Chipsmall

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

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

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

Contact us

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

Atmel AT42QT1245

24-key QMatrix FMEA IEC/EN/UL60730 Touch Sensor

PRELIMINARY DATASHEET

Features

- Number of keys:
  - Up to 24
- Technology:
  - Patented charge-transfer (transverse mode), with frequency hopping
- Key outline sizes:
  - 6 mm × 6 mm or larger (panel thickness dependent); different sizes and shapes possible
- Key spacings:
  - 8 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)
- Panel thickness:
  - Up to 50 mm glass, 20 mm plastic (key size dependent)
- Key sensitivity:
  - Individually settable via simple commands over communication interface
- Interface:
  - SPI
- Signal processing:
  - Self-calibration, auto drift compensation, noise filtering, Adjacent Key Suppression® (AKS®) technology
- 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:
  - +3 V to 5 V
- Package:
  - 32-pin 7 × 7 mm TQFP RoHS compliant
  - 32-pin 5 × 5 mm QFN RoHS compliant

1. Pinout and Schematic

1.1 Pinout Configuration
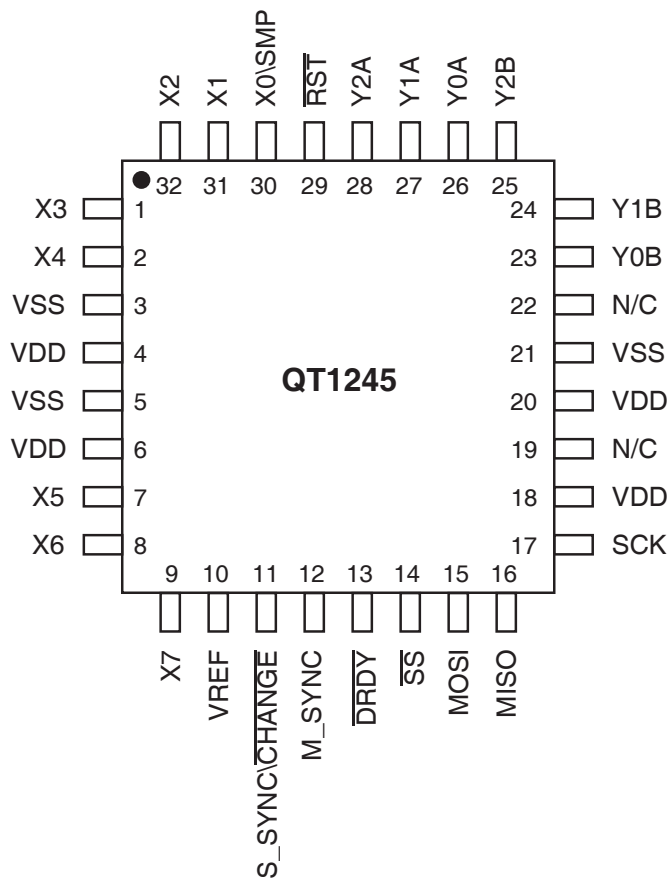
1.2 Pin Descriptions

Table 1-1. Pin Descriptions

Pin	Name	Type	Comments	If Unused, connect To...
1	X3	O	X matrix drive line	Leave open
2	X4	O	X matrix drive line	Leave open
3	VSS	P	Ground	–
4	VDD	P	Power	–
5	VSS	P	Ground	–
6	VDD	P	Power	–
7	X5	O	X matrix drive line	Leave open
8	X6	O	X matrix drive line	Leave open

Table 1-1. Pin Descriptions (Continued)

Pin	Name	Type	Comments	If Unused, connect To...
9	X7	O	X matrix drive line	Leave open
10	VREF	I	Connect to Vss	–
11	S_SYNC/$\overline{\text{CHANGE}}$	OD	Oscilloscope sync / Key touch change, active low. Has internal 20 kΩ – 50 kΩ pull-up resistor	Leave open
12	M_SYNC	I	Mains Sync input	Vdd
13	$\overline{\text{DRDY}}$	OD	It is essential to use this pin. 1 = SPI Comms ready. Need 40 µs grace period before checking. Has internal 20 kΩ – 50 kΩ pull-up resistor	–
14	$\overline{\text{SS}}$	I	SPI slave select; has internal 20 kΩ – 50 kΩ pull-up resistor	–
15	MOSI	I	SPI data input	–
16	MISO	O	SPI data output	–
17	SCK	I	SPI clock input	–
18	VDD	P	Power	–
19	N/C	I/O	Do not connect	Leave open
20	VDD	P	Power	–
21	VSS	P	Ground	–
22	N/C	I/O	Do not connect	Leave open
23	Y0B	I/O	Y line connection	Leave open
24	Y1B	I/O	Y line connection	Leave open
25	Y2B	I/O	Y line connection	Leave open
26	Y0A	I/O	Y line connection	Leave open
27	Y1A	I/O	Y line connection	Leave open
28	Y2A	I/O	Y line connection	Leave open
29	$\overline{\text{RST}}$	I	Reset low; has internal 30 kΩ – 60 kΩ pull-up resistor. This pin should be controlled by the host.	Leave open or Vdd
30	X0/SMP	O	X matrix drive line / Sample output	–
31	X1	O	X matrix drive line	Leave open
32	X2	O	X matrix drive line	Leave open

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

1.3   Schematic

Figure 1-1.   Typical Circuit

Follow regulator manufacturer's recommended values for input and output bypass capacitors Creg.
Keep the output capacitor close to pins 4 and 6.
Add a 100 nF capacitor close to pin 18.

