Features

- Configurations:
  - Can be configured as a combination of touchscreen, sliders/wheels and keys, with Adjacent Key Suppression™ (AKS™) technology between groups
- QField™ Touchscreen:
  - Two-touch capable with independent XY tracking for one or two concurrent touches in real time, with touch size reporting
  - Up to eight-inch diagonal screen size supported
  - 1024 x 1024 resolution
- Discrete Keys:
  - Up to 48 (subject to other configurations)
- QSlide™/QWheel™:
  - Configurable up to six independent sliders/wheels
- Linearity:
  - Screen design dependent but typically better than ±1 percent
- Filtering:
  - Advanced digital filtering (user configurable)
- Response Times:
  - Sub 15 ms possible, depending on filter settings
- General Purpose Outputs (GPOs):
  - Up to four user controllable outputs
- Technology:
  - Patented charge-transfer (transverse mode)
- Panel Thickness:
  - Glass up to 5 mm, screen size dependent
  - Plastic up to 3 mm, screen size dependent
- Channel Sensitivity:
  - Individually settable via simple commands over serial interface
- Interface:
  - I^2C-compatible slave mode, 100 kHz or 400 kHz with 2.7V or greater Vdd
- Power:
  - 1.8V to 5.5V (2.7V to 5.5V in high speed mode)
- Packages:
  - 44-pin 7 x 7 mm MLF RoHS compliant
  - 44-pin 10 x 10 mm TQFP RoHS compliant
  - 49-ball 5 x 5 mm BGA RoHS compliant
  - 44-pin 5 x 5 mm two-row staggered QFN RoHS compliant
- Signal Processing:
  - Self-calibration, auto drift compensation, noise filtering, Adjacent Key Suppression technology

QTwo™ 10-bit Touchscreen Controller

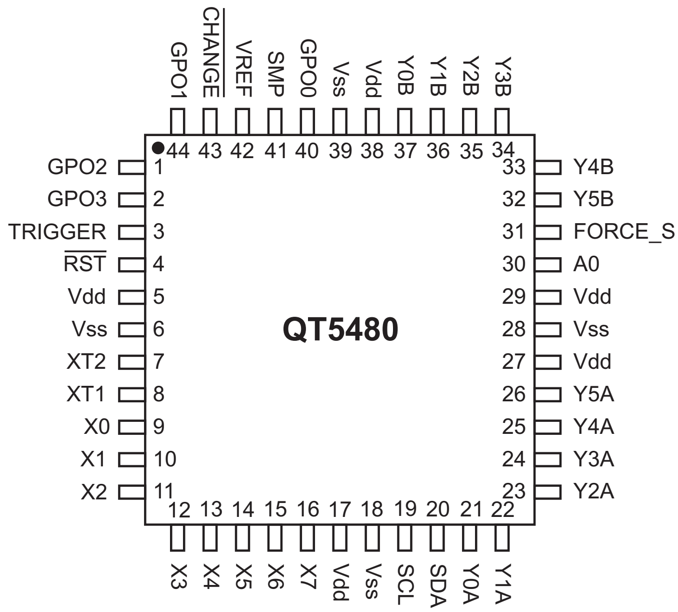
AT42QT5480

9510CX–AT42–12/08

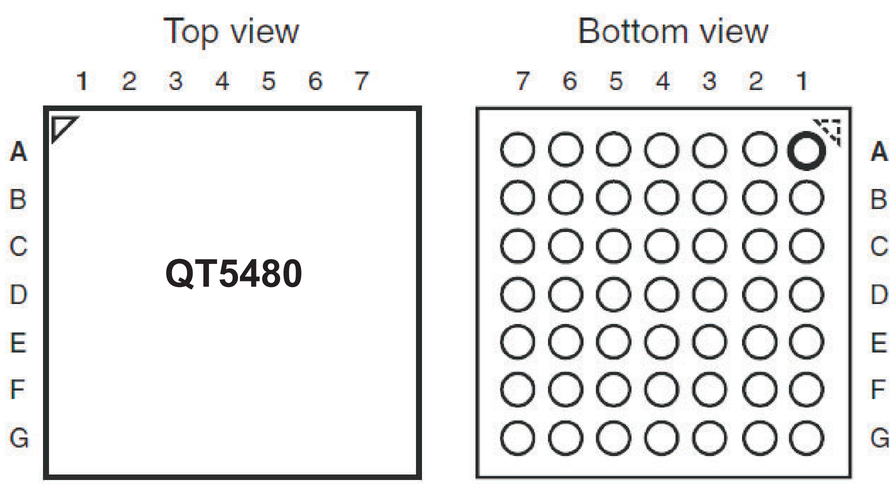
1. Pinout and Schematic

1.1 Pinout Configurations

1.1.1 MLF/TQFP/Two-row Staggered QFN



1.1.2 BGA



1.2 Pin Descriptions

Table 1-2. Pin Listing

Pin	Ball	Name	Type	Comments	If Unused, Connect To...
1	B2	GPO2	O	General purpose output 2	Leave open
2	B1	GPO3	O	General purpose output 3	Leave open
3	C3	TRIGGER	I	Trigger input (active low)	Vss
4	C2	$\overline{\text{RST}}$	I	Reset low; has internal 30k - 60k pull-up	Leave open or Vdd
5	A5	Vdd	P	Power	–
6	A1	Vss	P	Supply ground	–
7	D2	XT2	X	Clock resonator	–
8	E1	XT1	X	Clock resonator	–
9	D3	X0	O	X matrix drive line	Leave open
10	E2	X1	O	X matrix drive line	Leave open
11	F1	X2	O	X matrix drive line	Leave open
12	F2	X3	O	X matrix drive line	Leave open
13	G2	X4	O	X matrix drive line	Leave open
14	E3	X5	O	X matrix drive line	Leave open
15	F3	X6	O	X matrix drive line	Leave open
16	E4	X7	O	X matrix drive line	Leave open
17	C1	Vdd	P	Power	–
18	A4	Vss	P	Supply ground	–
19	F4	SCL	OD	Serial Interface Clock	–
20	G5	SDA	OD	Serial Interface Data	–
21	F5	Y0A	I	Y line connection	Leave open
22	G6	Y1A	I	Y line connection	Leave open
23	F6	Y2A	I	Y line connection	Leave open
24	E5	Y3A	I	Y line connection	Leave open
25	F7	Y4A	I	Y line connection	Leave open
26	E6	Y5A	I	Y line connection	Leave open
27	E7	Vdd	P	Power	–
28	A7	Vss	P	Supply ground	–
29	C7	Vdd	P	Power	–
30	D6	A0	I	I2C-compatible address select	–
31	C6	FORCE_S	I	Force sensor input	Vdd or Vss
32	B7	Y5B	I	Y line connection	Leave open
33	D5	Y4B	I	Y line connection	Leave open
34	B6	Y3B	I	Y line connection	Leave open
35	A6	Y2B	I	Y line connection	Leave open

