense of delaying the sampling of the next key. Each key has its own fast DI counter and limit value; these limits can be changed via the setups block on a per-key basis.

The outer counter is referred to as the normal DI; this DI counter increments whenever the fast DI counter has reached its limit value. The normal DI counter also has a limit value which can be set on a per-key basis. If a normal DI counter reaches its terminal count, the corresponding key is declared to be touched and becomes active. Note that the normal DI can only be incremented once per complete keyscan cycle; that is, more slowly, whereas the fast DI is incremented “on the spot” without interruption.

The net effect of this mechanism is a multiplication of the inner and outer counters and hence a highly noise-resistant sensing method. If the inner limit is set to 5, and the outer to 3, the net effect is a minimum of 5 × 3 = 15 threshold crossings to declare a key as active.

2.15 Sleep

The QT2640 can be configured for automatic sleep using the Sleep Drift Compensation (SDC) setup together with the Request Sleep command, and woken with a rising signal edge applied to the VREF/WS pin.

If the sleep feature is enabled using SDC (see Section 5.7.3 on page 45), and the Request Sleep command has been issued (Section 4.13 on page 35), the QT2640 sleeps whenever possible to conserve power. Periodically, it should be woken by the host using the VREF/WS pin. Upon being woken, the matrix is scanned and the QT2640 returns to sleep unless there is activity which demands further attention. The QT2640 returns to sleep automatically after a period of inactivity, the duration of which is defined by the AWAKE feature (see Section 5.12 on page 48).

At least one full matrix scan is always performed after waking up and before returning to sleep. At the end of each matrix scan, the part returns to sleep unless recent activity, such as a touch event, demands further attention. If there has been recent activity, the part performs another complete matrix scan before attempting to sleep once again. This process is repeated indefinitely until the activity stops and the part returns to sleep, or the Request Sleep command is revoked; The Request Sleep command is revoked upon receipt of the first byte of an SPI communications sequence.

Key touch activity forces the matrix scanning into free run whereby each matrix scan is not interleaved with sleep. The QT2640 will not sleep if any of the following conditions are present:

- SDC = 0
- The Request Sleep command has not been issued, or has been revoked
- The timer configured with AWAKE is running
- $\overline{\text{DRDY}}$ asserted (low level)
- $\overline{\text{SS}}$ low (assume host trying to send a command)
- A command is being processed or response data is being returned or pending return to the host
- Any key calibrating
- Any key in detect
- Any key touch delta exceeds the threshold (positive or negative)

Sleep should be used with caution if the QT2640 is being used in an FMEA or IEC/EN/UL60730 compliant design because all operations are stopped within the QT2640 while the part is asleep and the host might have difficulty distinguishing between the IEC/EN/UL60730 counters appearing to run slow because the part is intermittently sleeping, and faulty operation. However, in the knowledge it has configured the QT2640 for sleep, the host can take this into account. For example, the host could wake the QT2640 at suitable intervals, check for correct operation and then return the QT2640 to sleep.

2.16 FMEA Tests

Failure Modes and Effects Analysis (FMEA) is a tool used to determine critical failure problems in control systems. FMEA analysis is being applied increasingly to a wide variety of applications including domestic appliances. To survive FMEA testing the control board must survive any single problem in a way that the overall product can either continue to operate in a safe way, or shut down.

The most common FMEA requirements regard opens and shorts analysis of adjacent pins on components and connectors. However, other criteria must usually be taken into account, for example complete QT2640 failure.

The QT2640 incorporates a number of special self-test features which allow products to pass such FMEA tests easily, and enable key failure to be detected. These tests are performed in an extra burst slot after the last enabled key.

The sequence of tests are performed repeatedly during normal running once all initialization is complete. During initialization, all FMEA error flags are cleared. Any FMEA errors are reported as the tests are performed for the first time.

The FMEA testing is done on all enabled keys in the matrix, and results are reported via the serial interface. Disabled keys are not tested.

Assuming the part does not sleep, the interval from the start of one set of FMEA tests to the start of the next set, never exceeds 2 s.

Also, since the QT2640 only communicates in slave mode, the host can determine immediately if the QT2640 has suffered a catastrophic failure.

The FMEA tests performed are:

- X drive line shorts to Vdd and Vss
- X drive line shorts to other pins
- X drive signal deviation
- Y line shorts to Vdd and Vss
- Y line shorts to other pins
- X to Y line shorts
- Cs capacitor checks including shorts and opens
- Vref test
- Key gain (see Section 5.10.1 on page 47)

Other tests incorporated into the QT2640 include:

- A test for signal levels against a preset minimum value (Lower Signal Limit (LSL) setup, see Section 5.9 on page 46). If any signal level falls below this level, an error flag is generated.
- 16-bit CRC communications checks on all data returns.

2.17 IEC/EN/UL60730 Compliance

The QT2640 also incorporates special test features which, together with the FMEA tests, allow products to achieve IEC/EN/UL60730 compliance with ease. IEC/EN/UL60730 compliance demands dynamic verification of all safety related components and sub-components within a product. The QT2640 is able to verify some sub-components internally, but others require verification by a separate, independent processing unit with another timing source.

To this end the QT2640 exposes a number of internal operating parameters through its serial communications interface and requires the cooperation of a host to check and verify these parameters regularly. It is also necessary for the host to verify the communications by checking and validating the CRC, which the QT2640 appends to data returns. If a CRC check should fail, the host should not rely on the data but retry the transmission.

Occasional CRC failures might be anticipated as a result of noise spikes. Repeated CRC failures might indicate a safety-critical failure. Where the QT2640 is able to verify sub-components internally, but any such verification fails, the QT2640 disables serial communication and locks up in an infinite loop. The host can detect this condition if repeated CRC failures are observed.

During normal operation the host must perform regular reads of the IEC/EN/UL60730 counters (see Section 4.3 on page 31) to verify correct operation of the QT2640. The host must also perform regular reads of the QT2640 status (see Section 4.4 on page 32) and verify there are no errors reported. The FMEA error flag, LSL error flag and Host CRC error flag must all be considered as part of an IEC/EN/UL60730 compliant design.

The host can try to recover from any safety critical failure by resetting the QT2640 using its $\overline{\text{RST}}$ pin. The host should allow a grace period in consideration of the start-up and initialisation time the QT2640 requires after reset to ability to communicate (see Section 2.12 on page 13).

The sub-components that the QT2640 is able to verify internally are tested repeatedly during the normal running of the device, and the various tests run in parallel. As each test ends the result is recorded and the test is restarted. The real time that elapses from the start of each test to the start of the next iteration of the same test is called the failure detect time, or hazard time, the maximum time for which an error could be undetected.

Each test is broken down into a number of smaller parts, each of which is processed in turn during each matrix scan. Each test is therefore completed either after a number of matrix scans, as shown in Table 2-3.

Table 2-3. Test run times (measured in matrix scans)

Test	Required Matrix Scans to complete test
FMEA	8
Other	18
Variable Memory	2304
Firmware CRC	1984
Setups CRC	60

Table 2-4 shows matrix scan times for setups that yield the shortest matrix scan time and a much longer scan time resulting from the use of long dwell and low frequency settings.

Table 2-4. Matrix Scan Times

Setups Conditions	Matrix Scan Time (ms)
BL = 0 (16 pulses), DWELL = 0 (0.13 µs), FREQ0 = 1, All keys enabled, FHM = 0, MSYNC = 0 (Off), SDC = 0 (sleep disabled), DEBUG = 0 (Off).	8.5
BL = 3 (64 pulses), DWELL = 13 (5.1 µs), FREQ0 = 25, All keys enabled, FHM = 0, MSYNC = 0 (Off), SDC = 0 (sleep disabled), DEBUG = 0 (Off).	17

Longer matrix scan times are possible than those shown in Table 2-4 by using even longer dwell times and higher values for FREQ0 (lower burst frequencies), but these are considered extreme settings.

Table 2-5 shows the failure detect times for the internal tests assuming a matrix scan time of 9 ms, which is valid for typical setups.

Table 2-5. Failure Detect Time

Test	Failure Detect Time (ms)
FMEA	72
Other	162
Variable Memory	20,736
Firmware CRC	17,856
Setups CRC	540
Note: Conditions: Matrix scan time = 9ms. QT2640 does not sleep for duration of tests.	

Longer failure detect times are possible than those shown in Table 2-5 where the matrix scan time is longer. The failure detect times are proportional to the matrix scan time. The failure detect time for other setups can therefore be determined by observing the matrix scan time using an oscilloscope and scaling the times given in Table 2-5 accordingly. Alternatively, the failure detect times can be calculated by taking the numbers from Table 2-3 and multiplying them by the matrix scan time.

Unnecessarily long settings of dwell and low burst frequencies should be avoided because these will also result in undesirably long failure detect times.

2.17.1 UL approval / VDE compliance

The QT2640 has been given a compliance test report by VDE and is approved by UL as a component suitable for use in both class B and class C safety critical products. By using this device and following the safety critical information throughout this datasheet, manufacturers can easily add a touch sense interface to their product, and be confident it can also readily pass UL or VDE testing.

2.18 VREF/WS pin

The VREF/WS pin is shared for multiple purposes: **VREF** definition, a **W**ake-up signal, and a **Sync** signal. Its fundamental purpose is to provide the reference voltage at the internal VREF node for the analogue-to-digital conversion of each sample. Secondary uses are as an input for a noise synchronisation signal, and as an input for wake-up from sleep mode.

During the conversion of each sample to digital form, the VREF/WS pin is coupled to the internal VREF node, and must be maintained at a stable voltage near Vss. If the sync and sleep features are not in use, VREF/WS can be connected directly to Vss.

The sync and wake signals are able to share a single pin together with VREF through the use of time division multiplexing because the sync and wake signals are not needed while the device is performing a matrix scan.

When MSYNC is enabled, a sync pulse triggers one full matrix scan, with a delay necessary between the end of each matrix scan and the sync pulse to trigger the next one. During the matrix scan, the VREF/WS pin is used to define VREF. Once the matrix scan is complete, the pin is available for use as the noise sync input.

If the sleep feature is used (see SDC), the wake-up signal must occur when the device is sleeping and not while it is performing a matrix scan.

A simple external circuit is required when either MSYNC or sleep are enabled. This circuit, consisting of two transistors connected in open-drain mode, with their drains both connected to VREF/WS operates together with a dynamic internal pull-up at the VREF/WS pin. See the schematic of Figure 1-1 on page 6 for an example circuit arrangement. One transistor couples the sync signal or the wake-up signal to VREF/WS, and the other is used to define VREF during each analogue-to-digital conversion, with the open-drain arrangement allowing the VREF definition to temporarily override the sync or wake-up signal. Most transistors are suitable for this purpose, including low-cost MOSFETs such as 2N7002 and BSS138.

If either MSYNC is enabled or if sleep is used, an internal pull-up, with value between 20 kΩ and 50 kΩ, is applied to the VREF/WS pin whenever the QT2640 is waiting for a sync signal or is in sleep mode. At all other times the VREF/WS pin is internally maintained near Vss, although this is not sufficient alone to guarantee the stable VREF definition needed during each conversion.

Wake-up from sleep and the sync feature are both triggered by a rising edge at the VREF/WS pin.

2.19 X7/SMP pin

The SMP function shares a pin with the X7 matrix drive line. Pin X7/SMP drives both functions but at different times, so there is no conflict between them. It generates the burst for X7 and, quite independently, it generates the SMP digital conversion signal for all samples on all X lines, despite the fact it shares a pin with X7.

This pin must be connected to both the X7 matrix line and the digital sample ramp resistors Rs0 – Rs7 (See Figure 1-1 on page 6). It can never be left unconnected. Even if X7 is not used, the X7/SMP pin must still be connected to the Rs resistors.

2.20 Frequency Hopping

The QT2640 supports frequency hopping to avoid a clash between the sampling frequency and noise at specific frequencies elsewhere in products or product-operating environments. It tries to hop away from the noise.

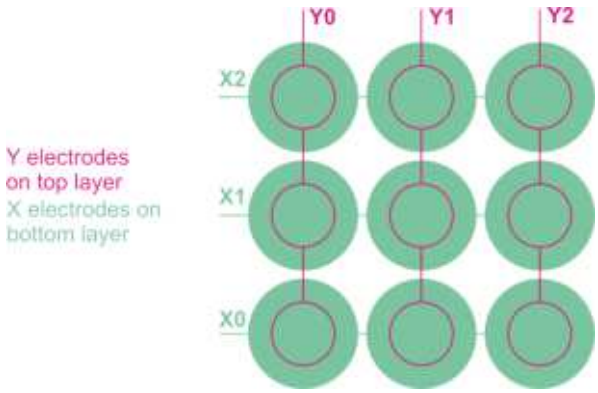
During the acquisition bursts, a sequence of pulses are emitted with a particular spacing, which equates to a particular sampling frequency. If the latter should coincide with significant noise generated elsewhere, touch sensing may be seriously impaired or false detections may occur.

To help combat such noise, the burst frequency can either be preset to one specific frequency (hopping disabled) away from the noisy frequency, or frequency hopping can be enabled and set to switch dynamically between three specific configured frequencies or even set to sweep a configured range of frequencies.

2.21 Key Design

Figure 2-7 shows part of a keypad laid out with a regular matrix of 3 x 3 keys using electrode patterns based on a flooded-X design.

Figure 2-7. Key Example Electrode Patterns



Flooded-X designs are one of the easiest to implement, produce excellent results with many overlay panels and can be considered first for all new QT2640 designs where a 2-layer PCB is intended for the touch interface.

For more information about the electrode design for sliders and wheels refer to the design guide listed in Appendix C. on page 69.

2.22 Sliders/Wheels

The QT2640 supports up to eight sliders and wheels in any combination, each constructed from a group of between two and eight consecutively numbered keys on the same Y line. A slider is an array of keys laid out to form a one dimensional track along the length of which a single touch position is reported with 8-bit resolution. A wheel, whilst also reporting a single touch with 8-bit resolution, is laid out as equal pieces of a pie to form a circle or wheel, typically with an insensitive void at the hub.

Keys that are used to form a slider/wheel cannot also be used as individual keys. The standard key processing must be disabled, by setting SLD, when the key is used in the construction of a slider or wheel.

At a minimum, setups SLD, SSN and SLEN must be configured to indicate which keys are used in sliders/wheels, where each slider/wheel starts in the logical key matrix, and how many keys have been used to construct each. In addition, SW is used to declare the type as either slider or wheel, position hysteresis can be programmed using SPH, a filter can be enabled with SF, and SRS can be used to adapt the reported position to fit the 8-bit full scale deflection.