* optional - for emission suppression
** optional - for RF susceptibility improvement

Note:
1) It is important to place all Cs, Rs, Rx and Ry components physically near to the chip.

2) Leave YnA, YnB unconnected if not used.

For component values in Figure 1-1 check the following sections:

- Section 3.3 on page 6: Cs capacitors (Cs0 – Cs2)
- Section 3.5 on page 8: Sample resistors (Rs0 – Rs2)
- Section 3.7 on page 8: Matrix resistors (Rx0 – Rx7, Ry0 – Ry2)
- Section 3.10 on page 11: Voltage levels

2. Overview

The AT42QT1245 (QT1245) device is a digital burst mode charge-transfer (QT™) sensor, designed specifically for matrix 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 rear of the PCB 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 QT1245 employs transverse charge-transfer (QT) sensing, a technology that senses changes in electrical charge forced across two electrode elements by a pulse edge (see Figure 3.).

Figure 2-1. Field Flow Between X and Y Elements
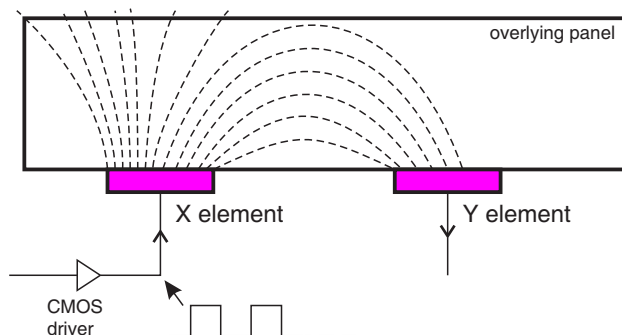
The QT1245 allows a wide range of key sizes and shapes to be mixed together in a single touch panel (see Section 3.8 on page 10).

The device uses an SPI interface to allow key data to be extracted and to permit individual key parameter setup. This interface uses a memory mapped structure, designed to minimize the amount of data traffic while maximizing the amount of information conveyed.

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

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

3. Hardware and Functional

3.1 Matrix Scan Sequence

The circuit operates by scanning each key sequentially. The keys are numbered from 0 – 23. Key scanning begins with location X = 0, Y = 0 (key 0). All keys on Y0 are scanned first, then Y1 and finishing with all keys on Y2.

For example, the sequence is:

X0Y0, X1Y0 to X7Y0

X0Y1, X1Y1 to X7Y1

X0Y2, X1Y2 to X7Y2

Table 3-1 shows the key numbering.

Table 3-1. 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	

Each key is sampled in a burst of acquisition pulses whose length is determined by the setups parameter BL (Section 7. on page 46). This can be set on a per-key basis. A burst is completed entirely before the next key is sampled. At the end of each burst the resulting signal is converted to digital form and processed. The burst length directly impacts key gain. Each key can have a unique burst length in order to allow tailoring of key sensitivity on a key-by-key basis.

3.2 Burst Paring

Keys that are disabled by setting NDIL to zero (Section 6.2.1 on page 31) are eliminated from the scan sequence to save scan time and thus power. As a consequence, the fewer keys that are used the faster the device can respond. All calibration times are reduced when keys are disabled.

3.3 Cs Sample Capacitor Operation

Cs capacitors absorb charge from the key electrodes on the rising edge of each X pulse. On each falling edge of X, the Y matrix line is clamped to ground to allow the electrode and wiring charges to neutralize in preparation for the next pulse. With each X pulse charge accumulates on Cs causing a staircase increase in its differential voltage.

After the burst completes, the device clamps the Y line to ground causing the opposite terminal to go negative. The charge on Cs is then measured using an external resistor to ramp the negative terminal upwards until a zero crossing is achieved. The time required to zero cross becomes the measurement result.

The Cs capacitors should be connected as shown in Figure 1-1 on page 4. They should be NP0 (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. If the transverse capacitive coupling from X to Y is large enough the voltage on a Cs capacitor can saturate, destroying gain. In such cases the burst length should be reduced and/or the Cs value increased. See Section 3.4.

If a Y line is not used its corresponding Cs capacitor may be omitted and the pins left floating.

3.4   Sample Capacitor; Saturation Effects

Cs voltage saturation at a pin YnB is shown in Figure 3-1. Saturation begins to occur when the voltage at a YnB pin becomes more negative than –0.25 V at the end of the burst. 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 3-1.   VCs – Nonlinear During Burst
(Burst too long, or Cs too small, or X-Y transcapacitance too large)
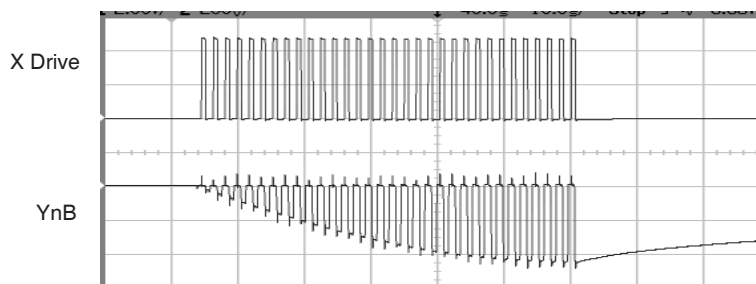

The cause of this is either from the burst length being too long, the Cs value being too small, or the X-Y transfer coupling being too large. Solutions include loosening up the key structure interleaving, greater separation of the X and Y lines on the PCB, increasing Cs, and decreasing the burst length.