Table 1-2. Pin Listing (Continued)

Pin	Ball	Name	Type	Comments	If Unused, Connect To...
36	C5	Y1B	I	Y line connection	Leave open
37	B5	Y0B	I	Y line connection	Leave open
38	G3	Vdd	P	Power	–
39	D1	Vss	P	Supply ground	–
40	B4	GPO0	O	General purpose output 0	Leave open
41	C4	SMP	O	Sample output.	–
42	A3	Vref	I	Reference input	–
43	B3	$\overline{\text{CHANGE}}$	OD	State change notification	–
44	A2	GPO1	O	General purpose output 1	Leave open
–	D4	Vss	P	Supply ground	–
–	D7	Vss	P	Supply ground	–
–	G1	Vss	P	Supply ground	–
–	G4	Vss	P	Supply ground	–
–	G7	Vss	P	Supply ground	–

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

1.3 Schematic

Figure 1-1. Typical Circuit

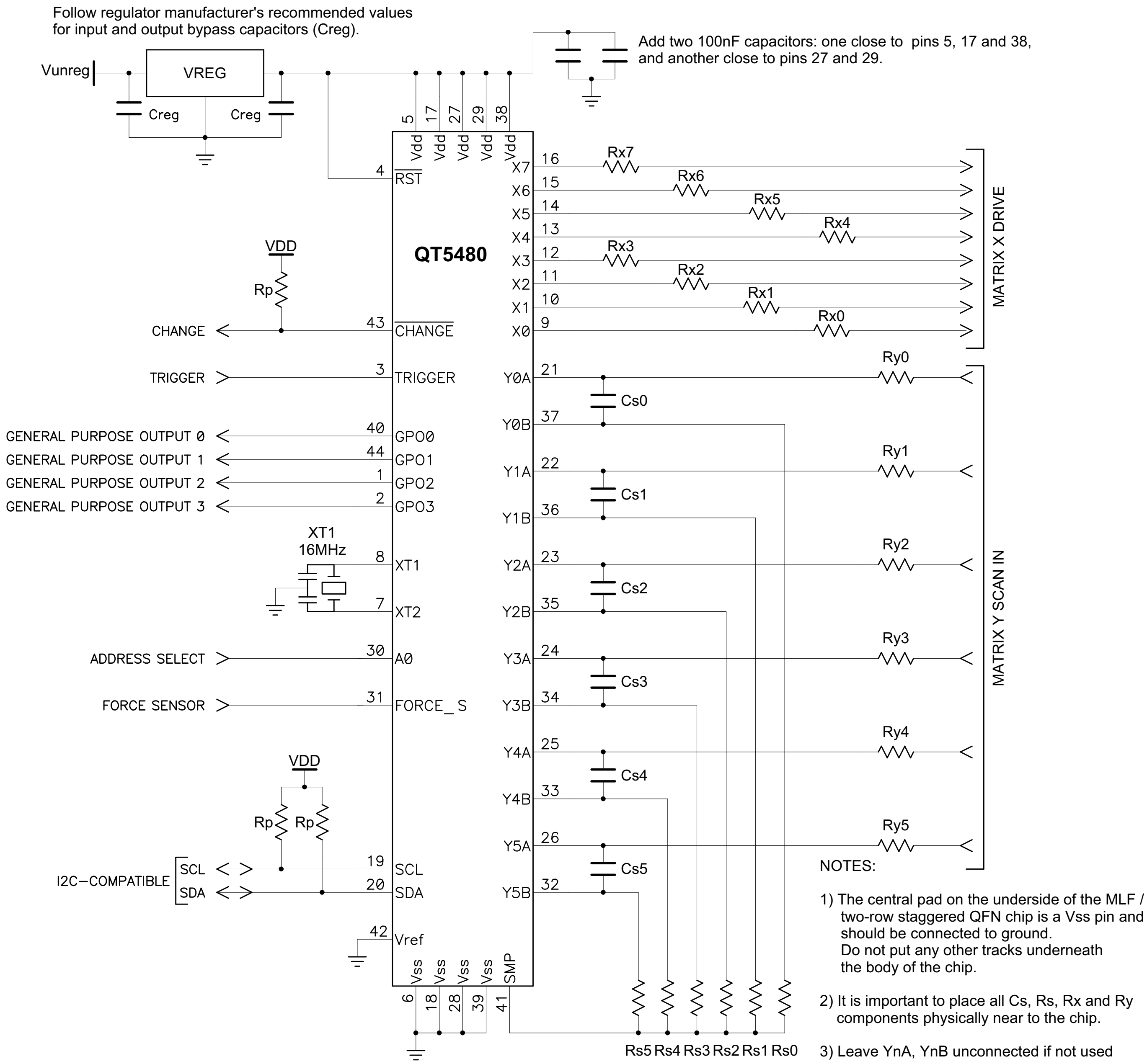
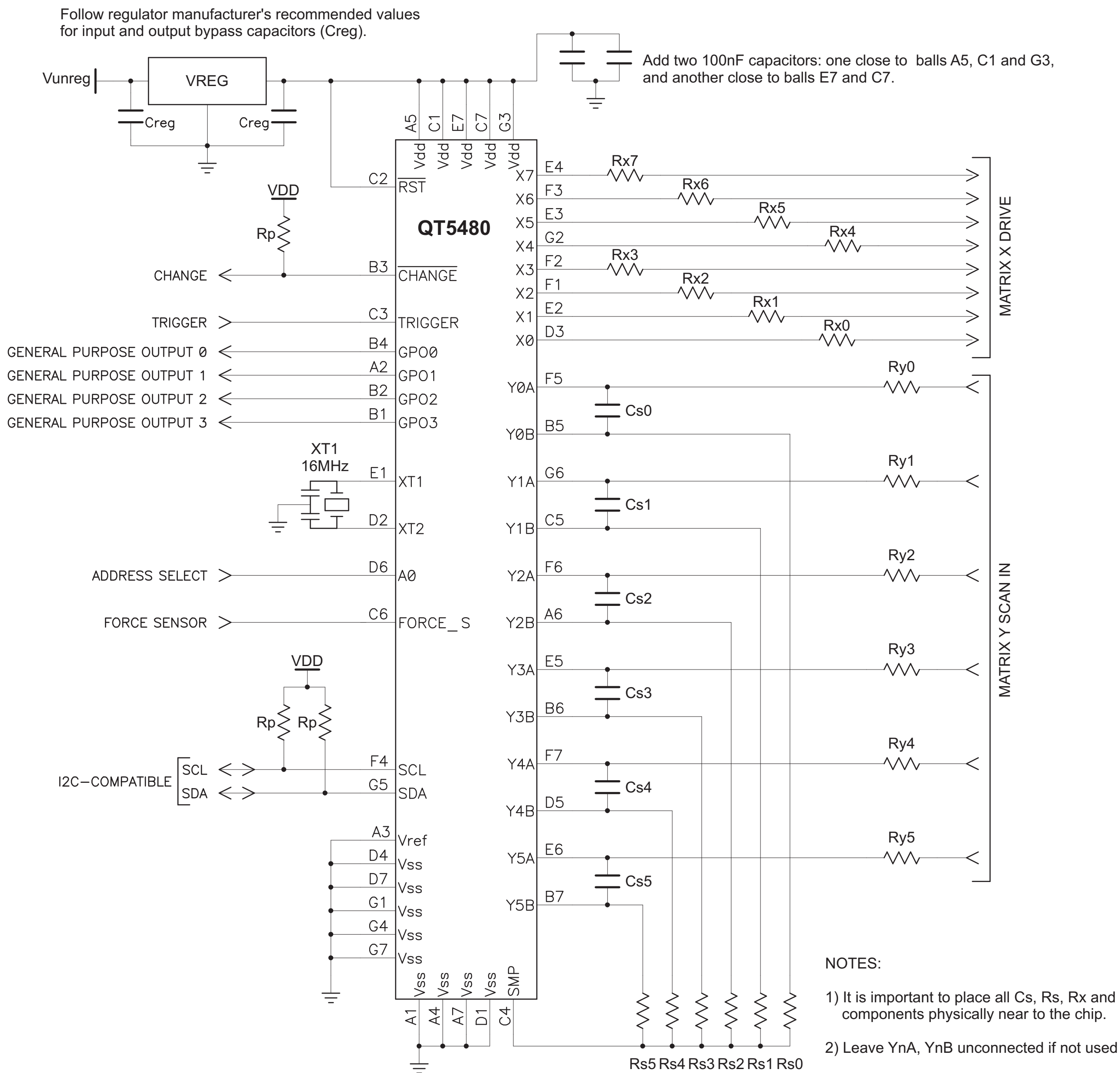