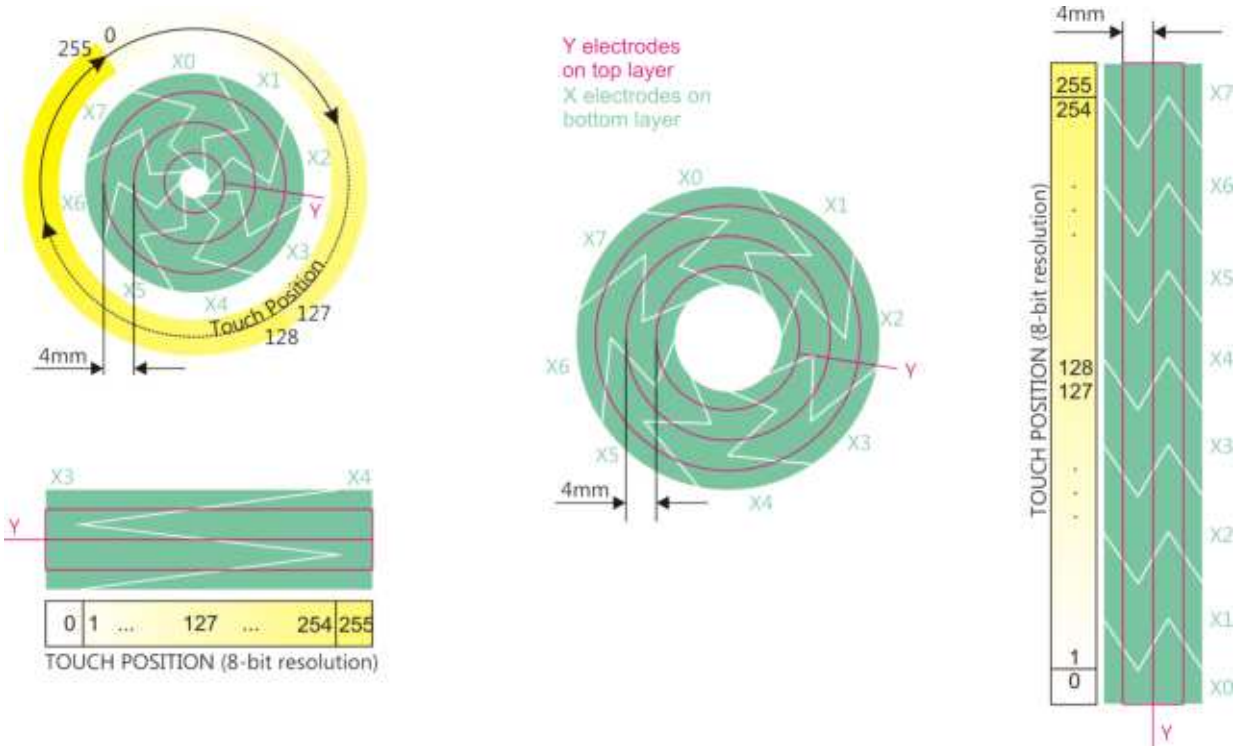
The group of keys used to form a slider/wheel may start almost anywhere within the logical matrix of keys provided the entire group is located on the same Y line. A group is not allowed to span across Y lines. For example, a slider formed from two keys can start at any X line from X0 to X6, but cannot start at X7, the last key on the Y line, whereas one formed from eight keys can only start at X0. All other keys on the same Y line as a slider/wheel but not used in the slider/wheel behave as normal. A single Y line can be used for more than one slider/wheel; Up to four sliders/wheels can be constructed on the same Y line.

The slider is physically constructed using a group of keys placed adjacent to each other. A minimum group size of two keys and a maximum group size of eight keys may be used, but the group must be laid out sequentially in numerical key order.

A slider can be constructed very simply by placing standard key patterns adjacent to each other, but other designs, for example ones based on flooded-X, can deliver better results. Short sliders can be constructed from just two keys, while better resolution will be achieved in longer sliders using a greater number of keys.

Figure 2-8 on page 21 shows some example slider and wheel electrode patterns based on flooded-X design. Two sliders are shown, a horizontal one based on just two keys, and a vertical one based on eight keys. The 2-key based horizontal slider uses keys from X3 and X4 arbitrarily as an example only.

Figure 2-8. Slider & Wheel Example Electrode Patterns



Flooded-X designs are one of the easiest to implement, for sliders and wheels as well as keys, produce excellent results with many overlay panels and can be considered first for all new QT2640 designs where a 2-layer PCB is intended for the touch interface.

For more information about the electrode design for sliders and wheels refer to the design guide listed in Appendix C. on page 69.

Setups NDIL, NTHR and NRD for a slider/wheel are all taken from the lowest numbered key in the slider group, but ignored at all other keys within the group. AKS is non-functional for sliders and must be disabled for all slider members. The fast detect integrator cannot be used within a slider and so FDIL must be set to 1 for all slider members.

3. Serial Communications

3.1 Introduction

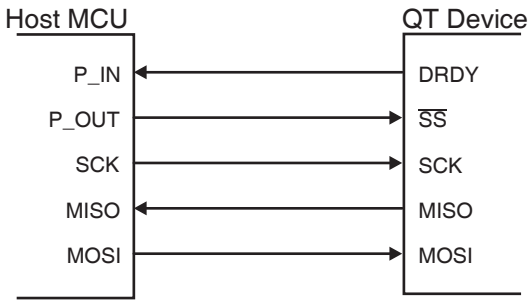
The QT2640 uses an SPI interface for communications with a host. The QT2640 always operates as a slave and must be driven from the host acting as SPI master.

The device also includes a Debug output interface, which can be used to monitor many operating variables during product development.

3.2 SPI Interface

The SPI host device always initiates communications sequences. This is intentional for FMEA and IEC/EN/UL60730 purposes so that the host always has total control over the communications with the QT2640. Even return data is controlled by the host. The QT2640 employs a CRC on return data to provide for robust communications.

Figure 3-1. Communications Signals – SPI



There is an essential $\overline{\text{DRDY}}$ line that handshakes transmissions. This is needed by the host from the QT2640 to ensure that transmissions are not sent when the QT2640 is busy or has not yet processed a prior transfer. If the host does not observe the correct $\overline{\text{DRDY}}$ timing, random communication errors may result.

Initiating or Resetting Communications: After a reset, or should communications be lost due to noise or out-of-sequence reception, the host should repeatedly wait for a period not less than the QT2640 communications time-out (20 ms ±5 ms). The host should then read location 0, followed by the CRC bytes, until the correct response is received back from location 0 and validated by the CRC. Location 0 should read as `1A` hex (26 decimal). The host can then resume normal run mode communications from a clean start.

Poll Rate: The typical poll rate in normal run operation should be no faster than once per 10 ms; 25 ms is more than fast enough to extract status data.

SPI communications operate only in slave mode, and obey $\overline{\text{DRDY}}$ control signaling. The clocking is as follows:

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

SPI requires five 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. MISO floats in three-state mode between bytes when $\overline{\text{SS}}$ is high, to facilitate multiple devices on one SPI bus.

SCK: SPI clock, input only clock from host. The host must shift out data on the falling SCK edge and the QT2640 clocks data in on the rising edge. The QT2640 likewise shifts data out on the falling edge of SCK back to the host so that the host can shift the data in on the rising edge.

Note: SCK must idle high; it should never float.

$\overline{SS}$: Slave select, input only. Acts as a framing signal to the sensor from the host. $\overline{SS}$ must be low before and during each byte transfer with the host. It must not go high again until the SCK line has returned high; $\overline{SS}$ must idle high. This pin includes an internal pull-up resistor of 20 kΩ – 50 kΩ. When $\overline{SS}$ is high, MISO floats.

$\overline{DRDY}$: Data Ready, active-high, indicates to the host that the QT2640 is ready to send or receive data. This pin idles high and is an open-drain output with an internal 20 kΩ – 50 kΩ pull-up resistor. Most communications failures are the result of failure to properly observe the $\overline{DRDY}$ timing.

Serial communications pacing is controlled by $\overline{DRDY}$. Use of $\overline{DRDY}$ is critical to successful communications with the QT2640. The host is permitted to perform an SPI transfer only when $\overline{DRDY}$ has returned high. After each SPI byte transfer $\overline{DRDY}$ goes low after a short delay and remains low until the QT2640 is ready for another transfer. A short delay occurs before $\overline{DRDY}$ is driven low because the QT2640 may be otherwise busy and requires a finite time to respond. $\overline{DRDY}$ may go low only for a few microseconds. During the period from the end of one transfer until $\overline{DRDY}$ goes low and back high again, the host should not perform another transfer. Therefore, before each byte transmission, including the first byte of each sequence, the host should first check that $\overline{DRDY}$ is high again.

If the host wants to perform a byte transfer with the QT2640 it should behave as follows:

1. Wait at least 100 µs after the previous SPI transfer (time S5 in Figure 3-2 on page 24: $\overline{DRDY}$ is guaranteed to go low before this 100 µs expires).
2. Wait until $\overline{DRDY}$ is high (it may already be high again).
3. Perform the next SPI transfer with the QT2640.

The time it takes for $\overline{DRDY}$ to go high again after each transfer depends if the host is performing

A setups write, or is performing a read, as follows:

Setups write: ≤ 20 ms

Read: ≤ 1 ms

The $\overline{DRDY}$ times above are valid when the maximum operating frequency (FREQ0 = 1) is used. These times increase as the operating frequency is reduced. With very low operating frequency add 5 ms to the above times. With the Debug interface enabled, add 11ms to the above times.

Other $\overline{DRDY}$ specifications:

Min time $\overline{DRDY}$ is low: 1 µs

Min time $\overline{DRDY}$ is low after reset: 80 ms

Null Bytes: When the QT2640 responds with data requested in a read operation, the host should issue null bytes (0x00) in order to recover the response bytes back. The host should not start a new communications sequence until all the response and CRC bytes are accepted back from the QT2640.

Timeout: A successful communications sequence consists of a number of byte transfers. The QT2640 expects each byte transfer within a sequence to occur within 20 ms (± 5 ms) of the previous transfer. If more than 20 ms elapses between any two bytes, the QT2640 abandons the current sequence and starts a new sequence at the next byte transfer.

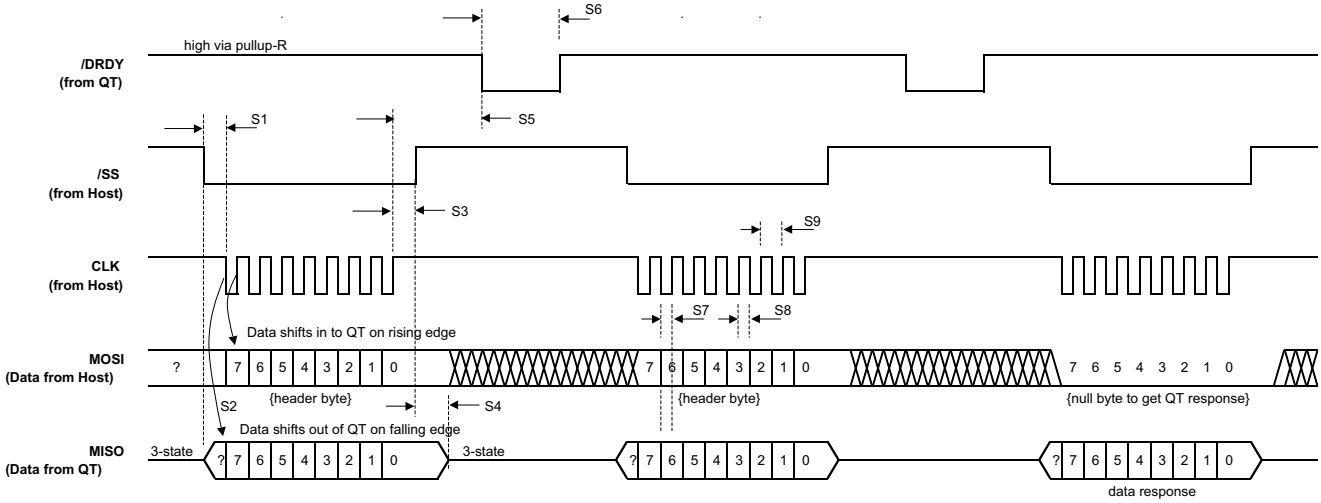
Wake-up: The QT2640 can be configured to automatically sleep, but the host must awaken the QT2640, when required, with a rising signal edge at the VREF/WS pin, which should be accomplished through a simple transistor as described in Section 2.18 on page 18.

With the $\overline{SS}$ line used to drive this transistor, the host can simply pulse $\overline{SS}$ to wake the QT2640. The host should not send an actual SPI byte to prevent the QT2640 from seeing a byte it cannot properly interpret due to timing errors during wake-up. There is an interval of approximately 1.5 ms from the pulse on VREF/WS before the QT2640 is able to resume processing. Transmissions to the QT2640 within this interval are discarded.

SPI Line Noise: In some designs it is necessary to run SPI lines over ribbon cable across a lengthy distance on a PCB. This can introduce ringing, ground bounce, and other noise problems which can introduce false SPI clocking or false data. Simple RC networks and slower data rates are helpful to resolve these issues.

A CRC check appends all data responses in order to detect transmission errors to a high level of certainty.

Figure 3-2. SPI Slave Mode Timing



S1 > 125 ns	S2 < 20 ns	S3 > 25 ns	S4 < 20 ns	
S5 < 100 µs	S6 > 1 µs	S7 >125 ns	S8 > 125 ns	S9 > 250 ns
Note: See "Timing Specifications" on page 60 for more detailed definitions.				

3.3 Writing Data to the Device

The sequence of events required to write data to the device is shown below:

Byte	1		2		3		4	
MOSI	Mem.Addr.bits 7..0	D	bit 7: 0 = write bit 6: n, bit 8 bits 5..3: undefined bits 2..0: Mem.Addr.bits 10..8	D	n. bits 7..0	D	Data	D
MISO	U		U		U		U	

Table 3-1. Key to Write Sequence

Byte #	Symbol	Description
1, 2	Mem.Addr.	Target memory address within QT2640. This is an 11-bit address formed with bits from both Byte 1 and Byte 2. The 3 most significant bits are from Byte 2, bits 10..8, and the 8 least significant bits are from Byte 1.
	U	Undefined data byte, should be ignored.
	D	Delay 100 µs then wait until $\overline{\text{DRDY}}$ high.
2	write	Set bit 7 to zero, indicating write operation to the QT2640.
2, 3	n	# bytes to write, a 9-bit value with the most significant bit located in Byte 2, bit 6.
4+	Data	Data byte(s) to write to QT2640.