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 ramps 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 3-2 shows a defective waveform similar to that of Figure 3-1, 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 3-2.   VCs – Poor Gain, Nonlinear During Burst
(Excess capacitance from Y line to Gnd)
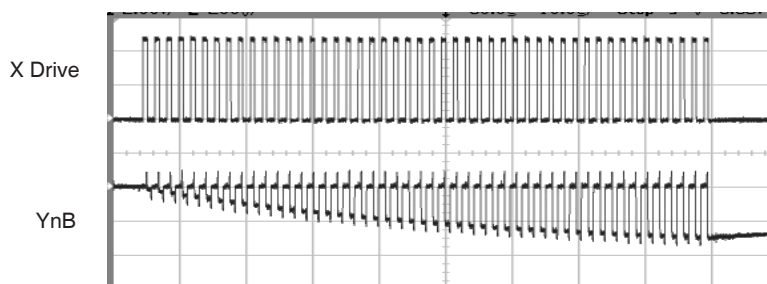
This phenomenon is more subtle. It can be best detected by increasing BL to a high count and watching what the waveform does as it descends towards and below –0.25 V. The waveform appears deceptively straight, but it slowly starts to flatten even before the –0.25 V level is reached.

A correct waveform is shown in Figure 3-3. Note that the bottom edge of the bottom trace is substantially straight (ignoring the downward spikes).

Figure 3-3. VCs – Correct
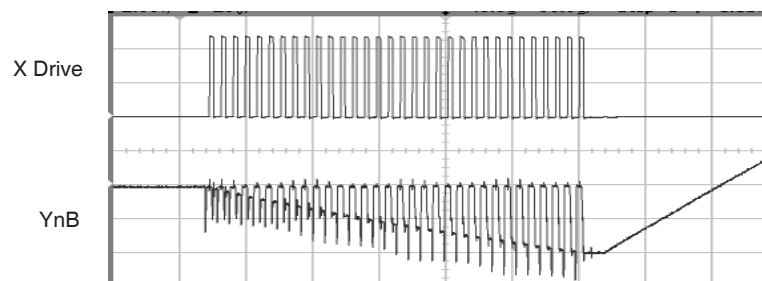
Unlike other QT circuits, the Cs capacitor values on QT1245 have no effect on conversion gain. However, they do affect conversion time.

Unused Y lines should be left open.

3.5   Sample Resistors

The sample resistors (Rs0 – Rs2) are used to perform single-slope ADC conversion of the acquired charge on each Cs capacitor. These resistors directly control acquisition gain. Larger values of Rs proportionately increase signal gain. For most applications Rs should be 1 MΩ. Unused Y lines do not require an Rs resistor.

3.6   Signal Levels

Using Atmel 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 200 – 750 counts with properly designed key shapes (see the *Touch Sensors Design Guide*, available on the Atmel website) and values of Rs. 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 web site.

The signal swing from the smallest finger touch should preferably exceed 8 counts, with 12 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 does increasing the sampling resistor (Rs) values.

3.7   Matrix Series Resistors

The X and Y matrix scan lines can use series resistors (Rx0 – Rx7 and Ry0 – Ry2 respectively) for improved EMC performance (Figure 1-1 on page 4).

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

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. Increasing the dwell time recovers some of the signal levels lost to higher values of Rx and Ry, as shown in Figure 3-4. 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. Dwell time is adjustable (see Section 6.10.1 on page 40) to one of 8 values.

Figure 3-4. Drive Pulse Roll-off and Dwell Time
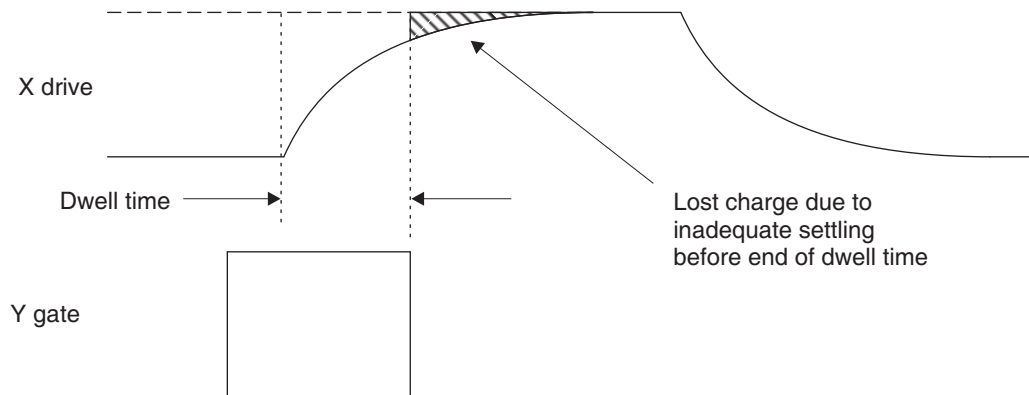
Dwell time problems can also be solved by either reducing the stray capacitance on the X line(s) (by a layout change, for example by reducing X line exposure to nearby ground planes or traces), or, the Rx resistor needs to be reduced in value (or a combination of both approaches).