Figure 1-2. Typical Circuit – BGA



Suggested regulator manufacturers:

- Torex (XC6215 series)
- Seiko (S817 series)
- BCDSemi (AP2121 series)

Re Figure 1-1 on page 5 and Figure 1-2 check the following sections for component values:

- Section 5.12.1 on page 21: Cs capacitors (Cs0 – Cs5)
- Section 5.12.2 on page 22: Sample resistors (Rs0 – Rs5)
- Section 5.12.5 on page 22: Series resistors Ry0 – Ry5 and Rx0 – Rx7
- Section 5.12.4 on page 22: Resonator XT1
- Section 8.2 on page 51: Voltage levels

- Section 5.3.5 on page 13: SDA, SCL pull-up resistors
- Section B.8.3 on page 68: LED traces

2. Overview of the QT5480

2.1 Introduction

The QT5480 is a versatile capacitive touchscreen controller, able to support a diagonal touchscreen of up to 8 inches. The IC supports Two Touch™ operation, part of the QTwo™ family of devices from Atmel®.

The QT5480 uses Atmel's patented QMatrix™ capacitive sensing technique, which offers excellent moisture tolerance, fast acquisition and outstanding ground load immunity.

A unique feature of the QT5480 is that it allows a choice to be made as to how many of the capacitive measurement channels form part of a touchscreen, and which ones form discrete keys or sliders.

This controller offers unrivalled flexibility to create touchscreens, sliders and keys. The device can report two touches on a touchscreen making it suitable for next generation touch interfaces. Concurrent use of a touchscreen plus keys or sliders is also possible.

By treating all capacitive channels equally during measurement, and then applying additional signal processing, the device allows the channels to be used as part of a touchscreen, or part of one or more sliders, or as discrete touch keys.

Touchscreens can be created that are of arbitrary channel length and width. Channels not used in the touchscreen can either be turned into sliders or keys. There are some constraints on the starting channels for touchscreens and sliders, but these have no practical impact for most applications.

The controller also has the ability to save a Y line when configuring a touchscreen, reusing it in the touchscreen pattern at the two edges. This saved Y line can then be used to create extra objects like a slider or multiple keys, while allowing the touchscreen to be sized as though it was one Y line larger. In this wrapped Y line mode Two Touch processing cannot be used.