The host initiates a write sequence by sending a sequence of three header bytes followed by n data bytes. The three header bytes define the internal memory address to be written to, the flag to indicate a write operation (0), and n, the number of QT2640 addresses to be written. See Table 3-1. After n data bytes have been written, the QT2640 automatically terminates the write operation and will start a new SPI sequence with the next byte transfer.

Between each byte transfer, the host must follow the $\overline{\text{DRDY}}$ handshake procedure (wait 100 µs and then wait until $\overline{\text{DRDY}}$ is high). If the host sends more than one data byte, they are written to consecutive memory addresses, the device automatically increments the target memory address after writing each data byte.

The host should not try to write beyond the last setups address.

The raw SPI protocol defines simultaneous bidirectional byte transfers. For each byte sent from the host, another byte is received back from the slave. During a write sequence, the bytes returned by the QT2640 are undefined and should be ignored.

3.4 Reading Data From the Device

The sequence of events required to read data from the device is shown below:

Byte	1		2		
MOSI	Mem.Addr.bits 7..0	D	bit 7: 1 = read bit 6: n, bit 8 bits 5..3: undefined bits 2..0: Mem.Addr.bits 10..8	D	…
MISO	U		U		

	3		4											
…	n. bits 7..0	D	NULL	D	NULL	D	…	NULL	D	NULL	D	NULL	D	
	U		Data 1		Data 2		…	Data n		CRC LSB		CRC MSB		

Table 3-2. Key to Read Sequence

Byte #	Symbol	Description
1, 2	Mem.Addr.	Target memory address within QT2640. This is an 11-bit address formed with bits from both Byte 1 and Byte 2. The 3 most significant bits are from Byte 2, bits 10..8, and the 8 least significant bits are from Byte 1.
	U	Undefined data byte, should be ignored.
	D	Delay 100 µs then wait until $\overline{\text{DRDY}}$ high.
2	read	Set bit 7 to 1, indicating a read operation.
2, 3	n	# bytes to read, a 9-bit value with the most significant bit located in Byte 2.
4+	Data	Data byte(s) read from QT2640.

The host initiates a read sequence by sending a sequence of three header bytes followed by n NULL bytes to recover the n data bytes from the QT2640. The three header bytes indicate the internal memory address to be read, the flag to indicate a read operation(1), and n, the number of QT2640 addresses to be read. See Table 3-1.

When all n data bytes have been returned, the device returns a 16-bit CRC, LSB first. The device calculates the double-word (16-bit) CRC using the three header bytes and the n data bytes themselves, all in the same sequence they occur during the transmission (see Appendix A. on page 64).

Between each byte transfer, the host must follow the $\overline{\text{DRDY}}$ handshake procedure (wait 100 µs and then wait until $\overline{\text{DRDY}}$ is high). If the host reads more than one data byte, they are read from consecutive memory addresses, the device automatically increments the target memory address after returning each data byte.

The host should not try to read beyond the last setups address.

The raw SPI protocol defines simultaneous bidirectional byte transfers. For each byte sent from the host, another byte is received back from the slave. During a read sequence, the bytes transmitted by the QT2640 while the host is sending the three header bytes are undefined and should be ignored.

3.5 $\overline{\text{CHANGE}}$ Pin

The $\overline{\text{CHANGE}}$ pin can be used to alert the host to key touches, key releases, and changes in slider position, thus reducing the need for unnecessary communications. Normally, the host can simply not bother to communicate with the device, except when the $\overline{\text{CHANGE}}$ pin becomes active.

$\overline{\text{CHANGE}}$ becomes active after reset and when there is a change in key/slider state (either touch or touch release), or a change in slider position, and becomes inactive again only when the host performs a read from address 6, the detect status register for all keys on Y0. $\overline{\text{CHANGE}}$ does not self-clear; only an SPI read from location 6, or a device reset, clears it.

It is important to read all eight key state addresses, and the slider detect status to ensure the host has a complete picture of which keys and sliders have changed.

After the device is reset it performs internal initialisation and then sets $\overline{\text{CHANGE}}$ active (low) to signal the host that it is ready to communicate.

$\overline{\text{CHANGE}}$ is an open-drain output with an internal 20 kΩ–50 kΩ pull-up resistor. This allows multiple devices to be connected together in a single wire-OR logic connection with the host. When the $\overline{\text{CHANGE}}$ pin goes active, the host can poll all devices to identify which one is reporting a touch change.

IEC/EN/UL60730 compliant products cannot rely on the $\overline{\text{CHANGE}}$ pin because its operation cannot be verified. The $\overline{\text{CHANGE}}$ pin can still be utilized but only to optimize the key response time. The host must also poll the QT2640 Detect Status bytes (Addresses 6 – 13), but at a rate suitable to guarantee IEC/EN/UL60730 compliance. A poll rate of once every 100 ms would impose very little extra load on the QT2640.

The DBG_DATA and the $\overline{\text{CHANGE}}$ output share an I/O pin, and can interfere with each other. DEBUG is intended as an aid only during development, and should be disabled for production.

3.6 Debug Output Interface

The QT2640 includes a debug interface which may be used for observing many internal operating variables, in real time, even while the part is actively communicating with a host over the SPI serial interface. The Debug interface provides a useful aid during product development (see Section B. on page 65).

4. Memory Map

Addresses 0 – 767 are read-only and allow direct access to operating data within the QT2640. Address 768 allows for control commands to be written to the device. Addresses 769 - 1247 allow the device configuration to be read and changed.

Table 4-1. Internal Register Address Allocation

Address	Use	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Device ID	Device ID = 1A (26)							
1	Part Rev.	Part Revision							
2	100 ms counter	(IEC/EN/UL60730)							
3	Signal Fail counter	(IEC/EN/UL60730)							
4	Matrix Scan counter	(IEC/EN/UL60730)							
5	Device Status		Cal Error	FMEA Fail	Host CRC Mismatch	Mains Sync Fail	LSL Fail	Calibrating	
6	Detect Status Y0	Key 7	Key 6	Key 5	Key 4	Key 3	Key 2	Key 1	Key 0
7	Detect Status Y1	Key 15	Key 14	Key 13	Key 12	Key 11	Key 10	Key 9	Key 8
8	Detect Status Y2	Key 23	Key 22	Key 21	Key 20	Key 19	Key 18	Key 17	Key 16
9	Detect Status Y3	Key 31	Key 30	Key 29	Key 28	Key 27	Key 26	Key 25	Key 24
10	Detect Status Y4	Key 39	Key 38	Key 37	Key 36	Key 35	Key 34	Key 33	Key 32
11	Detect Status Y5	Key 47	Key 46	Key 45	Key 44	Key 43	Key 42	Key 41	Key 40
12	Detect Status Y6	Key 55	Key 54	Key 53	Key 52	Key 51	Key 50	Key 49	Key 48
13	Detect Status Y7	Key 63	Key 62	Key 61	Key 60	Key 59	Key 58	Key 57	Key 56
14	Slider Detect Status	Slider 7	Slider 6	Slider 5	Slider 4	Slider 3	Slider 2	Slider 1	Slider 0
15	Slider 0 Position	Range 0 - 255							
16	Slider 1 Position	Range 0 - 255							
17	Slider 2 Position	Range 0 - 255							
18	Slider 3 Position	Range 0 - 255							
19	Slider 4 Position	Range 0 - 255							
20	Slider 5 Position	Range 0 - 255							
21	Slider 6 Position	Range 0 - 255							
22	Slider 7 Position	Range 0 - 255							
23	Frequency Hop Mode	Frequency Hop Mode							
24	Current Frequency	Current Frequency							
25	Current pulse spacing	Current pulse spacing							
26	Key 0 – Signal	Signal (bits 7 .. 0)							
27	Key 0 – Cal.State / Signal	Calibration State			Signal (bits 12... 8)				
28	Key 0 – Reference	Reference (bits 7 .. 0)							

Table 4-1. Internal Register Address Allocation (Continued)

Address	Use	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
29	Key 0 – Status / Reference	FMEA fail	Detect	LSL fail	Reference (bits 12 .. 8)				
30	Key 0 – FDI count / DI count	FDI count				DI count			
31	Key 0 – Detect Timeout, negative	Detect Timeout, negative							
32	Key 0 – Detect Timeout, positive	Detect Timeout, positive							
33	Key 0 – Drift counter	Drift counter							
34 – 41	key 8	Data set for key. The data sets are the same for all keys. Refer to the data set for key (addresses 26 to 33) for details.							
42 – 49	key 16	Data set for key							
50 – 57	key 24	Data set for key							
58 – 65	key 32	Data set for key							
66 – 73	key 40	Data set for key							
74 – 81	key 48	Data set for key							
82 – 89	key 56	Data set for key							
90 – 97	key 1	Data set for key							
98 – 105	key 9	Data set for key							
106 – 113	key 17	Data set for key							
114 – 121	key 25	Data set for key							
122 – 129	key 33	Data set for key							
130 – 137	key 41	Data set for key							
138 – 145	key 49	Data set for key							
146 – 153	key 57	Data set for key							
154 – 161	key 2	Data set for key							
162 – 169	key 10	Data set for key							
170 – 177	key 18	Data set for key							
178 – 185	key 26	Data set for key							
186 – 193	key 34	Data set for key							
194 – 201	key 42	Data set for key							
202 – 209	key 50	Data set for key							
210 – 217	key 58	Data set for key							
218 – 225	key 3	Data set for key							
226 – 233	key 11	Data set for key							
234 – 241	key 19	Data set for key							
242 – 249	key 27	Data set for key							

Table 4-1. Internal Register Address Allocation (Continued)

Address	Use	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
250 – 257	key 35	Data set for key							
258 – 265	key 43	Data set for key							
266 – 273	key 51	Data set for key							
274 – 281	key 59	Data set for key							
282 – 289	key 4	Data set for key. The data sets are the same for all keys. Refer to the data set for key (addresses 26 to 33) for details.							
290 – 297	key 12	Data set for key							
298 – 305	key 20	Data set for key. The data sets are the same for all keys. Refer to the data set for key (addresses 26 to 33) for details.							
306 – 313	key 28	Data set for key							
314 – 321	key 36	Data set for key							
322 – 329	key 44	Data set for key							
330 – 337	key 52	Data set for key							
338 – 345	key 60	Data set for key							
346 – 353	key 5	Data set for key							
354 – 361	key 13	Data set for key							
362 – 369	key 21	Data set for key							
370 – 377	key 29	Data set for key							
378 – 385	key 37	Data set for key							
386 – 393	key 45	Data set for key							
394 – 401	key 53	Data set for key							
402 – 449	key 61	Data set for key							
410 – 417	key 6	Data set for key							
418 – 425	key 14	Data set for key							
426 – 433	key 22	Data set for key							
434 – 441	key 30	Data set for key							
442 – 449	key 38	Data set for key							
450 – 457	key 46	Data set for key							
458 – 465	key 54	Data set for key							
466 – 473	key 62	Data set for key							
474 – 481	key 7	Data set for key							
482 – 489	key 15	Data set for key							
490 – 497	key 23	Data set for key							
498 – 505	key 31	Data set for key							

Table 4-1. Internal Register Address Allocation (Continued)

Address	Use	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
506 – 513	key 39	Data set for key							
514 – 521	key 47	Data set for key							
522 – 529	key 55	Data set for key							
530 – 537	key 63	Data set for key. The data sets are the same for all keys. Refer to the data set for key (addresses 26 to 33) for details.							
538 – 767	Reserved	Reserved							
768	Command Address	Control Command							
769 – 1247		Setups, see Section 5. on page 37							

4.1 Address 0 – Device ID

Table 4-2. Device ID

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Device ID							

Device ID: Holds the device ID of 0x1A.

4.2 Address 1 – Part Rev

Table 4-3. Firmware Version

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Version							

Version: Holds the firmware version (for example 1).

4.3 Addresses 2 to 4 – Report IEC/EN/UL60730 Counters

These counters can be used by the host to check the correct speed and operation of the device. The host must check these values regularly to meet the requirements of IEC/EN/UL60730, which requires that each component of a system be checked for correct operation. Where correct speed of operation must be confirmed and the device has no way to perform such a cross-check internally, counters are exposed through the communication interface to enable independent cross checking by the host.

Address 2: 100 ms counter (IEC/EN/UL60730)

This is an 8-bit unsigned counter that is incremented once every 100 ms, counting 256 steps repeatedly from 0 to 255. When the counter has reached 255 it wraps back to 0 at the next 100 ms interval. The counter should take between 25 and 26 s (256 × 100 ms = 25.6 s) to count up from zero through 255 and wrap back to zero again. The host must read this counter regularly and cross-check the counting rate against one of its own clock sources.

If the 100 ms counter is read once every second, for example, the host should find the counter has increased by 10 counts from the value returned at each previous read and should traverse one full count range (256 steps) when the

host has read the counter 25 or 26 times. The host should verify the 100 ms counter is incrementing at the expected rate. If the counter advances faster or slower than expected, there could be a fault with the QT2640 or the host, and the host should adopt an appropriate strategy to meet the required safety standard.

Address 3: Signal fail counter (IEC/EN/UL60730)

This is an 8-bit unsigned counter that is incremented each time a signal capture failure occurs. Signal capture failure can occur where keys are enabled but do not physically exist. Only keys that exist should be enabled; All other keys should be disabled.

Signal capture failure can also occur where heavy noise spikes corrupt the signal; Occasional capture failure is to be expected and does not unduly affect the device performance. Regular capture failure would extend the key response time. The host must check this counter regularly. Tests should be made with a heavy noise source during development to determine how the key response time is affect and determine a maximum acceptable count rate for this counter.

Address 4: Matrix scan counter (IEC/EN/UL60730)

This is an 8-bit counter that is incremented before the start of each matrix scan, or keyscan cycle, counting 256 steps repeatedly from 0 to 255. When the counter has reached 255 it wraps back to 0 at the start of the next keyscan cycle. The keyscan cycle time should be measured with an oscilloscope during development. The Matrix Scan count rate can be calculated directly from this.