One way to determine X settling time is to monitor the fields using a patch of metal foil or a small coin over the key (Figure 3-5). Only one key along a particular X line needs to be observed, as each of the keys along that X line will be identical. The chosen dwell time should exceed the observed 95% settling of the X-pulse by 25% or more.

Figure 3-5. Probing X-Drive Waveforms With a Coin
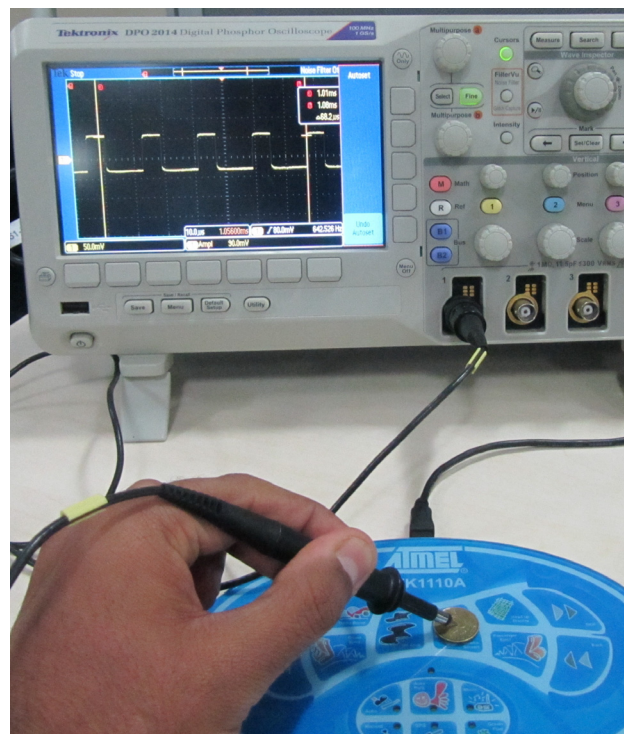
3.8 Key Design

For information about key design refer to the *Touch Sensors Design Guide* on the Atmel website.

3.9 PCB Layout, Construction

3.9.1 Overview

The chip should be placed 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 antennae. 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.

Note: 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 and LED drive lines) 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 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.

3.9.2 LED Traces and Other Switching Signals

Digital switching signals near the Y lines induces transients into the acquired signals, deteriorating the SNR performance of the device. 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 to suppress capacitive coupling effects which can induce false signal shifts. The bypass capacitor does not need to be next to the LED, in fact it can be quite distant. The bypass capacitor is noncritical and can be of any type.

LED terminals which are constantly connected to Vss or Vdd do not need further bypassing.

3.9.3 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 the no-clean flux changes. 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.

3.10 Power Supply Considerations

For the power supply range see Section 8.2 on page 48. If the power supply fluctuates slowly with temperature, the device tracks and compensates for these changes automatically with only minor changes in sensitivity. If the supply voltage drifts or shifts quickly, the drift compensation mechanism is not able to keep up, causing sensitivity anomalies or false detections.

As this device uses the power supply itself 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 that is not also 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.

Caution: A regulator IC shared with other logic can result in erratic operation and is not advised.

A regulator can be shared among two or more devices on one board.

A single ceramic 0.1 µF bypass capacitor, with short traces, should be placed very close to each supply pin of the IC. Failure to do so can result in device oscillation, high current consumption and erratic operation.

3.11 Startup/Calibration Times

The device employs a rigorous initialization and self-check sequence for EN 60730 compliance. The last step in this sequence enables the serial communication interface, but only if the self-checks pass. The communication interface is not enabled if a safety critical fault is detected during the startup sequence. A maximum of 95 ms is required from reset before the device is ready to communicate.

The device 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 device 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 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 device can report during this interval that the key(s) affected are still in calibration via status function bits. Table 3-2 shows keyscan cycles times and calibration times per key versus dwell time and burst length for all 24 keys enabled. The values given assume that MSYNC = off, SLEEP = off, FHM = off .

Table 3-2.   Keyscan Cycle and Calibration Times

Setups	Keyscan Cycle Time	Calibration Time (min)	Calibration Time (max)
BL = 0 (16 pulses), DWELL = 0 (125 ns) FREQ0 = 0 Signal level = 100 counts	7 ms	77 ms (11 × 7)	434 ms (62 × 7)
BL = 3 (64 pulses), DWELL = 7 (4.5 µs) FREQ0 = 63 Signal level = 4000 counts	105 ms	1155 ms (11 × 105)	6510 ms (62 × 105)

Keyscan cycle and thus calibration time varies with different setups as well as keypad design. Increasing capacitive loading at each key increases the signal levels and hence increases the keyscan cycle time. Conversely, disabled keys are subtracted from the burst sequence and thus the keyscan cycle time is shortened. The keyscan cycle time should be measured on an oscilloscope for the specific setups in use.

The initialization time and calibration time must be added together to determine the total time from reset to a keys ability to report detection.

3.12   Reset Input

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

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 essential. Where the QT1245 has detected a failure of one of the internal EN60730 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 $\overline{RST}$ pin.

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

3.13   Frequency Hopping

The QT1245 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 (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.

3.14   Detection Integrators

See also Section 6.2.1 on page 31.

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 QT1245 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 attempt 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 is settable 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.