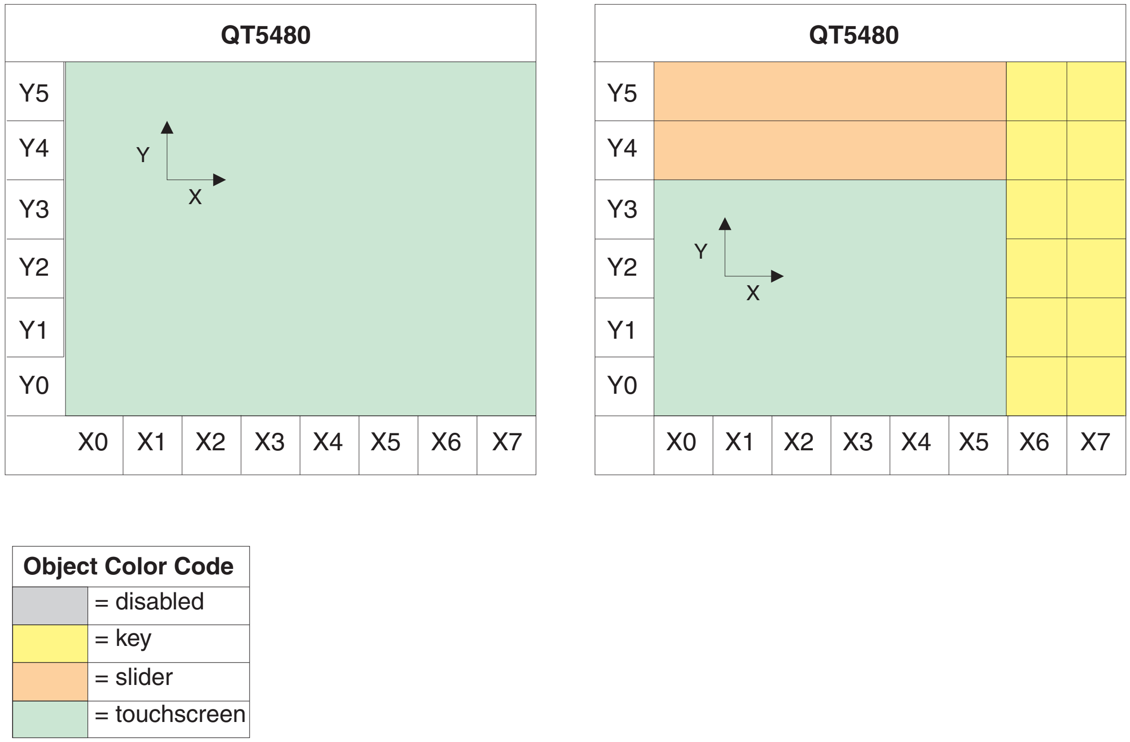
See Figure 2-1 for configuration examples.

2.2 Understanding Unfamiliar Concepts

If some of the concepts mentioned in this datasheet are unfamiliar, see the following sections for more information:

- Appendix A on page 62 for a glossary of terms
- Appendix B on page 64 for QMatrix technology
- Appendix C on page 71 for I^{2}C-compatible operation
- Appendix D on page 74 for touchscreen linearization
- Appendix E on page 76 for the checksum calculation

Figure 2-1. Example Touchscreen Configurations



3. Touchscreen Basics

3.1 Sensor Construction

A touchscreen is usually constructed from a number of transparent electrodes, typically on a glass or plastic substrate, or can also be made using non-transparent electrodes e.g. copper or carbon. Electrodes are normally formed by etching a material called Indium Tin Oxide (ITO). This is a brittle ceramic material, of high optical clarity and varying resistivity. Thicker ITO yields lower resistivity (perhaps tens to hundreds of Ω/square) at the expense of reduced optical clarity. Lower resistances are generally more compatible with capacitive sensing. Thinner ITO leads to higher resistivity (perhaps hundreds to thousands of Ω/square) with some of the best optical characteristics.

Long thin features, like interconnecting tracks, formed in ITO, can inhibit the capacitive sensing function due to the excessive RC time constants formed between the resistance of the track and the capacitance of the electrode to ground. In such cases, ITO tracks should be replaced by screen printed conductive inks (non-transparent) outside of the viewing area of the touchscreen.

A range of trade-offs also exist with regard to the number of layers used for construction. Atmel has pioneered single-layer ITO capacitive touchscreens and for many applications these offer a near optimum cost/performance balance. With a single layer screen, the electrodes are all connected using ITO out to the edges of the sensor. From there the connection is picked up with printed silver tracks. Sometimes two overprinted silver tracking layers are used to reduce the margins between the edge of the substrate and the active area of the sensor.

Two-layer designs can have a strong technical appeal where ultra-narrow edge margins are required or where the capacitive sensing function needs to have a very precise cut-off as a touch is moved to just off the active sensor area. With a two-layer design the QMatrix transmitter electrodes are normally placed nearest the bottom and the receiver electrodes nearest the top. The separation between layers can range from hundreds of nanometers to hundreds of microns, with the right electrode design and considerations of the sensing environment.

3.2 Electrode Configuration

The specific electrode designs used in Atmel's touchscreens are the subject of various patents and patent applications. Further information is available on request.

Two methods are used to determine X and Y coordinates. In this datasheet it should be assumed that the long axis of the touchscreen is referred to as the X direction and the short axis is the Y direction. The Y direction is resolved by electrodes that are spatially interpolated, whereas the X direction is resolved using electrodes that are resistively interpolated. Some of the figures in this datasheet refer to the resistors used for this interpolation in the X axis.

The QT5480 supports 8 X lines and 6 Y lines i.e. 48 channels.

3.3 Scanning Sequence

All channels are scanned in sequence by the QT5480. However, there is some parallelism in the scanning sequence to improve overall response time. The channels are scanned by measuring capacitive changes at the intersections formed between X0 and Y0 – Yn, then the intersections between X1 and Y0 – Yn etc. until all X and Y combinations have been measured.

Due to the ability to configure the device in various ways, it is possible to disable some channels so that they are not scanned at all. This can be used to improve overall scanning time.