For example, if the keyscan cycle time is measured as 10 ms, the counter counts 256 steps in 2560 ms (256 × 10 ms). The host must read this counter regularly to check the matrix scan is operating at the expected rate. If the Matrix Scan counter is read once every 100 ms, for example, the host should find the counter has increased by 10 counts from the value returned at each previous read and should traverse one full count range (256 steps) when the host has read the counter 25 or 26 times. The host should verify the counter is incrementing at the expected rate. If the counter advances faster or slower than expected, there could be a fault with the QT2640 or the host, and the host should adopt an appropriate strategy to meet the required safety standard.

4.4 Address 5 – Report Device Status

Address 5: Device status

This byte contains the general status bits. The bits report as follows:

Bit	Description
7	Reserved
6	1 = Cal Error
5	1 = FMEA failure detected
4	1 = Internally computed setups CRC does not match HCRC
3	1 = Mains sync error
2	1 = LSL failure detected
1	1 = any key in calibration
0	Reserved

Bit 6: Set if a calibration error occurs.

Bit 5: Set if an FMEA error was detected. See Section 2.16 on page 15.

Bit 4: Set if the setups CRC does not match the CRC uploaded by the host (HCRC). The CRC is computed repeatedly and checked against the value uploaded by the host. This bit is set if the two values do not match.

Bit 3: Set if there was a mains sync error, for example there was no Sync signal detected within the allotted 100 ms. See Section 5.7.2 on page 44. This condition is not necessarily fatal to operation, however the device operates very slowly and may suffer from noise problems if the sync feature was required for noise reasons. Reset the device to clear this bit after the MSYNC setup has been cleared (OFF).

Bit 2: Reports that an enabled key has a very low reference value, lower than the user-configurable LSL value (see Section 5.10.3 on page 47).

Bit 1: Set if any key is in the process of calibrating.

A host in an IEC/EN/UL60730-compliant product must check bits 1, 2, 4, and 5 and handle persistent errors appropriately to maintain safety.

4.5 Addresses 6 to 13 – Detect Status Y0 – Y7

Table 4-4. Key Status

Address		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
6	Detect Status Y0	Key 7	Key 6	Key 5	Key 4	Key 3	Key 2	Key 1	Key 0
7	Detect Status Y1	Key 15	Key 14	Key 13	Key 12	Key 11	Key 10	Key 9	Key 8
8	Detect Status Y2	Key 23	Key 22	Key 21	Key 20	Key 19	Key 18	Key 17	Key 16
9	Detect Status Y3	Key 31	Key 30	Key 29	Key 28	Key 27	Key 26	Key 25	Key 24
10	Detect Status Y4	Key 39	Key 38	Key 37	Key 36	Key 35	Key 34	Key 33	Key 32
11	Detect Status Y5	Key 47	Key 46	Key 45	Key 44	Key 43	Key 42	Key 41	Key 40
12	Detect Status Y6	Key 55	Key 54	Key 53	Key 52	Key 51	Key 50	Key 49	Key 48
13	Detect Status Y7	Key 63	Key 62	Key 61	Key 60	Key 59	Key 58	Key 57	Key 56

Detect status for keys, one bit per key – see Table 4-1 on page 28.

These bits indicate which keys are in detect, if any. Touched keys report as 1, untouched or disabled keys report as 0 (some keys may continue to report detect after setups change and before the device is reset). The first key in a slider/wheel group will report as 1 when the slider is touched, all other keys in the group report as 0. A change in these bytes will cause the $\overline{\text{CHANGE}}$ line to be asserted (low). The $\overline{\text{CHANGE}}$ line becomes inactive on reading address 6.

4.6 Address 14 – Slider Detect Status

Table 4-5. Slider Detect Status

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
14	Slider 7	Slider 6	Slider 5	Slider 4	Slider 3	Slider 2	Slider 1	Slider 0

One bit for each of the eight sliders/wheels. These bits indicate which sliders/wheels are in detect, if any. Those in detect report as 1, otherwise 0. A change in these bytes will cause the $\overline{\text{CHANGE}}$ line to be asserted (low). The $\overline{\text{CHANGE}}$ pin becomes inactive on reading address 6, so its important to read addresses 6 through 22 in one operation to ensure a valid set of data is collected and no changes are missed.

The host should use these detect flags to determine the sliders in detect, and should not try to use the key detect flags reported in bytes 6 to 13.

4.7 Addresses 15 to 22 – Slider Positions

Table 4-6. Slider Touch Positions

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
15	Slider 0							
16	Slider 1							
17	Slider 2							
18	Slider 3							
19	Slider 4							
20	Slider 5							
21	Slider 6							
22	Slider 7							

Each address reports the touch position for one slider/wheel, to 8-bit resolution, as an unsigned integer. Each address is updated only when the corresponding slider/wheel is touched and the relevant bit in the Slider Detect Status byte is set. After touch is removed from a slider/wheel, the value reported will remain at the last known touch position.

A change in this value will cause the $\overline{\text{CHANGE}}$ line to be asserted. The $\overline{\text{CHANGE}}$ pin becomes inactive on reading address 6, so its important to read addresses 6 through 22 in one operation to ensure a valid set of data is collected and no changes are missed.

4.8 Address 23 – Frequency Hop Mode

The value in this address is a copy of the FHM Setup.

4.9 Address 24 – Current Frequency

When FHM = 1 or FHM = 2, the frequency hopping module switches between three configurable frequencies. This byte indicates which of the three frequency selections is currently in use.

Table 4-7. Current Frequency

Value	Description
0	Configured frequency 0 (FREQ0) is in use
1	Configured frequency 0 (FREQ1) is in use
2	Configured frequency 0 (FREQ2) is in use

4.10 Address 25 – Current Pulse Spacing

The burst sampling frequency is changed by using different idle times, or pulse spacing, between consecutive pulses in the burst. This byte reflects that idle time, with each increment representing an increase of approximately 188ns.

4.11 Addresses 26 to 33 – Data Set for Key 0

These eight read-only addresses allow access to the internal raw data for Key 0. The data read from these addresses is dynamically updated as new samples are obtained from the matrix and processed.

Table 4-8. Format of Data Set for One Key

Offset Within Data Set	Description
1 – 0	Bits 12 – 0: Signal, holds the latest sample. Bits 15 – 13: Calibration State: 0 = Pending. Calibration has been scheduled for this key, and will start shortly. 1 – 4 = The key is in the process of being calibrated 5 = Success. The key was calibrated successfully. 6 = Failed, the reference could not be confirmed as converging with the true signal average.
3 – 2	Bits 12 – 0: Reference Bit 13: 1= LSL fail. Reference < LSL (Low Signal Level) Bit 14: 1= Detect. The key is in detect. Bit 15: 1= FMEA failure.
4	Bits 7 – 4: Fast DI Bits 3 – 0: Normal DI
5	Negative Detect Timeout. Time remaining before the key is recalibrated. Each count = 500 ms
6	Positive Detect Timeout. Time remaining before the key is recalibrated. Each count = 100 ms
7	Drift compensation counter, range –127 to +127. Each count = 100 ms. Recalibration occurs when the drift counter reaches the value set by NDRIFT (negative count) or PDRIFT (positive count)

4.12 Addresses 34 to 537 – Data for Keys 1 – 63

These read-only addresses allow access to the internal raw data for keys 1 – 63. The range is divided equally between the keys, with eight bytes allocated to each and formatted the same as for key 0. Refer to Table 4-8 for details of the use of each address.

The data read from these addresses is dynamically updated as new samples are obtained from the matrix and processed.

4.13 Address 768 – Command Address

The Command Address is a write-only location. Reading from this address will not cause any side-effects but returns undefined values. The value written defines the action taken.

(0xFF) Calibrate All:

Shortly after the QT2640 receives a value of 0xFF at the Command Address it recalibrates all keys, sliders and wheels, and restarts operation.

The host can monitor the progress of the calibration by checking the device status byte, and the data set for each key.

(0xFE) Setups Write-Enable:

Writing a value of 0xFE to the Command Address write-enables the setups block of the device. Normally the setups are write-protected with the protection being engaged as soon as a read operation is performed at any address. By writing a value of 0xFE to the Command Address, the write-protection is disengaged. The Command Address is located conveniently immediately before the setups so that the write protection may be disengaged and the setups written in a single communication sequence.

(0xFD) Low Level Cal and Offset:

Shortly after receiving this command the QT2640 performs a calibration and offset procedure across all keys. If a previous command 0xFD is still being processed, the new request will be ignored.

The low level cal and offset procedure is intended as a one-time factory event that can take up to 16 seconds to complete. The host can monitor the progress of the calibration by checking the QT2640 Device Status. The calibration bit will be set throughout the process.

The procedure involves the device calibrating each key in turn at each of the operating frequencies selected with FREQ0, FREQ1 and FREQ2, calculating the difference between the signals at those frequencies and storing the results as offsets into CFO_1 and CFO_2 for each key. When the procedure is complete, the host can read back the setups and record CFO_1 and CFO_2 into its own copy of the setups block. The QT2640 does not change the Host CRC, so there will be a mismatch in the Host CRC after this command completes. The onus is on the host to compute the CRC and upload a definitive setups block to the QT2640.

(0xFC) Request Frequency Hop:

This command requests a frequency hop, and is effective only when FHM = 1 or FHM = 2. The QT2640 uses a weighting system to favour the frequency exhibiting least signal noise, and the frequency hop may not therefore occur for up to 255 matrix scans after the hop is requested when the favoritism for the current frequency has run out. The frequency hop request is cancelled if calibration is active or pending for any key before the hop occurs.

(0xFB) Sleep Request:

Upon receiving this command the QT2640 sets an internal flag to indicate that low power sleep mode has been requested. The flag does not force the device to sleep immediately but requests the QT2640 to sleep at the end of each matrix scan. Sleep must also be enabled in setups (see Section 5.7.3 on page 45), otherwise this command has no effect.

The internal flag is cleared upon receiving the first byte of a new SPI sequence. Subsequently, the device will not sleep again until another Sleep Request command is received.

The QT2640 must be awake to receive an SPI sequence. To clear the internal sleep request flag, issue a wake-up signal at the VREF/WS pin and then issue an SPI sequence (other than a Sleep Request command) within one matrix scan cycle; The QT2640 is awake for a minimum of one matrix scan cycle time after a wake-up signal at the VREF/WS pin. The matrix scan cycle time is dependent on a number of setups parameters and should be measured using an oscilloscope.

See Section 2.15 on page 15 for details of the sleep behavior. See also Section 2.18 on page 18 for details of the VREF/WS pin.

(0x40) Force Reset:

Within 20ms of receiving this command, the device performs a reset. After any reset, the device automatically performs a full key calibration on all keys and slider/wheel members.

(*k*) Calibrate Key:

Writing the key number *k* (range 0 – 63) as the command, requests the QT2640 recalibrate key *k*. The operation is the same as if command 0xFF were written except only one key is affected. 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 allows the host to try to recover a single key that has failed calibration or is not calibrated correctly.

5. Setups

The QT2640 calibrates and processes all signals using a number of algorithms specifically designed to provide for high survivability in the face of adverse environmental challenges. It provides 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 the algorithm to suit each application. The setups are loaded into the QT2640 over the serial interface and stored in an onboard EEPROM array. The setups can only be loaded when write-enabled (see Section 4.13 on page 35).

After a setups block load, the QT2640 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. This reset can be either a hardware or software reset.

Many setups employ lookup-table (LUT) value translation. The setups Block Summary tables show all translation values. Refer to Section 5.23 on page 56 for a list of all setups.

Block length issues: The setups block is 479 bytes long (including the two CRC bytes) to accommodate 64 keys and eight sliders/wheels. This can be a burden on smaller host controllers with limited memory. In larger quantities the QT2640 can be procured with the setups block preprogrammed from Atmel. 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.

Default values shown are factory defaults.

5.1 Addresses 769 to 832 – Thresholds

Table 5-1. Thresholds

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
769	PTHR (Key 0)				NTHR (Key 0)			
…	…				…			
832	PTHR (Key 63)				NTHR (Key 63)			

5.1.1 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 is also 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' (signal swing from touch), thus requiring smaller NTHR values to detect touch. Use the largest possible value for NTHR for best noise immunity.

The negative threshold is programmed on a per-key basis using the Setup process. See Table 5-24 on page 56 and also Section 5.7.1 on page 44 (Threshold Multiplier – THRM). The negative threshold for a slider/wheel is taken from NTHR at the lowest numbered key in the slider group, but ignored at all other keys within the group.

NTHR Default value: 6 (18 counts of threshold)

NTHR Range: 0 to 15 (6 to 36 counts of threshold)

5.1.2 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 hysteresis: PHYST is fixed at 12.5% of the positive threshold value and cannot be altered.

Positive threshold levels are programmed using the Setup process on a per-key basis. See also Section 5.7.1 on page 44 (Threshold Multiplier – THRM)

Default value: 6 (18 counts of threshold)

Range: 0 to 15 (6 to 36 counts of threshold)

5.2 Addresses 833 to 896 – Drift Compensation – NDRIFT, PDRIFT

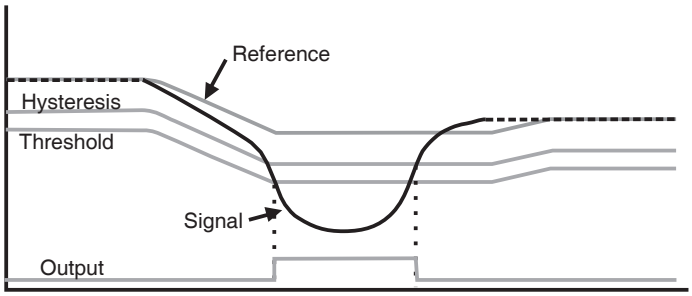
Table 5-2. Drift Compensation

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
833	PDRIFT (Key 0)				NDRIFT (Key 0)			
…	…				…			
896	PDRIFT (Key 63)				NDRIFT (Key 63)			

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

Drift compensation (Figure 5-1 on page 38) 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 QT2640 drift compensates using a slew-rate limited change to the reference level; the threshold and hysteresis values are slaved to this reference.

Figure 5-1. Thresholds and Drift Compensation



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) causes 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 sense pad, for which the sensor has already made full allowance, could suddenly be removed leaving the sensor with an artificially suppressed reference level and thus become insensitive to touch. In the latter case, the sensor should compensate for the object's removal by raising the reference level relatively quickly.

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 and PDRIFT are effective while the part is awake. If sleep is enabled, SDC must also be configured so drift compensation operates at the desired rate. NDRIFT and PDRIFT are both configured on a per-key basis.