3.15 Sleep

If the sleep feature is enabled (see Section 7. on page 46), the device sleeps whenever possible to conserve power. Periodically, it wakes automatically, scans the matrix, and returns to sleep unless there is activity which demands further attention. The device returns to sleep automatically once all activity has ceased. The time for which it sleeps before automatically awakening can be configured.

A new communication with the device while it is asleep causes it to wake up, service the communication and scan the matrix. 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 demands further attention. If there has been recent activity, the part performs another complete matrix scan and then attempts to sleep once again. This process is repeated indefinitely until the activity stops and the part returns to sleep.

Key touch activity prevents the part from sleeping. The part will not sleep while any key is calibrating or if any touch events were detected at any key in the most recent scan of the key matrix, or while a serial communication is in progress.

If the sleep feature is disabled in the setups, the device never sleeps. Sleep should be disabled if the device is being used in an FMEA or EN 60730 compliant design because all operations are stopped within the device while the part is asleep and the host is not able to distinguish between faulty operation and EN 60730 counters appearing to run slow because the part is intermittently sleeping. It should also be noted that the drift compensation interval will effectively be stretched each time the device sleeps because the device is fully halted during sleep and cannot perform drift compensation. If sleep is enabled, a shorter drift compensation interval may be required.

3.16 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 device failure.

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

All FMEA tests are repeated every few seconds or faster during normal run operation, if the sleep feature is disabled. Sometimes FMEA errors can occur intermittently (for example, due to momentary power fluctuations). It is advisable to confirm a true FMEA fault condition by making sure the error flags persist for several seconds.

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

The FMEA tests performed allow detection of fault conditions including the following:

- 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 shorts and open circuits
- Vref
- Key gain (see Section 6.9.2 on page 39 (Key Gain Test Threshold, KGTT))

Other tests incorporated into the device include:

- A test for signal levels against a preset minimum value (LSL setup, see Section 7. on page 46). If any signal level falls below this level, an error flag is generated.
- CRC communications checks on all data read from the device.

Some very small key designs have very low X-Y coupling. In these cases, the amount of signal is very small, and the key gain is low. As a result, small keys can fail the LSL test see (Section 6.9.1 on page 39) or the FMEA key gain test. In such cases, the burst length of the key should be increased so that the key gain increases. Failing that, a small ceramic capacitor, for example 3 pF, can be added between the X and Y lines serving the key to artificially boost signal strength. This capacitor must be located at the key's site to ensure the FMEA tests are not compromised.

3.17 IEC/EN 60730 Compliance

The device also incorporates special test features which, together with the FMEA tests, allow products to achieve IEC/EN60730 compliance with ease.

IEC/EN60730 compliance demands dynamic verification of all safety related components and sub-components within a product. The QT1245 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 QT1245 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 themselves by checking and validating the communications CRC, which the QT1245 computes during each read transfer sequence (see Section 5.5 on page 25).

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 QT1245 is able to verify sub-components internally, but any such verification fails, the device disables serial communication and locks up in an infinite loop. The host can detect this condition as repeated serial communication CRC failures.

During normal operation the host must perform regular reads of the IEC/EN60730 counters (see Section 5.4 on page 24) to verify correct operation of the QT1245. The host must also perform regular reads of the device status (see Section 5.5 on page 25) and verify there are no errors reported. The FMEA error flag, LSL error flag and setups CRC error flag must all be considered as part of an IEC/EN60730 compliant design.

The host can try to recover from any safety critical failure by resetting the QT1245 using its $\overline{RST}$ pin. The host should allow a grace period in consideration of the startup and initialization time the QT1245 requires after reset to communicate (see Section 3.11 on page 11).

The sub-components that the QT1245 is able to verify internally are tested repeatedly during the normal running of the device, with 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 3-3.

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

Test	Required Matrix Scans to complete test
FMEA	24
Other	18
Variable Memory	768
Firmware CRC	342
Setups CRC	5

Table 3-4 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 3-4.   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), SLEEP = 0 (sleep disabled).	7.5
BL = 3 (64 pulses), DWELL = 7 (4.5 µs), FREQ0 = 25, All keys enabled, FHM = 0, MSYNC = 0 (off), SLEEP = 0 (sleep disabled).	38

Longer matrix scan times are possible than those shown in Table 3-4 by using even higher values for FREQ0 (lower burst frequencies), but these are considered extreme settings.

Table 3-5 shows the failure detect times for the internal tests assuming a matrix scan time of 9 ms, which is valid for typical Setups.

Table 3-5. Failure Detect Time

Test	Failure Detect Time (ms)
FMEA	216
Other	162
Variable Memory	6912
Firmware CRC	3078
Setups CRC	45
Conditions: Matrix scan time = 9 ms. QT1245 does not sleep for duration of tests.	

Longer failure detect times are possible than those shown in Table 3-5 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 3-5 accordingly. Alternatively, the failure detect times can be calculated by taking the numbers from Table 3-5 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.

3.17.1 UL approval / VDE compliance

The QT1245 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.

4. Serial Communications

4.1 Introduction

The device uses an SPI interface for communications with a host.

4.2 SPI Interface

The SPI host device always initiates communications sequences. This is intentional for FMEA and IEC/EN60730 purposes so that the host always has total control over the communications with the QT1245. Even return data is controlled by the host. The QT1245 employs a CRC on return data to provide for robust communications.

There is an essential $\overline{\text{DRDY}}$ line that handshakes transmissions. This is needed by the host from the QT1245 to ensure that transmissions are not sent when the QT1245 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 QT1245 communications time-out (30 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 19 hex (25 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:	1.5 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 QT1245 clocks data in on the rising edge. The QT1245 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: Important – SCK must idle high, it should never float.