3.4 Touchscreen Sensitivity

3.4.1 Adjustment

Sensitivity of touchscreens can vary across the extents of the electrode pattern due to natural differences in the parasitics of the interconnections, control chip etc. Another important factor in the uniformity of sensitivity is the electrode design itself. It is a natural consequence of a touchscreen pattern that the edges form a discontinuity and hence tend to have a different sensitivity; the electrodes at the far edges do not have a neighboring electrode on one side and this affects the electric field distribution in that region.

QMatrix technology has the unique feature that every channel’s sensitivity can be individually adjusted by controlling the number of charge transfer pulses used to measure their capacitance. This allows a coarse adjustment of the screen's sensitivity.

A second sensitivity adjustment is available for each channel. This adjustment is a more basic algorithmic threshold that defines when a channel is considered to have enough signal change to qualify as being in detect.

Between these two adjustment methods, the best sensitivity uniformity can be achieved.

3.4.2 Mechanical Stackup

The mechanical stackup refers to the arrangement of material layers that exist above and below a touchscreen. The arrangement of the touchscreen in relation to other parts of the mechanical stackup has an effect on the overall sensitivity of the screen. QMatrix technology has an excellent ability to operate in the presence of ground planes close to the sensor. QMatrix sensitivity is attributed more to the interaction of the electric fields between the transmitting (X) and receiving (Y) electrodes than to the surface area of these electrodes. For this reason, stray capacitance on the X or Y electrodes does not strongly reduce sensitivity.

It is possible to construct an ITO touchscreen that includes a ground shield layer on the back. This can have advantages where noisy LCDs are encountered or where mechanical movement could be encountered on the front panel to which the touchscreen is mounted (so changing the distance and hence the loading caused by the LCD). This latter point is very important; any changes in the distance between the sensor and a nearby ground, i.e. an LCD, causes changes in the capacitive signals, which in turn lead to shifts in the coordinates reported by the chip. Normally it is possible to use sufficiently stiff front panels to avoid this, but it should be considered early in the design phase.

Front panel dielectric material has a direct bearing on sensitivity. Plastic front panels are usually suitable up to about 3 mm, and glass up to around 5 – 6 mm (dependent upon the screen size and layout). The thicker the front panel, the lower the signal-to-noise ratio of the measured capacitive changes and hence the lower the resolution of the touchscreen. In general, glass front panels are near optimal because they conduct electric fields almost twice as easily as plastic panels.

4. QMatrix Primer

See Appendix B on page 64 for QMatrix information.

5. Detailed Operation

5.1 Power-up/Reset

There is an internal Power-on Reset (POR) in the device. After power-up, the device takes 10 ms before it is ready to start communications. In order to effect a proper POR, Vdd must drop to below 0.6V. See Section 8. on page 51 for further specifications.

The device also has a $\overline{\text{RST}}$ pin that, when asserted, returns the device to its reset state. The $\overline{\text{RST}}$ pin must be asserted low for at least 10 µs to cause a reset. After releasing the $\overline{\text{RST}}$ pin the device takes 10 – 15 ms before it is ready to start communications. The $\overline{\text{RST}}$ pin includes an internal Vdd pull-up resistor of 30 – 60 kΩ and so can be left unconnected if not required. Generally, it is recommended connecting the $\overline{\text{RST}}$ pin to a host controller to allow it to initiate a full hardware reset without requiring a power-down.

Note: If using level shifters to run the host controller at a lower Vdd than the QT chip, the internal pull-up resistor cannot be disabled and so may cause a parasitic-powering effect to the host.

A software reset command can be used to reset the chip via the I^2C-compatible interface (see Section 6.2.4 on page 28). A software reset takes ~30 ms.

After the chip has finished it asserts the $\overline{\text{CHANGE}}$ line to signal to the host that it is ready. The reset flag is set in the status register to show the host that it has just completed a reset cycle. This bit can be used by the host to detect any unexpected brownout events and so allow it to take any necessary corrective actions, such as reconfiguration.

On detecting the $\overline{\text{CHANGE}}$ assertion the host reads the chip, triggering the initialization to continue. During this read the chip may clock stretch for up to 10 ms.

The device requires an initialization time of 6 matrix scans at the minimum cycle time before it commences normal scanning of the capacitive channels.

5.2 Calibration

Calibration is the process by which the sensor chip assesses the background capacitance on each channel. Channels are only calibrated on power-up and when:

- the channel is enabled

  AND

  – held in detect for longer than the negative recalibration delay (NRD), as configured by the user

  OR

  – the signal delta on a channel is at least three-quarters of the detection threshold but in the anti-touch direction, while no other touches are present on the channel matrix

  OR

  – the user issues a recalibrate command

A status message is generated on the start and completion of a calibration.

5.3 Communications

5.3.1 Communications Protocol

See Appendix C on page 71 for details of the I^2C-compatible protocol.

The chip's higher level protocol is based on an event-driven communication system that generates data packets indicating which areas of status memory have changed. The change line going active signifies that a new data packet is available. A mask register allows events to be selectively enabled and disabled.

The chip is not designed to be polled, as it only presents data packets when internal changes have occurred.

5.3.2 I^2C-compatible Addresses

Four preset I^2C-compatible addresses are selectable through pin A0 (see Table 5-1). This is sampled once at power-up so if the address is changed it requires a reset or restart to enable the new address. The address input is a multilevel analog input which divides the power supply range into four equal bands to give the four possible I^2C-compatible addresses.

Table 5-1. I^2C-compatible Address Selection

A0	I^2C Address	I^2C Mode/Speed	Vdd to be Used	Address Selection (A0)
0 ≤ 0.25 Vdd	0x20	100 kHz	1.8 – 5.5V	Vss
0.25 ≤ 0.5 Vdd	0x30	100 kHz/400 kHz	2.7 – 5.5V	30kΩ to Vss, 50kΩ to Vdd
0.5 ≤ 0.75 Vdd	0x40	100 kHz/400 kHz	2.7 – 5.5V	50kΩ to Vss, 30kΩ to Vdd
0.75 ≤ 1.0 Vdd	0x11	100 kHz	1.8 – 5.5V	Vdd

5.3.3 Reading/Writing Setup and Status

Write: A WRITE cycle to the device consists of a START condition followed by the I^2C-compatible address of the device. The next two bytes are the address of the location into which the writing starts; the first byte is the Least Significant Byte (LSByte) of the address, the second byte is the Most Significant Byte (MSByte). This address is then stored as the address pointer.