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

NDRIFT Range: 0 to 15
(0.1 to 10 s per count of drift compensation, page 56)

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

PDRIFT Range: 0 to 15
(0.1 to 10 s per count of drift compensation, page 56)

5.3 Addresses 897 to 960 – Detect Integrator Limits – NDIL, FDIL

Table 5-3. Detect Integrator Limits

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
897	FDIL (Key 0)				NDIL (Key 0)			
…	…				…			
960	FDIL (Key 63)				NDIL (Key 63)			

The NDIL parameter is used to enable and disable keys in the matrix and to provide signal filtering. To enable a key, its NDIL parameter should be nonzero (NDIL = 0 disables a key).

To suppress false detections caused by spurious events like electrical noise, the device incorporates a 'detection integrator' or DI counter mechanism. A per-key counter is incremented for each sample where the key has exceeded its threshold and is decremented otherwise. When this counter reaches a preset limit the key is finally declared to be touched.

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. When fast-burst is entered the QT device locks onto the key and repeats the acquire burst rapidly until the fast-DI counter reaches FDIL or drops back to zero. 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 on a per key basis via the setups process. The total number of required confirmations is equal to FDIL times NDIL.

- If FDIL = 6 and NDIL = 2, the total detection confirmations required is 12, even though the QT2640 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 QT2640 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 ≥ 2, then the fast-DI counter also operates in addition to the NDIL counter.
- If Signal < NTHR: The fast-DI counter is incremented towards FDIL due to touch.
- If Signal > NTHR then the fast-DI counter is decremented due to lack of touch.

Disabling a key: If NDIL = 0, the key becomes disabled. Keys disabled in this way are pared from the burst sequence in order to improve sampling rates and thus response time. See Section 2. on page 7.

Note: It is very important to disable keys that do not physically exist in the layout, otherwise real keys can incorrectly report their signal as zero.

This function is programmed on a per-key basis. Do not use FDIL = 0 because it is invalid. FDIL must be set to one (FDIL=1) for all keys used to create a slider or wheel.

The DI limit for a slider/wheel is taken from NDIL at the lowest numbered key in the slider group, but ignored at all other keys within the group.

NDIL Default value: 2

NDIL Range: 0 (Off) – 15

FDIL Default value: 5

FDIL Range: 1 – 15 (0 is illegal, do not use)

5.4 Addresses 961 to 1024 – Negative Recal Delay – NRD

Table 5-4. Negative Recal Delay

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
961	NRD (Key 0)							
…	…							
1024	NRD (Key 63)							

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 is automatically recalibrated. After a recalibration has taken place, the affected key once again functions 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 never autorecalibrates 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 above 0 is expressed in 0.5 s increments. Thus if NRD = 120, the timeout value is actually 60 seconds.

The negative recal delay for a slider/wheel is taken from NRD at the lowest numbered key in the slider group, but ignored at all other keys within the group.

NRD Default value: 20 (10 s)

NRD Range: 0 – 255 (∞, 0.5 s – 127.5 s)

5.5 Addresses 1025 to 1088 – Slider member / AKS / Burst Length

Table 5-5. Slider member / AKS / Burst Length

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1025	SLD (Key 0)	AKS (Key 0)	BL (Key 0)		Reserved			
…	…	…	…					
1088	SLD (Key 63)	AKS (Key 63)	BL (Key 63)		Reserved			

5.5.1 Slider Member – SLD

Sliders and Wheels are constructed from a group of keys. Keys can operate as part of a Slider/Wheel or independently, as keys, but not both. A Key is declared as being part of a slider/wheel by setting its SLD bit, this inhibits the normal key processing which would otherwise interfere with the slider operation.

The data set for keys is undefined when used as part of a slider/wheel, and should be ignored.

Other configuration is also required for slider/wheel operation. See setups SLD, SSN and SLEN at a minimum and also SW, SPH, SF, and SRS. AKS, and FDIL must be disabled for slider/wheel members, and BL must be set appropriately for slider/wheel members. See the respective sections for details.

SLD default value: 0 (Normal key operation)

SLD Range: 0 (Normal key operation)
1 (Key participates in a wheel/slider)

5.5.2 Adjacent Key Suppression Technology – AKS

This QT2640 incorporates Adjacent Key Suppression (AKS) technology that can be selected on a per-key basis. AKS technology permits the suppression of multiple key presses based on relative signal strength. This feature assists in solving the problem of surface moisture which can bridge a key touch to an adjacent key, causing multiple key presses. This feature is also useful for panels with tightly spaced keys, where a fingertip might inadvertently activate an adjacent key.

AKS technology works for keys that are AKS-enabled anywhere in the matrix and is not restricted to physically adjacent keys. The QT2640 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 during the DI process. Once a key reaches detect it stays in detect as long as the touch remains, regardless of the signal strength on any other AKS-enabled keys.

This feature does not account for varying key gains (burst length) but ignores the actual negative detection threshold setting for the key. If AKS-enabled keys have different sizes, it may be necessary to reduce the gains of larger keys to equalize the effects of AKS technology. The signal threshold of the larger keys can be altered to compensate for this without causing problems with key suppression.

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

AKS is not supported on slider and wheel elements, and should be configured off at all keys used to construct a slider/wheel.

AKS Default value: 0 (Off)

AKS Range: 0 (Off)
1 (On)

5.5.3 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 X line. Each QT process is simply the pulsing of an X line once, with corresponding Y lines enabled to capture the resulting charge passed through the keys capacitance Cx.

QT2640 uses 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 Table 5-24 on page 56.

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 can be set for each key individually; charge is selectively captured on all Y lines simultaneously during the burst on each X line.

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 Default value: 1 (32 pulses)

BL Range: 0, 1, 2, 3 (16, 32, 48, 64 pulses / burst)

5.6 Addresses 1089 to 1216 – Calibrated Frequency Offset – CFO_1 and CFO_2

Table 5-6. Calibrated Frequency Offset

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1089	CFO_1 (Key 0)							
…	…							
1152	CFO_1 (Key 63)							
1153	CFO_2 (Key 0)							
…	…							
1216	CFO_2 (Key 63)							

CFO_1 and CFO_2 are only used if FHM = 2.

If frequency hopping is enabled with FHM = 2, the QT2640 adjusts each key reference at each frequency hop. The amount of the adjustment must first be configured using the setups CFO_1 and CFO_2. These values are the amounts by which reference levels need to be adjusted to track the signal changes when changing from the frequency set by FREQ0 to the frequency set by FREQ1 or FREQ2. Each key uses its individual settings of CFO_1 and CFO_2.

FREQ0 must define the highest of the three selected frequencies. ie be configured with the lowest value. CFO_1 indicates the offset to be applied when the frequency is changed between those defined by FREQ0 and FREQ1. CFO_2 indicates the offset to be applied when the frequency is changed between those defined by FREQ0 and FREQ2. The QT2640 automatically uses combinations of these offset values, as necessary, when switching between the three different frequencies.

The Calibrated Frequency Offset (CFO) values are used to adjust each key reference whenever a frequency hop occurs, taking into account any differences in calibrated signal at the different frequencies. The device uses the highest selected frequency as a point of reference and calculates offsets in calibrated signals at the other two frequencies.

Once FREQ0, FREQ1 and FREQ2 have been configured for the three chosen frequencies, CFO_1 and CFO_2 should be loaded with the signal offsets appropriate for the signal shifts observed when switching between the three different frequencies.

Follow these steps to determine the different signal levels for each key at each frequency and to determine appropriate values to load into CFO_1 and CFO_2.

1. Configure FREQ0, FREQ1 and FREQ2.
2. Configure FHM = 0 to disable frequency hopping temporarily.
3. Configure FREQ0 to select the highest of the chosen frequencies. i.e. FREQ0 should be set to a lower value than FREQ1 and FREQ2.
4. Recalibrate all keys.
5. Make a note of each key's reference level, Ref(k, f0)
6. Configure FREQ0 to select the second of the chosen frequencies.
7. Recalibrate all keys.
8. Make a note of each key's reference level, Ref(k, f1).
9. Configure FREQ0 to select the last of the chosen frequencies.
10. Recalibrate all keys.
11. Make a note of each key's reference level, Ref(k, f2).
12. Determine the difference in signal for each key in turn when the frequency changes from FREQ0 to FREQ1, using the following equation:
13. Diff(k, f0, f1) = Ref(k, f0) - Ref(k, f1)

    (this value will nearly always be positive. If it is negative, set Diff(k, f0, f1) to zero)
14. Store Diff(k, f0, f1) into the corresponding CFO_1 for each key k.

    Determine the difference in signal at each key when the frequency changes from FREQ0 to FREQ2, using the following equation:

    Diff(k, f0, f2) = Ref(k, f0) - Ref(k, f2)

    (this value will nearly always be positive. If it is negative, set Diff(k, f0, f1) to zero)
15. Store Diff(k, f0, f2) into the corresponding CFO_2 for each key k.
16. Configure FHM = 2.

Command 0xFD (Low level calibration and offset) can be written to the Command Address (768) to automatically compute offsets and fill out the CFO_1 and CFO_2 Setup entries, which can then be fine tuned manually.

These functions are programmed on a per-key basis. See Table 5-24 on page 56

CFO_1/2 Default value: 0

CFO_1/2 Range: 0 – 255

5.7 Address 1217 – Threshold Multiplier / Mains Sync / Sleep Drift / Neg. Hysteresis

Table 5-7. Threshold Multiplier / Mains Sync / Sleep Drift / Neg Hys

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1217	THRM		MSYNC	SDC			NHYST	

5.7.1 Threshold Multiplier – THRM

It is sometimes useful to be able to operate the QT2640 with much higher detect thresholds than can be set with NTHR alone. The Threshold Multiplier (THRM) is a multiplier which extends the range of NTHR and PTHR considerably. The operating detect threshold for a key is arrived at by multiplying NTHR or PTHR for that key by THRM. Note that the detect threshold range is extended at the expense of the step size. Table 5-8 shows the extended threshold range for each value of THRM.

Table 5-8. Extended Detect Threshold

THRM	Multiplier	Extended Detect Threshold
0	× 1	6 – 36
1	× 2	12 – 72
2	× 4	24 – 144
3	× 8	48 – 288

This function is programmed on a global basis. See Table 5-24 on page 56.

THRM Default value: 0 (× 1)

THRM Possible range: 0, 1, 2, 3 (× 1, × 2, × 4, × 8)

5.7.2 Mains Sync – MSYNC

The MSync feature uses the VREF/WS pin. The Sleep and Sync features can be used simultaneously, in which case the QT2640 wakes on the mains sync signal, scans the matrix and then returns to sleep automatically.

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 50 Hz or 60 Hz 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 VREF/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 the lowest numbered enabled key in the matrix. If it does not sleep at the end of the matrix scan, the QT2640 waits for the sync signal for up to 100 ms after the end of a preceding full matrix scan, then the matrix is scanned in its entirety again. If the QT2640 sleeps, it waits indefinitely for the mains sync

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. The QT2640 synchronizes to the rising sync edge. If the sync feature is enabled, two transistors should be used to drive the VREF/WS pin (See Section 2.18 on page 18).

Since noise sync is highly effective and inexpensive to implement, it is strongly advised to take advantage of it anywhere there is a possibility of encountering low frequency (50/60 Hz) electric fields. The Atmel QmBtn software can show such noise effects on signals, and therefore assist in determining the need to make use of this feature.

If the sync feature is enabled but no sync signal exists, the sensor continues to operate but with a delay of 100 ms from the end of one scan to the start of the next, and hence has a slow response time. A failed Sync signal (one exceeding a 100 ms period) causes an error flag which is reported in the device status (address 5). From reset, the QT2640 first reports a mains sync error after initialisation followed by a delay of 100 ms waiting for the sync signal. This time interval may be determined by adding 100 ms to the initialisation times stated in Section 2.12 on page 13.

MSYNC Default value: 0 (Off)

MSYNC Range: 0, 1 (Off, On)

5.7.3 Sleep Drift Compensation – SDC

SDC allows the QT2640 to be configured for automatic sleep, and for modified drift compensation when sleep is enabled. Whenever the QT2640 goes to sleep, the whole device is shutdown, including the clock generator. All operations are stopped including matrix scanning and timers, which results in the internal time keeping running very slow and, in particular, drift compensation runs at a rate much slower than configured by NDRIFT and PDRIFT (see Section 5.2 on page 38).

For example, with NDRIFT and PDRIFT configured such that drift compensation occurs once every second when the QT2640 is awake, and with the QT2640 being awake for 5 ms to scan the matrix before falling asleep for 495 ms, all internal timers are slowed by a factor of 100 (5/500), and drift compensation would occur at the much slower rate of just once every 100 s. This would typically result in the key references not tracking signal variations adequately, and could result in false detections. To help resolve this, SDC can be configured so that the QT2640 performs drift compensation after a specific number of sleeps.

- With SDC = 0, sleep is disabled
- With SDC = 1, drift compensation occurs after every sleep
- With SDC = 7, drift compensation is applied after every 64 sleeps

This function is programmed on a global basis. See Table 5-26 on page 58.

SDC Default value: 0 (Off, Sleep disabled)

SDC Range: 0 – 7 (Off, 1 – 64)

See also Section 2.15 on page 15 and Section 5.12.2 on page 48.

5.7.4 Negative Hysteresis – NHYST

The QT2640 employs 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 drops 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 only used once the key has been declared to be in detection, in order to determine when the key should drop out of detect.

Excessive amounts of hysteresis can result in stuck keys that do not release. Conversely, low amounts of hysteresis can cause key chatter due to noise or minor amounts of finger motion.

The hysteresis level is set for all keys globally; it is not possible to set the hysteresis differently from key to key. Sliders and Wheels ignore NHYST and do not employ detect hysteresis.

NHYST Default value: 1 (12.5%)

NHYST Range: 0, 1, 2, 3 (6.25%, 12.5%, 25%, 50%)

5.8 Address 1218 – Dwell Time – DWELL

Table 5-9. DWELL

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1218	Reserved				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 required where the keypanel design includes higher-resistance tracks such as silver and ITO. Longer durations are also usually more compatible with EMC requirements, permitting the use of larger series resistors in the X and Y lines to suppress RFI effects, without compromising key gain (Section 2.9 on page 10). This setup lets the designer trade one requirement for another.