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

$\overline{\text{DRDY}}$: Data Ready, active-high, indicates to the host that the QT1245 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 $\overline{\text{DRDY}}$ timing.

Serial communications pacing is controlled by $\overline{\text{DRDY}}$. Use of $\overline{\text{DRDY}}$ is critical to successful communications with the QT1245. The host is permitted to perform an SPI transfer only when $\overline{\text{DRDY}}$ has returned high. After each SPI byte transfer $\overline{\text{DRDY}}$ goes low after a short delay and remains low until the QT1245 is ready for another transfer. A short delay occurs before $\overline{\text{DRDY}}$ is driven low because the QT1245 may be otherwise busy and requires a finite time to respond. $\overline{\text{DRDY}}$ may go low only for a few microseconds. During the period from the end of one transfer until $\overline{\text{DRDY}}$ goes low and back high again, the host should not perform another transfer. Therefore, before each byte transmission the host should first check that $\overline{\text{DRDY}}$ is high again.

Figure 4-1. SPI Slave-only Mode Timing
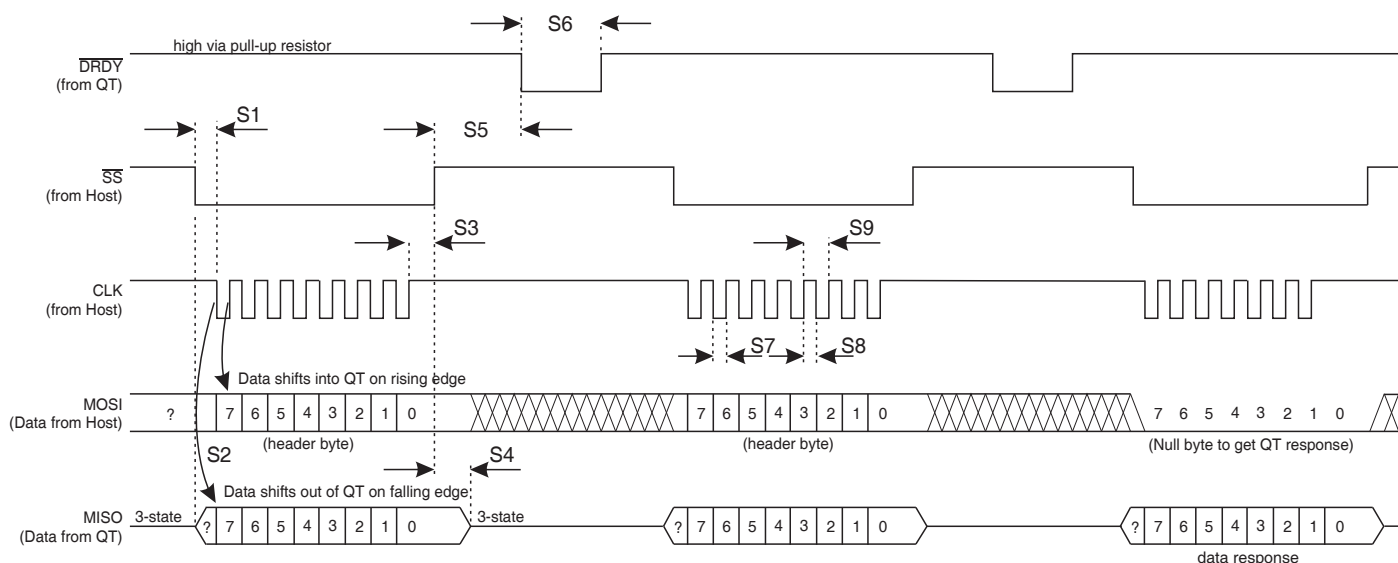
S1: > 333 ns [1]	S2: ≤ 20 ns	S3: ≥ 25 ns	S4: ≤ 20 ns	
S5: ≤ 40 μs	S6: ≤ 1 μs	S7: ≥ 333 ns	S8: ≥ 333 ns	S9: ≥ 667 ns

1. Or > 3 μs if sleep is enabled

If the host wants to perform a byte transfer with the QT1245 it should behave as follows:

1. Wait at least 40 μs after the previous SPI transfer (time S5 in Figure 4-1: $\overline{\text{DRDY}}$ is guaranteed to go low before this 40 μs expires).
2. Wait until $\overline{\text{DRDY}}$ is high (it may already be high again).
3. Perform the next SPI transfer with the QT1245.

The time it takes for $\overline{\text{DRDY}}$ 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: ≤4.5 ms

Read: ≤1 ms

The $\overline{\text{DRDY}}$ 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.

Other $\overline{\text{DRDY}}$ specifications:

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

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

Null Bytes: When the QT1245 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 QT1245.

Timeout: A successful communications sequence consists of a number of byte transfers. The QT1245 expects each byte transfer within a sequence to occur within 30 ms (± 5 ms) of the previous transfer. If more than 30 ms elapses between any two bytes, the QT1245 abandons the current sequence and starts a new sequence at the next byte transfer.

Wakeup: If a communication starts when the device is sleeping (see Section 6.5.1 on page 36), the device will be woken, triggered by the falling edge at $\overline{\text{SS}}$, but will not be fully awake and ready to receive the communication for up to 3 µs. If sleep is enabled, allow 3 µs from the $\overline{\text{SS}}$ falling edge to the first clock (S1 in Figure 4-1 on page 18).