Subsequent bytes in a multibyte transfer are written to the location of the address pointer, location of the address pointer +1, location of the address pointer +2 etc. This ends with the STOP condition on the I^2C-compatible bus. A new WRITE cycle involves sending another address pointer.

Read: Forced memory reads for debugging are done by writing the most significant 8 bits of the 10-bit address to the read address pointer. For example, for addresses 0x80, 0x81, 0x82 and 0x83, write 0x20 to the read address pointer (these are the most significant 8 bits of the desired address). This generates a data packet containing the data found at the requested address. The address value sent with the data packet matches the value written to the read address pointer.

Note: After a forced read is requested, the next data packet generated may not be the requested data if other packets are waiting to be read (e.g. touchscreen position messages) but the requested data will always eventually be delivered.

5.3.4 Data Packets

A data packet consists of five bytes. All reads should be a multibyte transfer of five bytes of data. Data reads are simple multibyte reads with no address pointer required. The data packet contains information about the location of the data. The first byte is the most significant 8 bits of the 10-bit address of the first data byte. This is followed by the four data bytes found at, and after, that location.

There is no need for the host to ask for specific data. Whenever a status byte changes, a data packet containing this information is automatically generated. For example, a data packet with address 4 indicates a change in the status bytes 16 – 19. This is a touchscreen position change.

The host should keep a copy of the address map status area (addresses 0 – 35), which it updates using the data packets from the QT5480.

Note: Each write or read cycle must end with a STOP condition. The QT5480 may not respond correctly if a cycle is terminated by a new START condition.

5.3.5 SDA, SCL

The I^2C-compatible bus transmits data and clock with SDA and SCL, respectively. These are open-drain; that is, I^2C-compatible master and slave devices can only drive these lines low or leave them open. The termination resistors (Rp) pull the line up to Vdd if no I^2C-compatible device is pulling it down.

The termination resistors commonly range from 1 kΩ to 10 kΩ and should be chosen so that the rise times on SDA and SCL meet the I^2C-compatible specifications (see Section 8.5 on page 52).

5.3.6 $\overline{\text{CHANGE}}$ Line

The $\overline{\text{CHANGE}}$ line is an output that is used to alert the host that a new data packet is available, thus reducing the need for wasteful I^2C-compatible communications. The host should not read the data packets at any other time. The change line is deasserted during the data packet and is asserted again at the end of the packet read if another message is available.

If the chip is read when the $\overline{\text{CHANGE}}$ line is not asserted, it returns a packet with the header byte 0xFF, indicating that the packet data is invalid.

5.3.7 Clock Stretching

Clock stretching is used by the device during normal communications with its host. This is in accordance with the I^2C-compatible specification.

The device additionally has an internal monitor that resets its I^2C-compatible hardware if either line is held low, without the other line changing, for more than 100 ms in low speed (100 kHz) mode, or 50 ms in high speed (400 kHz) mode. It is important that no other device on the bus clock stretches for 100 ms+ in low speed mode, or 50 ms+ in high speed mode, otherwise the monitor resets the I^2C-compatible hardware, and transfers with the chip may be corrupted.

5.4 Operational Modes

The QT5480 operates on a combination of two fixed cycle times. There is one measurement set per cycle. When no channels are touched, the cycle time is given by the LP mode setting. Every cycle, one measurement set is made and the device then sleeps for the remainder of the cycle.

If a channel is touched, the cycle changes to the minimum cycle time setting for a faster response. It remains in this mode until the awake timeout has expired after the last touch.

The low power (LP) mode, awake timeout and minimum cycle time are all user-selectable. Normally the LP mode is set to be much slower than the minimum cycle time, to conserve power. If the measurement takes longer than either of the cycle times, an overflow flag is set in the general status byte.

Note: If the trigger input is low (active) and enabled, it forces the device to run in minimum cycle mode (see Section 6.2.21 on page 34).

A setting of 0 for the LP mode and minimum cycle time causes the device to enter an ultra-low power mode (SLEEP), where no measurements are carried out. SLEEP mode also stops the internal watchdog timer, so that the part is dormant, and current drain is <2 µA. The QT5480 wakes from SLEEP mode if there is an address match on the I^2C-compatible bus, a hardware reset on the $\overline{RST}$ pin or an LP mode is set.

5.5 Objects

5.5.1 Introduction

Groups of channels can be configured as a touchscreen or sliders/wheels to form a grouping known as an object. Objects use Y lines as their basis; a slider requires exactly one Y line, a touchscreen requires a minimum of 3 Y lines (4 Y lines for Two Touch). Objects must start at X0. A touchscreen must start at X0Y0. Sliders can start at X0Y0 – X0Y5. The chip can support up to six sliders or one touchscreen combined with a slider per unused Y line. Channels not assigned to objects are treated as touch keys.

Positional data is calculated to 10-bit resolution for a touchscreen and to 8-bit resolution for slider/wheel objects. With objects the full 8-bit or 10-bit numerical range is not usually completely obtainable as there are inevitably some offsets at each end or edge of the object.

5.5.2 Slider/Wheels

A slider can consist of 2 – 8 channels starting at X0 on any Y line but a wheel requires at least three channels. This is done using one of the slider configurations (one for each Y line). The data reported is in the slider X position location for that Y line. The data range of 0 to 255 represents the 0 – 100 percent absolute position of the touch on the slider.

5.5.3 Touchscreen

A touchscreen can use up to 6 Y lines starting from Y0 and up to 8 X lines starting from X0. This is done by setting the touchscreen setup byte to the number of Y lines used and the touchscreen length configuration byte to a number, 3 – 8. A touchscreen using more X and Y lines gives the best linearity and can be made larger.