DWELL Default value : 1 (190 ns)

DWELL Range: 0 – 15 (125 ns – 9.9 µs), accuracy is ±10%

5.9 Addresses 1219 to 1220 – Lower Signal Limit – LSL

Table 5-10. LSL

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1219	LSL (bits 7 .. 0)							
1220	See Section 5.10					LSL (bits 10 .. 8)		

This Setup determines the lowest acceptable value of signal level for all keys. If any key's reference level falls below this value, the QT2640 declares an error condition in the key status bits (See Section 4.4 on page 32 and Section 4.11 on page 35).

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.

This parameter occupies 2 bytes (11 bits) of the setups table; address 1219 holds the LSbits, bits 0 – 2 of address 1220 the MSbits.

LSL Default value: 100

LSL Range: 0 – 2047

5.10 Address 1220 – FMEA Test Threshold (KGTT) / Restart Interrupted Burst / LSL

Table 5-11. FMEA Test Threshold / Restart Interrupted Burst / LSL

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1220	KGTT				RIB	See Section 5.9		

5.10.1 Key Gain Test Threshold – KGTT

The Key Gain test takes a special sample from each enabled key using half the usual burst length, and compares the resulting signal against each key's normal signal. The test passes if the signal has decreased by the Key Gain Test Threshold (KGTT). The following equation must hold for the test to pass:

(Normal Signal – Test Signal) >= KGTT

Disabled keys are not tested.

The Key Gain Test Threshold can be configured to a value between 4 and 64, via LUT (see Table 5-26 on page 58).

This function is programmed on a global basis.

KGTT Default value: 7 (32)

KGTT Possible range: 0 – 15 (4 – 64)

5.10.2 Restart Interrupted Burst – RIB

The RIB parameter in setups allows a burst to be interrupted, and restarted, by host communications over the serial bus. The QT2640 has limited processing resources available such that a burst and host communication cannot both be serviced simultaneously. One must give way to the other. This setup lets the designer prioritize one over the other.

If RIB is configured on, a burst can be interrupted by a host communication, and is automatically restarted.

If RIB is configured off, bursts cannot be interrupted but, rather, the host communication is delayed until the burst has completed, and the DRDY low period is stretched by the QT2640 during the burst.

This function is programmed on a global basis. See Table 5-26 on page 58.

RIB Default value: 0 (Off)

RIB Range: 0, 1 (Off, On)

5.10.3 Lower Signal Limit – LSL

See Section 5.9 on page 46

5.11 Address 1221 – DEBUG

Table 5-12. DEBUG

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1221	Reserved							DEBUG

The QT2640 supports a Debug output interface which is useful during product development. (See Appendix B. on page 65). If DEBUG is enabled, the device streams one frame of data out of the Debug port after each keyscan cycle.

When DEBUG is enabled it impacts the key response time because the next keyscan cycle is delayed until the debug frame has been fully transmitted.

The oscilloscope sync and the debug clock output share an I/O pin, and will interfere with each other if both are enabled at the same time. Ensure SSYNC and DEBUG are not enabled simultaneously.

$\overline{\text{CHANGE}}$ and the debug data output share an I/O pin, and would interfere with each other if they were both allowed to operate simultaneously. To prevent cross interference, all $\overline{\text{CHANGE}}$ activity is disconnected from the I/O pin when DEBUG is enabled. DEBUG takes precedence over $\overline{\text{CHANGE}}$.

This function is programmed on a global basis. See Table 5-26 on page 58.

DEBUG Default value: 0 (Off)

DEBUG Range: 0, 1 (Off, On)

5.12 Address 1222 – Drift Hold Time / AWAKE

Table 5-13. Drift Hold Time / AWAKE

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1222	DHT				AWAKE			

5.12.1 Drift Hold Time – DHT

Drift Hold Time (DHT) is used to suspend drift compensation from all keys while the keypad is being used. With the feature enabled, drift compensation is suspended while any key is touched and also for a period afterwards. DHT defines the length of time the drift compensation continues to be suspended after a key detection has finished.

This feature is particularly useful in cases of high-density keypads where touching a key or hovering a finger over the keypad would cause untouched keys to drift, and therefore create a sensitivity shift, and ultimately inhibit other touch detections.

DHT can be configured to one of 16 values between 100 ms and 25.4s via LUT (see Table 5-26 on page 58).

This function is programmed on a global basis.

DHT Default value: 1 (200 ms)

DHT Range: 0 – 15 (100 ms – 25.4 s)

5.12.2 Awake Time – AWAKE

The AWAKE feature is effective only if the part has been configured for automatic sleep, via SDC (see Section 5.7.3 on page 45), and if the Request Sleep command has been issued (see Section 4.13 on page 35). AWAKE determines the period of time that elapses from the last key release before the part tries to sleep.

An internal timer is restarted at each key release and runs for the time configured via AWAKE. The part will not enter sleep while this timer is running. See also Section 2.15 on page 15.

Note: If the sleep feature has been disabled, the QT2640 never sleeps and the AWAKE setup has no effect.

The AWAKE period can be configured to one of 16 values between 100 ms and 25.4 s via LUT (see Table 5-26 on page 58).

This function is programmed on a global basis.

AWAKE Default value: 15 (25.4 s)

AWAKE Range: 0 – 15 (100 ms – 25.4 s)

AWAKE Timeout accuracy: ±50 ms

5.13 Address 1223 – Oscilloscope Sync – SSYNC

Table 5-14. SSYNC, Generate scope sync for burst on selected X lines.

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1223	X7	X6	X5	X4	X3	X2	X1	X0

The S_SYNC pin can output a positive pulse oscilloscope sync that brackets the burst of a selected X line. More than one burst can output a sync pulse as determined by the setups parameter SSYNC which is divided into eight separate bits, each allowing selection of a scope sync at the burst on one X line.

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

Note: The oscilloscope sync and the debug clock output share an I/O pin, and will interfere with each other if both are enabled at the same time. Ensure SSYNC and DEBUG are not enabled simultaneously.

This function is supported in Atmel's QmBtn PC software.

SSYNC Default value: 0 (No scope sync for any burst)

SSYNC Range: 0 - 255 (No sync at any burst - sync at every burst)

5.14 Address 1224 – Frequency Hopping Mode / Positive Recalibration Delay

Table 5-15. Frequency Hopping Mode / Positive Recalibration Delay

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1224	FHM		Reserved		PRD			

5.14.1 Frequency Hopping Mode – FHM

If frequency hopping is enabled, the QT2640 continually monitors the noise across all keys and switches frequency to try and find the frequency with least signal noise. If frequency hopping is disabled, the QT2640 always uses the same frequency, defined by FREQ0.

This function is programmed on a global basis. See Table 5-24 on page 56.

FHM Default value: 3 (Sweep)

FHM Range: 0 – 3 (0 = off
1 = Calibrate all keys after hop
2 = Adjust each key's reference during hop)
3 = Sweep)

5.14.2 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 autorecalibration, 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 recalibrated to a new lower reference level. Then, when the condition causing the negative swing ceases to exist (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 once again functions normally. PRD is set on a global basis.

The functioning of the PRD setting is determined by an offset to a lookup table, found on page 56. The values of time can range from 0.1 s to 25 s. Setting the parameter to 0 disables the feature.

PRD Default value: 8 (2 s)
PRD Range: 0 – 15 (º, 0.1 – 25s)

5.15 Address 1225 – Frequency 0 – FREQ0

Table 5-16. FREQ0

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1225	FREQ0							

FREQ0 is used in all frequency hopping modes and even if frequency hopping is disabled. In all modes, it defines the idle time between pulses in the burst. Larger values yield longer times between pulses and thus a lower fundamental frequency.

FHM = 0 – Frequency hopping disabled

If frequency hopping is disabled, the QT2640 always uses the same frequency, defined by FREQ0. Frequency hopping might not be desirable in all applications and it might be more appropriate to preselect a burst frequency at the factory which is known not to coincide with other operating frequencies within the end product or other frequencies in the operating environment. In such cases, FHM can be set at zero, and the burst frequency set with FREQ0.

With DWELL set at the minimum, the fundamental frequency can be set in the range 31KHz – 943KHz. The frequency for a specific DWELL and FREQ0 combination should be measured using an oscilloscope (temporarily disable frequency hopping to make the measurement easier).

FHM = 1 or FHM = 2 – Frequency hopping between three frequencies

If frequency hopping is enabled with FHM = 1 or 2, the QT2640 can hop between three frequencies configured using FREQ0, FREQ1 and FREQ2. FREQ0 must be set as the highest frequency otherwise the behavior is undefined; that is, FREQ1 and FREQ2 must be set to values greater than or equal to FREQ0.

FHM = 3 – Frequency sweep

With FHM = 3, the QT2640 sweeps a range of frequencies, with the upper frequency boundary (shortest idle time) defined by FREQ0. FREQ1 must be set to a value greater than FREQ0; the behavior is otherwise undefined.

FREQ0 = 0 is invalid and must not be used.

This function is programmed on a global basis. See Table 5-24 on page 56.

FREQ0 Default value: 1 (delay cycle)

FREQ0 Range: 1 – 255 (Highest frequency to Lowest frequency)

0 is invalid.

5.16 Address 1226 – Frequency1 – FREQ1

Table 5-17. FREQ1

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1226	FREQ1							

FREQ1 is only used if FHM is set to a non-zero value; it is not used if frequency hopping is disabled with FHM = 0.

With FHM = 1 or FHM = 2, FREQ1 allows configuration of one of three operating frequencies by defining the idle time between pulses in the burst. Larger values yield longer times between pulses and thus a lower fundamental frequency.

With FHM = 3, the QT2640 sweeps a range of frequencies, with the lower frequency boundary (longest idle time) defined by FREQ1. FREQ1 must be set to a value greater than FREQ0; the behavior is otherwise undefined.

This function is programmed on a global basis. See Table 5-24 on page 56.

FREQ1 Default value: 6 (delay cycles)

FREQ1 Range: 1 – 255 (Highest frequency to Lowest frequency)

0 is invalid.

5.17 Address 1227 – Frequency2 – FREQ2

Table 5-18. FREQ2

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1227	FREQ2							

FREQ2 is only used if FHM = 1 or FHM = 2; it is not used if FHM is set to 0 or 3.

With FHM = 1 or FHM = 2, FREQ2 allows configuration of one of three operating frequencies by defining the idle time between pulses in the burst. Larger values yield longer times between pulses and thus a lower fundamental frequency.

This function is programmed on a global basis. See Table 5-24 on page 56.

FREQ2 Default value: 63 (delay cycles)

FREQ2 Range: 1 – 255 (Highest frequency to Lowest frequency)

0 is invalid.

5.18 Address 1228 – Noise Integrator Limit / Noise Threshold

Table 5-19. Noise Integrator Limit / Noise Threshold

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1228	NIL				NSTHR			

5.18.1 Noise Integrator Limit – NIL

NIL is only used if FHM = 1 or FHM = 2; it is not used if FHM is set to 0 or 3.

The QT2640 considers a hop to one of the other frequencies when the noise at the current frequency consistently exceeds the threshold configured with NSTHR. To prevent true touch events and other brief signal anomalies being considered as noise, the QT2640 employs counters, termed noise integrators, to track the number of signal deltas that exceed the noise threshold. It maintains two such counters, one for positive deltas and one for negative deltas.

The QT2640 considers a frequency hop only if both these counters reach the noise integrator limit (NIL). The counters are reset to zero at the end of each matrix scan.

This mechanism provides a robust way of detecting strong noise while suppressing unnecessary frequency hopping.

This function is programmed on a global basis. See Table 5-24 on page 56.

NIL Default value: 3

NIL Range: 3 – 15

(0 – 2 = factory use only)

5.18.2 Noise Threshold – NSTHR

NSTHR is only used if FHM = 1 or FHM = 2; it is not used if FHM is set to 0 or 3.

When FHM = 1 or 2, the QT2640 considers a hop to one of the other frequencies when the noise at the current frequency consistently exceeds the threshold configured with NSTHR. NSTHR is used by the frequency hopping algorithms to determine if a signal delta should be considered as noise. A delta, of either polarity, greater than or equal to NSTHR is considered as possible noise and forces the Noise Integrator counter to be incremented.

This function is programmed on a global basis. See Table 5-24 on page 56.

NSTHR Default value: 2 (11 counts)

NSTHR Range: 0 – 15 (5 – 50 counts)

5.19 Addresses 1229 – Slider Filter / Slider Position Hysteresis

Table 5-20. Slider Filter / Slider Position Hysteresis

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1229	Reserved					SF	SPH	

5.19.1 Slider Filter – SF

Optional filtering may be applied to the position reported for the sliders/wheels. This setting applies globally to all sliders/wheels, engaging the internal filter to remove some of the noise on the reported position at the expense of additional processing time.

SF default value: 1 (on)

SF Range: 0, 1 (off, on)

5.19.2 Slider Position Hysteresis – SPH

The device employs programmable hysteresis levels of between one and eight counts of signal. The hysteresis defines the distance by which the touch position must reverse direction before the reported position is updated confirming the change in direction.

Hysteresis prevents some noise in the reported position.

SPH is set using one global value which is applied across all sliders/wheels.

SPH Default value: 0 (1 count)

SPH Range: 0..3 (1 .. 8 counts)

5.20 Addresses 1230 to 1237 – Slider or Wheel/ Slider Start Node

Table 5-21. Slider or Wheel / Slider Start Node

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1230	Reserved	SW (Slider 0)	SSN (Slider 0)					
…		…	…					
1237	Reserved	SW (Slider 7)	SSN (Slider 7)					

5.20.1 Slider or Wheel– SW

The keys forming a slider/wheel may be physically laid out in either a linear array, in which case the result is called a slider, or arranged in a circular fashion where the two ends of the slider meet to form a shape reminiscent of a wheel. Different processing is required for these two designs to ensure a smooth transition in the reported position between the keys forming the end and beginning of a wheel. SW indicates the physical layout being used for each slider/wheel.

SW Default value: 0 (slider)

SW Range: 0, 1 (slider, wheel)

5.20.2 Slider Start Node – SSN

There are eight SSN, one for each of the possible sliders/wheels. These values indicate where in the key matrix each slider/wheel starts. Each SSN value indicates the lowest key number used in the slider/wheel.

Sliders/wheels must each be configured to occupy an exclusive range of nodes in the matrix, and must not be configured to overlap which each other otherwise the behavior is undefined.