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.

4.3 $\overline{\text{CHANGE}}$ Pin

The $\overline{\text{CHANGE}}$ pin can be used to alert the host to key touches or key releases, 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 state (either touch or touch release) 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 three key state addresses to ensure the host has a complete picture of which keys 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.

Every key can be individually configured to wake a host microcontroller upon a touch change. Therefore, a product can wake from sleep when any key state changes, or only when certain desired keys change state. The configuration is set in the setups block (see Appendix 6.2.3 on page 33) on a key-by-key basis.

IEC/EN60730 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 QT1245 Detect Status bytes (Addresses 6, 7, 8), but at a rate suitable to guarantee IEC/EN60730 compliance. A poll rate of once every 100 ms would impose very little extra load on the QT1245.

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

5. Memory Map

5.1 Introduction

The device features a set of commands which are used for control and status reporting.

As well as Table 5-1 refer to for further details.

Table 5-1.   Memory Map

Address	Use	Access
0	Reserved	Read
1	Reserved	Read
2	100 ms counter (IEC/EN60730)	Read
3	Signal fail counter (IEC/EN60730)	Read
4	Matrix Scan counter (IEC/EN60730)	Read
5	Device Status. Collection of bit flags	Read
6	Detect status for keys 0 to 7, one bit per key	Read
7	Detect status for keys 8 to 15, one bit per key	Read
8	Detect status for keys 16 to 23, one bit per key	Read
9	Reserved	Read
10	Current frequency	Read
11	Current pulse spacing	Read
12 – 15	Data for key 0. See Table 5-6 on page 27 for details.	Read
16 – 19	Data for key 1	Read
20 – 23	Data for key 2	Read
24 – 27	Data for key 3	Read
28 – 31	Data for key 4	Read
32 – 35	Data for key 5	Read
36 – 39	Data for key 6	Read
40 – 43	Data for key 7	Read
44 – 47	Data for key 8	Read
48 – 51	Data for key 9	Read
52 – 55	Data for key 10	Read
56 – 59	Data for key 11	Read
60 – 63	Data for key 12	Read
64 – 67	Data for key 13	Read
68 – 71	Data for key 14	Read
72 – 75	Data for key 15	Read

Table 5-1. Memory Map (Continued)

Address	Use	Access
76 – 79	Data for key 16	Read
80 – 83	Data for key 17	Read
84 – 87	Data for key 18	Read
88 – 91	Data for key 19	Read
92 – 95	Data for key 20	Read
96 – 99	Data for key 21	Read
100 – 103	Data for key 22	Read
104 – 107	Data for key 23	Read
108 – 139	Reserved	
140	Control command. Write `0xFF` to calibrate all keys. Write `0xFE` immediately before writing setups. Write 0xFD to perform low level calibration and offset for frequency hopping. Write k to calibrate key k. Write `0x18` to reset the device.	Write
141 – 250	Setups – see Section 6. on page 29 for details	Read/Write

Poll rate: The host can make use of the $\overline{\text{CHANGE}}$ pin output to initiate a communication. This guarantees the optimal polling rate.

If the host cannot make use of the $\overline{\text{CHANGE}}$ pin the poll rate in normal *run* operation should be no faster than once per matrix scan (see Section 8.4 on page 49). Typically 10 to 20 ms is more than fast enough to extract the key status. Anything faster will not provide new information and slows down the chip operation.

Run Poll Sequence: In normal run mode the host should limit traffic with a minimalist control structure. The host should just read the IEC/EN60730 counters, the Device Status and the three detect status registers (see Figure 5-1 on page 22).

Figure 5-1. Power-on or Hardware Reset Flow Chart
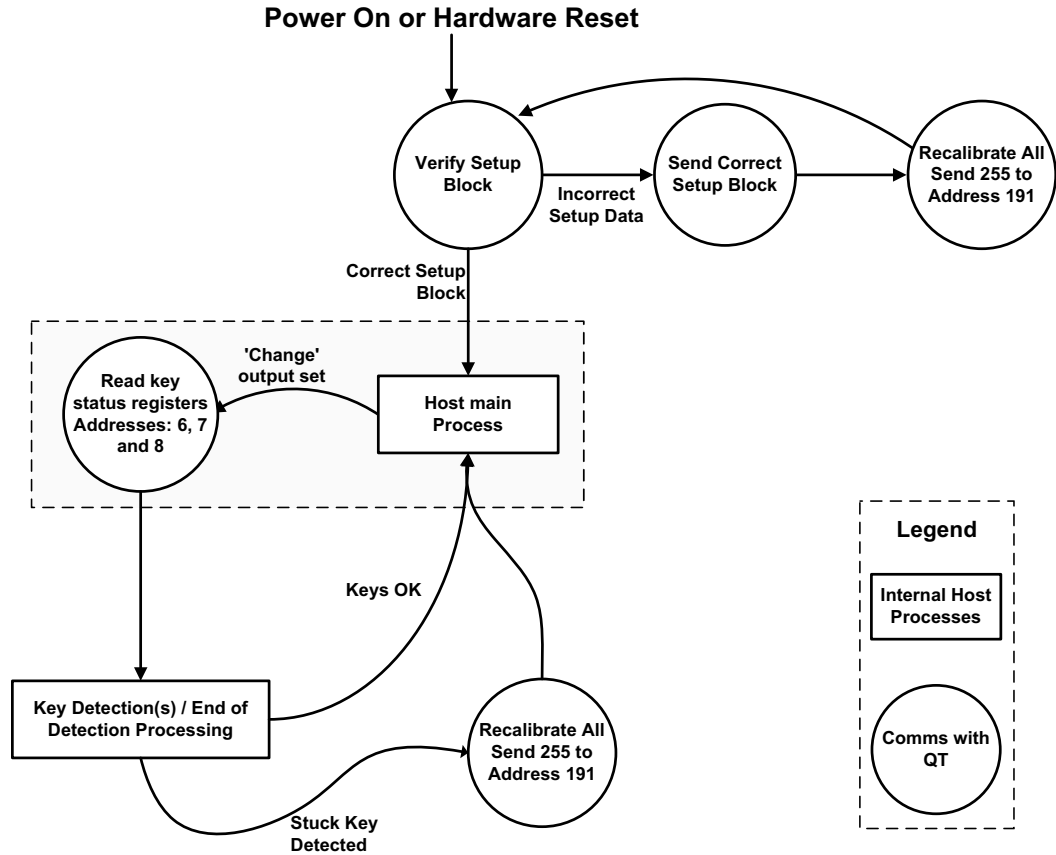
5.2 Writing Data to the Device