The number of channels used to form a touchscreen is strongly dependent on the required design performance. Generally, higher channel counts give higher resolution and are more suitable for two touch operation. Further information is available upon request.

5.5.4 AKS Technology and Objects

There can be up to three Adjacent Key Suppression (AKS) groups, implemented so that only one channel in each group can be reported as being touched at any one time. A channel with a higher delta signal dominates and goes into detect first. Each channel may be in one of the groups 1 – 3, or in group 0 meaning that it is not AKS enabled.

Assigning all channels within an object to the same AKS group causes that object to behave as though it, (the object) belongs to that AKS group. For example, if a slider is in the same AKS group as a number of separate keys, then touching anywhere on the slider causes the AKS feature to suppress the keys. Similarly touching the keys first suppresses the slider.

If the AKS feature is not required then the channels should be configured within AKS group 0 as this allows faster touch response times.

5.6 Gating

5.6.1 External Inputs

The QT5480 has gating mechanisms to decide which channels are allowed to go into detect. Each channel can be selected so that it is only allowed to go into detect if one or more of the following conditions is met:

- The trigger input (TRIGGER) is active (low) and enabled
- The force sensor (FORCE_S) is activated and enabled

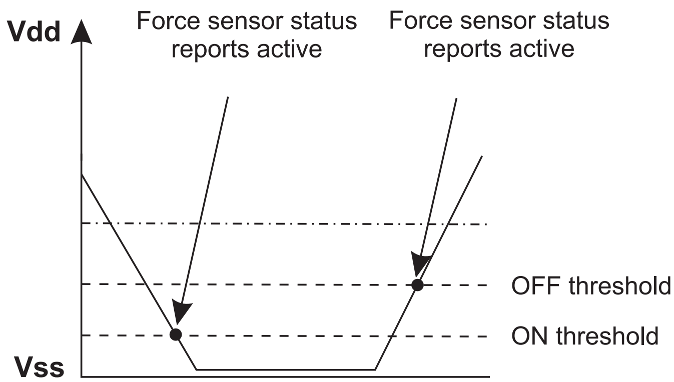
Note: Channels gated by a trigger that is not enabled can never detect.

This gating allows certain combinations of channels to be locked out and only enabled when their set of valid conditions are met.

The force sensor has a user selectable threshold level that the voltage must be less than in order to register as valid. There is also a user selectable hysteresis setting which is added to this that the voltage must exceed, to become invalid (see Figure 5-1).

The current ADC value (address 26) and a status byte indicating the gating status (address 27) can be polled by the host for debugging purposes.

Figure 5-1. Force Sensor Gate



5.6.2 Guard Channel

A guard channel is typically any channel that has been designed to wrap around the outside of several other channels being used as keys or sliders/wheels. The guard channel is normally designed to be extra sensitive so that accidental off-target touches or grips near the intended channels tend to activate the guard channel instead of, or as well as, the intended channels. Using this feature allows these false channels to be gated out. Refer to application note *QTAN0031, Avoiding False Touch Inputs*.

5.7 Signal Processing

5.7.1 Introduction

Figure 5-2 shows the parameters used in the different signal processing areas.

Figure 5-2. Signal Processing Parameters

Parameters:

Electrode Design: The electrode design affects sensitivity, noise tolerance and linearity.

Cs Capacitor Value: Higher values can reduce the effects of channel saturation but can result in larger unit-to-unit variance. Changing this value also affects the tolerance to certain system noise.
Typical Value: 4.7 – 10 nF

Burst Length: Higher values give more touch sensitivity and resolution but increase scan times and may increase risk of channel saturation. Lower values improve response time but may reduce sensitivity.
Typical Value: 32 – 64

Dwell Time: Higher values are better on heavily loaded (Resistance or Capacitance) layouts but will increase scan times and reduce system noise tolerance. Shorter dwell times improve scan time and noise tolerance but can generate undesired thermal effects if dwell time is marginal.
Typical Value: 1 µs

Parameters:

Slew Rate Filter: Higher values make the signal level able to slew faster but less able to reject noise, Lower values reject noise better but result in a slower response time.
Typical Value: 0 (off), 10 – 20 (on)

Median Filter: Higher values result in better filtering but a slightly slower response time,. Lower values improve response time but result in a lesser filtering effect.
Typical Value: 0 (off), 3 – 5 (on)

IIR Filter: Higher values result in stronger filtering but at the expensive of a longer response time. Lower values give less filtering effect but a better response time.
Typical Value: 0 (off), 1 – 3 (on)

Parameters:

Channel Threshold: Higher values are more robust against system noise but make the channel less sensitive. Lower values make the channel more sensitive but may result in over-sensitivity or false detections.
Typical Value: 10 – 20

NDIL and SDIL Settings: Higher values improve noise rejection but reduce response time. Lower values increase response time but may result in false detections at very low values due to system noise.
Typical Value: 2 – 5

AKS Group and Gating Controls: These settings determine when a channel is allowed to report as being touched. Enabling AKS technology processing on a group of channels enables the designer to create a dense keypad and ensure that only the key that a user intended to touch reports as being touched. Enabling the gating controls allows the designer to lock out detection on certain channels until a force sensor or logic input becomes active.

Parameters:

Keys: Enabled channels that are not configured within a slider or touchscreen exist as keys that report as "touched" or "not-touched". There are no additional controls for keys.

Touchscreen Length / Width: Physically larger screens or the screens that improve how close together two touches may be tracked require the use of more channels. Touchscreens using less channels may be scanned faster, improving the response time and power-consumption.

Slider/Wheel Length: Physically longer sliders/wheels require the use of more channels. Sliders/wheels using fewer channels can be scanned faster improving response time and power consumption.