SSN Default value: 0

SSN Range: 0 – 62

(63 is invalid)

5.21 Addresses 1238 to 1245 – Slider Range Stretch / Slider Length

Table 5-22. Slider Range Stretch / Slider Length

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1238	Reserved		SRS (Slider 0)			SLEN (Slider 0)		
…			…			…		
1245	Reserved		SRS (Slider 7)			SLEN (Slider 7)		

5.21.1 Slider Range Stretch – SRS

Whereas the theoretical reported output value of a slider is in the range 0 to 255, the practical range of achievable values will typically be somewhat less, with both ends of the theoretical range missing from the reported output. This is due to the fact that the ends of the slider are abruptly terminated in the first and last keys, and also perhaps because the overlaying touch panel may have physical end stops, preventing touch along the full length of the slider. These limitations may be overcome by stretching the default reported range to fill the possible full scale from 0 to 255.

The default reported range can be stretched in eight steps of 6.25% from 6.25% to 50%, and configured individually for each slider.

This setup is ignored if SW=1, ie the design is for a wheel.

SRS Default value: 0 (6.25%)

SRS Range: 0 – 7 (6.25% – 50%)

5.21.2 Slider Length – SLEN

There are eight SLEN, one for each of the possible sliders/wheels, each of which indicates the number of keys used to construct the corresponding slider/wheel. Values range from the minimum of 2 keys to the maximum of eight keys.

SLD must also be set for all keys in a slider/wheel group. SLD disables the normal key processing and makes the keys data set available for the necessary slider processing.

SLEN=0 disables the slider, and all sliders with a higher number too (all SLEN following any SLEN set at zero are ignored and also assumed to be zero).

Sliders/wheels must each be configured to occupy an exclusive range of nodes in the matrix, and must not be configured to overlap which each other otherwise the behavior is undefined.

SLEN Default value: 0 (Off)

SLEN Range: 0, 1 – 7 (Off, 2 – 8)

5.22 Address 1246, 1247 – Host CRC– HCRC

Table 5-23. Host CRC

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1246	HCRC, LSB							
1247	HCRC, MSB							

The setups block terminates with a 16-bit CRC, HCRC, of the entire block. The host should calculate this CRC and upload it here; The QT2640 repeatedly recomputes the CRC on the rest of the setups and compares the computed value against HCRC. If a mismatch is found, an error flag is reported at the device status address (see Section 4.4 on page 32).

The formulae for calculating this CRC is shown in Appendix A. The CRC is initialized to zero and is calculated by folding in each setups byte in address sequential order starting with the byte from the first setups block address(769) and finishing with the byte stored immediately before HCRC.

During development it is desirable to prove the CRC algorithm in the host. One way to do this is to calculate the CRC on the factory programmed setups block and compare the result against HCRC held in the QT2640. If this is done, it is recommended to use the setups block downloaded from the QT2640 because the status of some reserved bits are undefined and could make it difficult to initially reproduce the factory programmed HCRC.

5.23 Setups Block Summary

Table 5-24. Setups Block Summary – Per Key Settings

	Parameter											
Index	**NTHR Counts**	**PTHR Counts**	**NDRIFT Secs**	**PDRIFT Secs**	**NDIL Counts**	**FDIL Counts**	**NRD Secs**	**SLD**	**BL Pulses**	**AKS**	**CFO_1**	**CFO_2**
	Per Key											
0	6	6	0.1	0.1	Key off	Unused	0 (infinite)	**Key**	16	**Off**	**0**	**0**
1	8	8	0.2	0.2	1	1	0.5 – 127 s	Slider / Wheel	**32**	On	1 – 255	1 – 255
2	10	10	0.3	0.3	2	2	**10 s**		48			
3	12	12	0.4	0.4	3	3			64			
4	14	14	0.6	0.6	4	4						
5	16	16	0.8	0.8	5	5						
6	**18**	**18**	1	1	6	6						
7	20	20	1.2	1.2	7	7						
8	22	22	1.5	1.5	8	8						
9	24	24	2	2	9	9						
10	26	26	**2.5**	**2.5**	10	10						
11	28	28	3.3	3.3	11	11						
12	30	30	4.5	4.5	12	12						
13	32	32	6	6	13	13						
14	34	34	7.5	7.5	14	14						
15	36	36	10	10	15	15						

Notes: Bold text items indicate default settings. The number to send to the QT2640 is the index number in the leftmost column (0 – 15), not numbers from the table. The QT2640 uses a lookup table internally to translate the indices 0 – 15 to the parameters for each function.

NRD is an exception: It can range from 0 – 254 which is translated from 1 = 0.5 s to 254 = 127 s with zero = infinity.

CFO_1 and **CFO_2** are exceptions. Their values, ranging from 0 to 255, are used directly and without any translation.

Table 5-25. Setups Block Summary – Slider Settings

	Parameter							
Index	**SLD**	**SSN**	**SW**	**SLEN**	**SRS**		**SPH**	**SF**
	Per Slider						Global	
0	**Key**	**0**	**Slider**	**Off**	**6.25%**		**1**	Off
1	Slider / Wheel	0 – 62	Wheel	2	12.5%		2	**On**
2				3	18.75%		4	
3				4	25%		8	
4				5	31.25%			
5				6	37.5%			
6				7	43.75%			
7				8	50%			

Table 5-26. Setups Block Summary – Global Settings

	Parameter																		
Index	**RIB**	**NHYST**	**SDC**	**MSYNC**	**DWELL µs**	**PRD Secs**	**KGTT**	**AWAKE secs**	**DHT secs**	**SSYNC**	**THRM**	**FHM**	**FREQ0 cycles**	**FREQ1 cycles**	**FREQ2 cycles**	**NSTHR**	**NIL**	**LSL**	**DEBUG**
	Global									X line	Global								
0	**Off**	6.25%	**Off**	**Off**	0.13	0 (infinite)	4	0.1	0.1	**Off**	**× 1**	Off	-	-	-	5	0 (factory only)	0 – 2047	**Off**
1	On	**12.5%**	1	On	**0.19**	0.1	8	0.2	**0.2**	On	× 2	Recal	1 – 255	1 – 255	1 – 255	8	1 (factory only)	**100**	On
2		25%	2		0.4	0.2	12	0.3	0.3		× 4	Adjust	**1**	**6**	**63**	**11**	2 (factory only)		
3		50%	4		0.6	0.3	16	0.5	0.5		× 8	**Sweep**				14	**3**		
4			8		0.8	0.5	20	0.7	0.7							17	4		
5			16		0.9	0.7	24	1	1							20	5		
6			32		1.1	1	28	1.5	1.5							23	6		
7			64		1.3	1.5	**32**	2	2							26	7		
8					1.5	**2**	36	3.2	3.2							29	8		
9					1.7	3.2	40	4.5	4.5							32	9		
10					2.1	4.5	44	6	6							35	10		
11					2.6	6	48	9	9							38	11		
12					3.8	9	52	12.3	12.3							41	12		
13					5.1	12.3	56	15	15							44	13		
14					7.1	17.5	60	19	19							47	14		
15					9.9	25	64	**25.4**	25.4							50	15		

Notes: Bold text items indicate default settings. The number to send to the QT2640 is the index number in the leftmost column (0 – 15), not numbers from the table. The QT2640 uses a lookup table internally to translate the indices 0 – 15 to the parameters for each function.
LSL is an exception: Ranging from 0 to 2047, its value is used directly and without any translation.

6. Specifications

6.1 Absolute Maximum Specifications

Parameter	Specification
Vdd	6V
Max continuous pin current, any control or drive pin	±10 mA
Short circuit duration to ground, any pin	infinite
Short circuit duration to Vdd, any pin	infinite
Voltage forced onto any pin	–0.5 V to (Vdd + 0.5) V
Frequency of operation	17 MHz
EEPROM setups maximum writes	100,000 write cycles
CAUTION: Stresses beyond those listed may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum specification conditions for extended periods may affect device reliability.	

6.2 Recommended Operating Conditions

Parameter	Specification
Operating temp	–40°C to +105°C
Storage temp	–55°C to +125°C
Vdd	+4.75 V to +5.25 V
Supply ripple + noise	20 mV p-p max
Cx transverse load capacitance per key	0 to 20 pF
Fosc oscillator frequency	16 MHz ±2%

6.3 DC Specifications

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

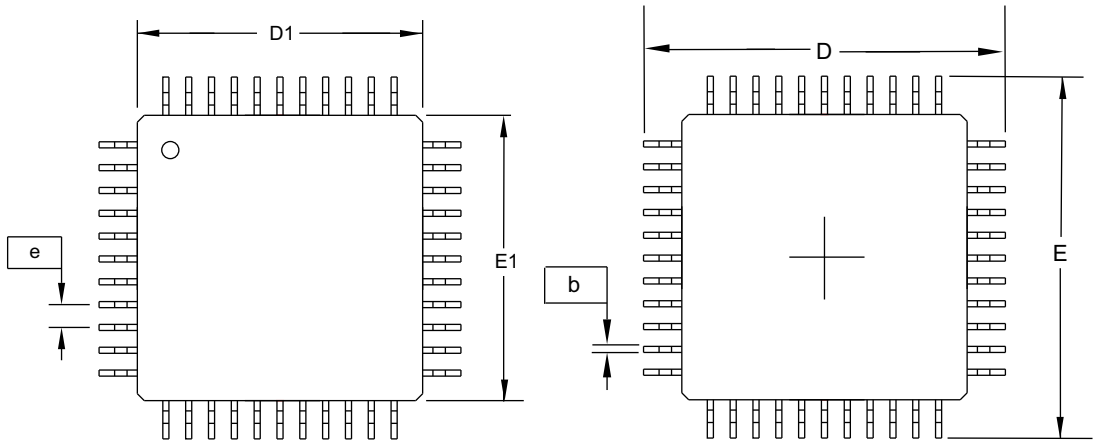
Parameter	Description	Min	Typ	Max	Units	Notes
Iddr	Supply current, running	–	13	–	mA	
Idds	Supply current, sleeping	–	15	–	µA	
Vr	Internal reset voltage	–	4	–	V	Internal brown-out protection
Vil	Low input logic level	–	–	0.2 × Vdd	V	
Vih	High input logic level	0.6 × Vdd	–	–	V	
Vol	Low output voltage	–	–	0.6	V	4 mA sink
Voh	High output voltage	Vdd – 0.7	–	–	V	1 mA source
Iil	Input leakage current	–	–	±1	µA	
Ar	Acquisition resolution	–	9	11	bits	
Rp	Internal pull-up resistors	20	–	50	kΩ	$\overline{\text{DRDY}}$, $\overline{\text{SS}}$, $\overline{\text{CHANGE}}$ pins
Rrst	Internal $\overline{\text{RST}}$ pull-up resistor	30	–	60	kΩ	

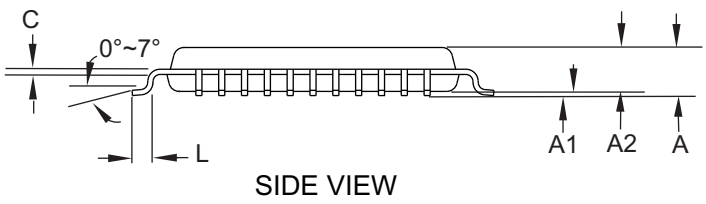
6.4 Timing Specifications

See Figure 3-2 on page 24 for definitions of the parameters.
Vdd = 5.0V, Cs = 4.7 nF, Freq = 16 MHz, Ta (Ambient Temperature) = recommended range, unless otherwise noted

Parameter	Description	Min	Typ	Max	Units	Notes
S1	↓$\overline{\text{SS}}$ to first ↓CLK edge	125	–		ns	SPI parameter controlled by host
S2	↓CLK to valid MISO	–	–	20	ns	SPI parameter controlled by QT2640
S3	Last ↑CLK to ↑$\overline{\text{SS}}$	25	–	–	ns	SPI parameter controlled by host
S4	↑$\overline{\text{SS}}$ to 3-state MISO	–	–	20	ns	SPI parameter controlled by QT2640
S5	↑$\overline{\text{SS}}$ to falling $\overline{\text{DRDY}}$	–	–	20	µs	SPI parameter controlled by QT2640
S6	$\overline{\text{DRDY}}$ low pulse width	1	–	–	µs	SPI parameter controlled by QT2640
S7	CLK low pulse width	125	–	–	ns	SPI parameter controlled by host
S8	CLK high pulse width	125	–	–	ns	SPI parameter controlled by host
S9	CLK period	250	–	–	ns	SPI parameter controlled by host
Fck	SPI Clock rate	–	–	4	MHz	SPI parameter controlled by host

6.5 Mechanical Dimensions





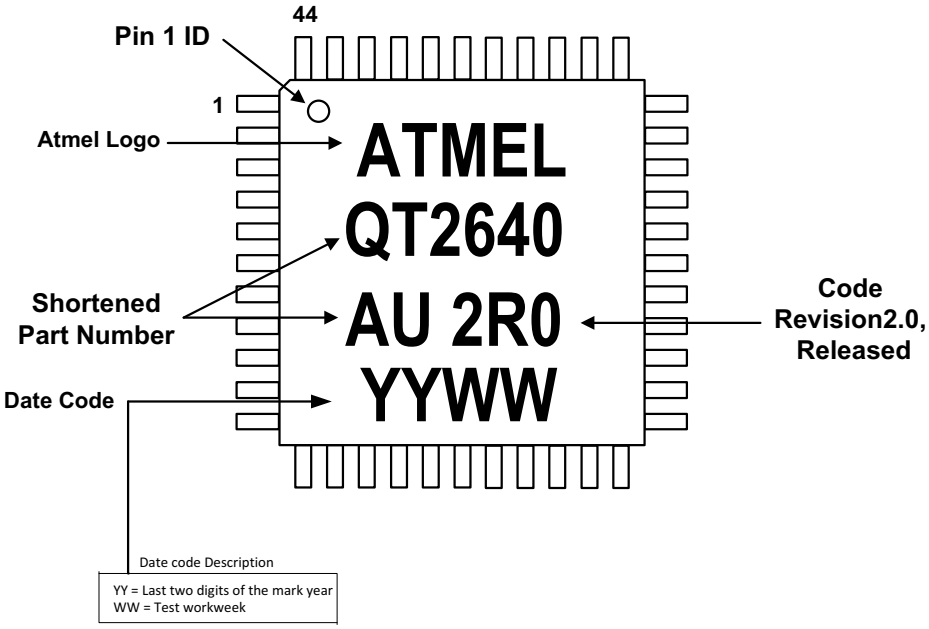
COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.20	
A1	0.05	–	0.15	
A2	0.95	1.00	1.05	
D	11.75	12.00	12.25	
D1	9.90	10.00	10.10	Note 2
E	11.75	12.00	12.25	
E1	9.90	10.00	10.10	Note 2
B	0.30	–	0.45	
C	0.09	–	0.20	
L	0.45	–	0.75	
e	0.80 TYP			

Notes:
1. This package conforms to JEDEC reference MS-026, Variation ACB.
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is 0.25 mm per side. Dimensions D1 and E1 are maximum plastic body size dimensions including mold mismatch.
3. Lead coplanarity is 0.10 mm maximum.