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

Byte #	1		2		3	
MOSI	MemAddress	D	n	D	Data	D
MISO	U		U		U	

Table 5-2. Key to Write Sequence

Byte #	Name	Description
1	MemAddress	Target memory address within QT1245
	U	Undefined data byte, should be ignored
	D	40 µs delay then wait until $\overline{DRDY}$ high
2	n	# bytes to write. MSB = 0 (write). Valid range is 1 – 111.
3+	Data	Data byte(s) written to QT1245

The host initiates a write sequence by sending the internal memory address it wishes to write followed by n, the number of QT1245 addresses it wishes to write to. The host then sends one or more data bytes. Between each byte transfer, the host must follow the $\overline{\text{DRDY}}$ handshake procedure (wait 40 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.

Byte 2 is split into two fields. The MSB indicates a read or write operation and should be set to 0 to perform a write. Bits 6 – 0 define the number of data bytes the host wishes to write to the device.

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 QT1245 are undefined and should be ignored.

5.3   Reading Data From the Device

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

Byte #	1		2		3		4		..				
MOSI	MemAddress	D	128 + n	D	NULL	D	NULL	D	..	NULL	D	NULL	D
MISO	U		U		Data 1		Data 2		..	Data n		CRC LSB	

(continued) ... NULL | D | NULL | D with CRC LSB / CRC MSB

Table 5-3.   Key to Read Sequence

Byte #	Name	Description
1	MemAddress	Target memory address within QT1245
	U	Undefined data byte, should be ignored
	D	40 µs delay then wait until $\overline{\text{DRDY}}$ high
2	n	Bit 7 = 1 (read). Bits 6 – 0 = # bytes to read, valid range is 1 – 127. CRC is available after reading n bytes.
	NULL	Placeholder, send from the host to recover data from the QT1245
3+	Data	Data returned from QT1245

The host initiates a read sequence by sending the internal memory address it wishes to read followed by 128+n, where n is the number of QT1245 addresses it wishes to read from. The host then sends NULL bytes to recover the n data bytes from the QT1245. If the host specifies n to be greater than 1, the data bytes are read from consecutive memory addresses. The device automatically increments the target memory address after reading each location. When all n data bytes have been returned, the device returns a 16-bit CRC, LSB first. Between each byte transfer, the host must follow the $\overline{\text{DRDY}}$ handshake procedure (wait 40 µs and then wait until $\overline{\text{DRDY}}$ is high).

Byte 2 is split into two fields. The MSB indicates a read or write operation and should be set to 1 to perform a read. Bits 6 – 0 define the number of data bytes the host wishes to read from the device.

The device calculates the double-word (16-bit) CRC using the internal memory start address, the number of bytes to be read (n), and the n data bytes themselves, all in the same sequence they occur during the transmission (see Appendix A. on page 54).

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 QT1245 while the host is sending the address and n are undefined and should be ignored.

5.4   Report IEC/EN60730 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/EN60730. IEC/EN60730 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/EN60730)

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 seconds (256 × 100 ms = 25.6s) 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 QT1245 or the host, and the host should adopt an appropriate strategy to meet the required safety standard.

Address 3: Signal fail counter (IEC/EN60730)

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 affected and determine a maximum acceptable count rate for this counter.

Address 4: Matrix scan counter (IEC/EN60730)

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 QT1245 or the host, and the host should adopt an appropriate strategy to meet the required safety standard.

5.5   Report Device Status

Address 5: Device status

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

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

Bit 4: Set if an FMEA error was detected during operation. See Section 3.16 on page 13.

Bit 3: Reports that an enabled key has a very low reference value, lower than the user-configurable LSL value (see Section 7. on page 46).

Bit 2: Set if any key is in the process of calibrating.

Bit 1: Set if there was a mains sync error, for example there was no Sync signal detected within the allotted 100 ms. See Section 7. on page 46. 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 0: 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.

A host in an IEC/EN60730 compliant product must check bits 0, 2, 3, and 4 and handle persistent errors appropriately to maintain safety.