Atmel Package Drawing Contact: packagedrawings@atmel.com	TITLE **44A**, 44-lead 10.0 x 10.0x1.0 mm Body, 0.80 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP)	GPC AIX	DRAWING NO. 44A	REV. D

6.6 Marking


44
Pin 1 ID
1
Atmel Logo
ATMEL
QT2640
Shortened
Part Number
AU 2R0
Code
Revision2.0,
Released
Date Code
YYWW
Date code Description
YY = Last two digits of the mark year
WW = Test workweek

6.7 Part Number

Part Number	Description
AT42QT2640-AU	44-lead 10 × 10 mm TQFP RoHS compliant IC
AT42QT2640-AUR	44-lead 10 × 10 mm TQFP RoHS compliant IC - Tape and reel

The part number comprises:

AT = Atmel

42 = Touch Business Unit

QT = Charge-transfer technology

2640 = (2) slider/wheel, (64) number of channels, (0) variant number

AU = TQFP chip

R = Tape and reel

6.8 Moisture Sensitivity Level (MSL)

MSL Rating	Peak Body Temperature	Specifications
MSL3	260°C	IPC/JEDEC J-STD-020

Appendix A. 16-bit CRC Algorithm

```
// 16 bits crc calculation. Initial crc entry value must be 0.
// The message is not augmented with 'zero' bits.
// polynomial = X16 + X12 + X5 + 1
// data is an 8 bit number, unsigned
// crc is a 16 bit number, unsigned
// repeat this function for each data block byte, folding the result
//   back into the call parameter crc

unsigned long sixteen_bit_crc(unsigned long crc, unsigned char data)
{     unsigned char index;// shift counter

      crc ^= (unsigned long)(data) << 8;
      index = 8;
      do                                              // loop 8 times
      {     if(crc & 0x8000)
            {     crc= (crc << 1) ^ 0x1021;
            }
            else
            {     crc= crc << 1;
            }
      } while(--index);
      return crc;
}
```

A CRC calculator for Microsoft Windows is available free of charge from Atmel.

Appendix B. DEBUG Output

The QT2640 includes a debug interface which may be used for observing many internal operating variables, in real time, even while the part is actively communicating with a host over the SPI serial interface. The Debug interface provides a useful aid during product development and uses two pins, one for clock and one for data, to stream data out of the part.

If DEBUG is enabled the QT2640 streams a multi-byte frame of data out of the two debug pins after each keyscan cycle. The transmission format is compatible with the Atmel Plug-in USB card (Part Number 9206) and the data can be viewed using Atmel's Hawkeye PC software (contact Atmel for information). Table 6-1 shows the Debug interface details.

See also Section 5.11 on page 47.

Table 6-1. Debug Interface

Interface Element	Description
Debug Clock output	Pin 40 (DBG_CLK), shared with S_SYNC
Debug Data output	Pin 43 (DBG_DATA), shared with $\overline{\text{CHANGE}}$
Data valid	Clock high
Data changing	Clock low
Clock frequency	Approximately 500 kHz
Blank time between byte transmissions	5.5 µs
Frame transmission time	10.5 ms
Frame Length	538 bytes
Byte transmission order	Most Significant Bit (MSB) first

The meaning of each byte in the frame is described in Table 6-2.

Table 6-2. Debug Output Data Frame

Frame Byte #	Description
0	Reserved
1	Reserved
2	100ms counter (IEC/EN/UL60730)
3	Signal fail counter (IEC/EN/UL60730)
4	Matrix scan counter (IEC/EN/UL60730)
5	Device status. See Section 4.4 on page 32
6	Detect status for keys 0 (bit0) to 7 (bit7), one bit per key
7	Detect status for keys 8 (bit0) to 15 (bit7), one bit per key
8	Detect status for keys 16 (bit0) to 23 (bit7), one bit per key
9	Detect status for keys 24 (bit0) to 31 (bit7), one bit per key
10	Detect status for keys 32 (bit0) to 39 (bit7), one bit per key

Table 6-2. Debug Output Data Frame (Continued)

Frame Byte #	Description
11	Detect status for keys 40 (bit0) to 47 (bit7), one bit per key
12	Detect status for keys 48 (bit0) to 55 (bit7), one bit per key
13	Detect status for keys 56 (bit0) to 63 (bit7), one bit per key
14	Detect status for sliders 0 (bit0) to 7 (bit7), one bit per slider/wheel
15	Slider 0 touch position
16	Slider 1 touch position
17	Slider 2 touch position
18	Slider 3 touch position
19	Slider 4 touch position
20	Slider 5 touch position
21	Slider 6 touch position
22	Slider 7 touch position
23	Frequency hop mode, copy of FHM
24	Index of current frequency
25	Current pulse spacing
26 – 33	Data for key 0. See Table 4-8 on page 35 for details of the data set for each key.
34 – 41	Data for key 8
42 – 49	Data for key 16
50 – 57	Data for key 24
58 – 65	Data for key 32
66 – 73	Data for key 40
74 – 81	Data for key 48
82 – 89	Data for key 56
90 – 97	Data for key 1
98 – 105	Data for key 9
106 – 113	Data for key 17
114 – 121	Data for key 25
122 – 129	Data for key 33
130 – 137	Data for key 41
138 – 145	Data for key 49
146 – 153	Data for key 57
154 – 161	Data for key 2
162 – 169	Data for key 10

Table 6-2. Debug Output Data Frame (Continued)

Frame Byte #	Description
170 – 177	Data for key 18
178 – 185	Data for key 26
186 – 193	Data for key 34
194 – 201	Data for key 42
202 – 209	Data for key 50
210 – 217	Data for key 58
218 – 225	Data for key 3
226 – 233	Data for key 11
234 – 241	Data for key 19
242 – 249	Data for key 27
250 – 257	Data for key 35
258 – 265	Data for key 43
266 – 273	Data for key 51
274 – 281	Data for key 59
282 – 289	Data for key 4
290 – 297	Data for key 12
298 – 305	Data for key 20
306 – 313	Data for key 28
314 – 321	Data for key 36
322 – 329	Data for key 44
330 – 337	Data for key 52
338 – 345	Data for key 60
346 – 353	Data for key 5. See Table 4-8 on page 35 for details of the data set for each key.
354 – 361	Data for key 13
362 – 369	Data for key 21
370 – 377	Data for key 29
378 – 385	Data for key 37
386 – 393	Data for key 45
394 – 401	Data for key 53
402 – 449	Data for key 61
410 – 417	Data for key 6
418 – 425	Data for key 14
426 – 433	Data for key 22

Table 6-2. Debug Output Data Frame (Continued)

Frame Byte #	Description
434 – 441	Data for key 30
442 – 449	Data for key 38
450 – 457	Data for key 46
458 – 465	Data for key 54
466 – 473	Data for key 62
474 – 481	Data for key 7
482 – 489	Data for key 15
490 – 497	Data for key 23
498 – 505	Data for key 31
506 – 513	Data for key 39
514 – 521	Data for key 47
522 – 529	Data for key 55
530 – 537	Data for key 63. See Table 4-8 on page 35 for details of the data set for each key.

Appendix C. Conducted Noise Immunity

Electrically conducted noise can increase the noise on the touch signals considerably and can lead to both false detects and missed touches. There is a specific test for conducted immunity, as part of typical EMC testing, which injects noise into the device under test across a broad range of frequencies and with significant amplitude. This test is designed to test the immunity of devices and products against environmental noise that is generated by commercial radio transmitters and other sources. Passing this test can be a challenge and might appear daunting, but with good design practise from the outset coupled with fine tuning of the QT2640 frequency hopping setups and, designs based on the QT2640 show excellent noise immunity and can pass the conducted noise test by some margin.

From the outset, the design must target achieving the best possible touch sensitivity while minimizing noise. A clean design can achieve excellent signal delta on touch, but its easy to destroy this quality by pushing the overall product requirements too far and poor attention to detail during the design. The more imperfections that exist in a design, the more the sensitivity will be eroded and the conducted immunity performance with it.

A good target for signal delta on touch is 100 or even 150 counts.

Best touch performance is achieved with thinner overlay panels, optimum key electrode design, clean tracking between the QT2640 and the keys with low stray capacitance between X and Y traces and low stray capacitance from X and Y traces to ground, coupled with appropriate selection of the external matrix components. See the documents referred to in “Associated Documents” on page 71 for further details on best practice for designing a touch sensor interface.

The following paragraphs make some recommendations for initial values or type for the additional matrix components to accompany a QT2640 design together with some setups the have been found to produce good noise immunity results.

Cs - NPO/COG

Use COG type for the charge sample capacitors (Cs0 –Cs7). This type of capacitor exhibits excellent stability, albeit at a higher cost, and is preferred over X7R and other types. Lower values can be used to reduce the cost or help with availability provided the charge transfer is not too high such as to saturate Cs. An excellent layout with low stray capacitance will typically allow Cs to be reduced as low as 1 nF.

Rs – 1 MΩ

Increasing the digital conversion ramp resistors (Rs0 – Rs7) increases sensitivity. The optimum selection for Rs is one that balances highest achievable sensitivity against conversion time and other undesirable side effects such as increased noise and possibly some reduction in temperature stability. Any increase in noise is typically insignificant compared to conducted noise observed during conducted immunity EMC testing.

Rs can be increased as high as 5 MΩ, although this is probably extreme in most cases. A good initial value is 1M ohm. This value increases sensitivity to a good level with acceptable additional signal noise and often achieves the best compromise in Signal-to-Noise Ratio (SNR). The increased sensitivity allows a higher detect threshold to be employed, preventing noise spikes exceeding the threshold and thus preventing false detects.

Ry - 47 kΩ

Immunity to conducted electrical noise can be increased considerably by increasing the series resistors in the matrix Y lines (Ry0 - Ry7) at the expense of moisture tolerance. There is a practical limit to how far their value can be increased because higher values must be accompanied by longer dwell times, possibly in excess of the longest settings available in the device. In any case excessively long dwell times would slow the response time intolerably.

DWELL

Increasing the Ry values alone results in a reduction in charge transfer and sensitivity. This can be recovered by increasing the charge transfer time, or dwell time, through higher DWELL settings. To achieve the optimum DWELL setting, start with the maximum value and observe the reference values. Then reduce the dwell, continue to observe the reference values, and choose the lowest DWELL setting where the reference values are not significantly reduced from their maximum.

Longer dwell times also result in reduced moisture tolerance, and so a careful balance might be necessary between these conflicting requirements.

NTHR & THRM

Together NTHR (and PTHR) and THRM allow the threshold to be set. The threshold should be set as high as possible to prevent false detects but should not be set so high that true touches are not detected. Atmel QmBtn or Hawkeye software can be used to observe the signals and signal delta on touch, and used to determine the optimum threshold settings for a specific design.

The threshold will typically need setting at a lower value to accommodate noise than is evident when testing in the absence of noise. This may initially seem counter intuitive until consideration is given to the fact that heavy noise can cause undesirable detect dropouts as well as missed touch events. The threshold should be set as high as possible, but not so high that the noise causes detect dropouts while the key is still touched.

Frequency Hopping – FHM = 3 (sweep)

The immunity of QT2640 based designs to higher noise amplitude can be increased by configuring the frequency hopping with three different frequencies and using FHM = 2. Choosing the three frequencies is a process of trial and error and varies from design to design, but a considerable increase in tolerable noise amplitude can be achieved once three appropriate frequencies have been identified.

However, frequency hopping using sweep mode (FHM = 3) is much easier to configure and may provide sufficient immunity for many applications and products.

Associated Documents

The following documents are a good source of general information regarding design and development of Atmel touch sensors and should be studied before starting the development of a new touch interface.

- **QTAN0079 - Buttons, Sliders and Wheels Sensor Design Guide**
  - Refer to this application note for details of different possible X/Y electrode designs and patterns, and the optimum geometry to match the panel thickness.
- **QTAN0062 - QTouch and QMatrix Sensitivity Tuning for Keys, Sliders and Wheels**
- **AVR3000 QTouch Conducted Immunity**
  - Refer to this guide for further information on immunity from conducted noise.

These documents are available on the Atmel website (www.atmel.com).

Revision History

Revision No.	History
Revision AX – April 2012	• Datasheet, first draft
Revision BX – January 2013	• Datasheet updated for memory mapped serial interface Updated schematic • Updated pinout drawing • Corrected and included all setups • Corrected Debug Appendix • Added details for sliders/wheels • Added details for VREF/WS pin • Added setups block summary • Other minor text corrections
Revision CX – April 2013	• Replaced Figure 2-6
Revision DX – November 2013	• Amended specifications in section 6.5 and 6.6 • Updated part number in section 6.7 • Other minor changes

Atmel | Enabling Unlimited Possibilities®

Atmel Corporation
1600 Technology Drive
San Jose, CA 95110
USA
Tel: (+1) (408) 441-0311
Fax: (+1) (408) 487-2600
www.atmel.com

Atmel Asia Limited
Unit 01-5 & 16, 19F
BEA Tower, Millennium City 5
418 Kwun Tong Roa
Kwun Tong, Kowloon
HONG KONG
Tel: (+852) 2245-6100
Fax: (+852) 2722-1369

Atmel München GmbH
Business Campus
Parkring 4
D-85748 Garching bei München
GERMANY
Tel: (+49) 89-31970-0
Fax: (+49) 89-3194621

Atmel Japan G.K.
16F Shin-Osaki Kangyo Bldg
1-6-4 Osaki, Shinagawa-ku
Tokyo 141-0032
JAPAN
Tel: (+81) (3) 6417-0300
Fax: (+81) (3) 6417-0370

© 2013 Atmel Corporation. All rights reserved. / Rev.: 9684DX–AT42–12/13

Atmel®, Atmel logo and combinations thereof, AKS®, Adjacent Key Suppression®, Enabling Unlimited Possibilities®,and others are registered trademarks, QmBtn™, QT™ and others are trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be registered trademarks or trademarks of others.

Disclaimer: The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. EXCEPT AS SET FORTH IN THE ATMEL TERMS AND CONDITIONS OF SALES LOCATED ON THE ATMEL WEBSITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS AND PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and products descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.