Touchscreen/Slider/Wheel Positional Hysteresis: Higher settings reduce jitter on the reported position at the expense of the user having to move their touch further before a direction change is indicated.
Typical Value: 2 – 10

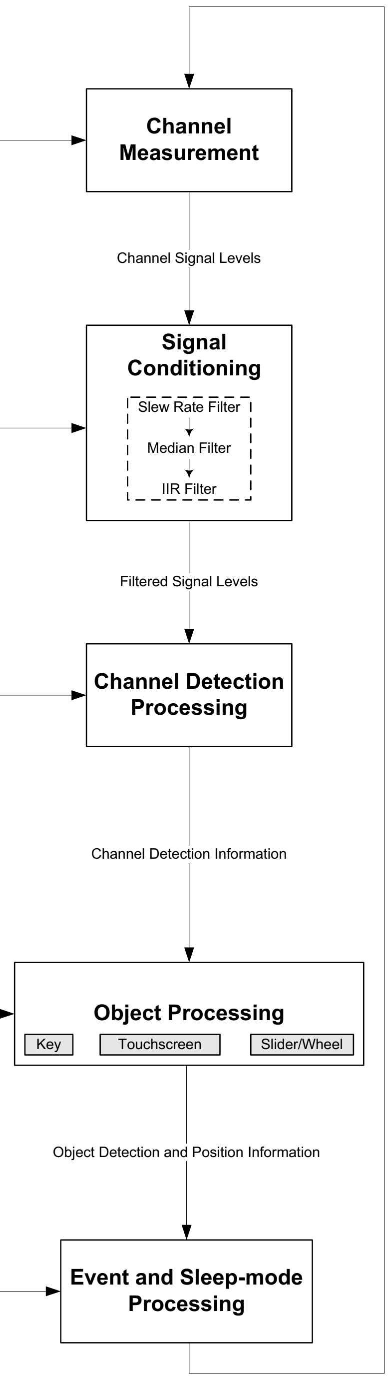
Touchscreen Two Touch Controls: Higher settings of the Two Touch hysteresis and plateau levels increase the minimum separation between independently tracked touches but reduce the possibility of flickering between two and one touches when the touches are close together.
Typical Values: Plateau: 0 – 5, Two Touch Hysteresis: 5 – 10

Parameters:

LP Mode: Higher values reduce power consumption. Lower values improve the response time to the first user touch after the last awake timeout has expired.
Typical Value: 32 – 64 ms

Minimum Cycle Time: Lower values improve the response time and increase the speed of movements that the chip can track. Higher values reduce power consumption.
Typical Value: 16 – 20 ms

Awake Timeout: Higher values improve the response time for repeated touches. Lower settings reduce power consumption.
Typical Value: 10 – 30s



5.7.2 AKS Technology

Adjacent Key Suppression (AKS) technology is a patented method used to detect which key is touched when keys are located close together. A touch in a group of AKS keys is only indicated on the key with the largest signal. This is assumed to be the intended key. AKS technology works best when it operates in conjunction with a detect integration setting of several acquisition cycles (see Section 6.2.30 on page 40).

Once a key in an AKS group is in detect, there can be no further detections on keys in that group until the key is released. By default, all channels are placed in AKS group 0 (AKS feature disabled for all channels).

AKS technology works slightly differently with objects (see Section 5.5.4 on page 14).

5.7.3 Detection Integrator

The device features a detection integration mechanism, which acts to confirm a detection in a robust fashion. A per-channel counter is incremented each time the channel has exceeded its threshold and stayed there for a number of acquisitions. When this counter reaches a preset limit the channel 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 10 acquisitions in succession without going below the threshold level, before the channel is declared to be touched. If, on any acquisition, the signal is not seen to exceed the threshold level, the counter is cleared and the process has to start from the beginning. See Section 6.2.30 on page 40 for additional information.

5.7.4 Filtering

The 5480 has three programmable signal-level filters that can be tuned to give optimum rejection of system noise. These filters consist of a slew rate filter, a median filter and an IIR (digital) filter, applied in order. See Section 6.2.41 on page 44, Section 6.2.42 on page 45 and Section 6.2.43 on page 45.

- Slew Rate Filter

  This filter is for the rejection of random spikes in the signals due to system noise sources. The coefficient for this filter configures the maximum amount that a channel signal level may be allowed to change between successive measurement cycles. It is recommended that the coefficient for this filter is set to be above the desired channel thresholds so that this filter does not interfere with normal touch detection.

- Median Filter

  This filter helps reject noisy samples; a median filter helps to reject bad samples whilst retaining a good response time. The length of the median applied may be programmed using the filter setting. Odd length settings apply a true median filter, even settings implement a filter where the two middle values are averaged together to create the filter output.

- IIR (Digital) Filter

  This filter smooths the channel signal levels and improves the resolution of position data. The filter coefficient may be adjusted using the filter control, allowing the strength of this low-pass filter to be modified, depending on the IC's cycle time and level of system noise.

- Position Filter

  The 5480 also has two position filters that are applied to the calculated slider and touchscreen positions. The first is an adaptive position filter that is applied to the slider and touchscreen positions, and the second is a hysteresis control that is applied afterwards.

The Adaptive position filter generates stable position data when the user moves their touch slowly but also allows a fast response when the user moves their finger quickly. The rate at which the filter can adapt can be modified using the position filter control setting.

- Hysteresis Filter

  The Hysteresis filter is applied last to the generated to positions to help the host distinguish between true intended user movements and small unintended movements or positional jitter. Once the user has started moving their touch the programmed hysteresis is applied so that there must be a deliberate change of direction before this is registered to the host.

5.8 Gestures

5.8.1 Introduction

The QT5480 supports the following on-chip gesture processing for each touch.

- Tap

  A tap occurs when the user quickly touches and releases the touchscreen. No significant movement takes place while the user's finger is on the touchscreen. It is characterized by a short touch duration. This could be used, for example, to activate a hyperlink on a displayed web page.

- Double Tap

  A double tap occurs when the user quickly touches and releases the touchscreen twice in quick succession. No significant movement takes place while the user's finger is on the touchscreen, or between successive touches. It is characterized by short touch durations, and a short gap between the first release and the second touch. This could be used, for example, to select a word in a displayed document.

- Press

  A press occurs when the user touches and holds the touchscreen. No significant movement takes place while the user's finger is on the touchscreen. This could be used, for example, to select a number from a displayed numeric keypad. The same mechanism could be used to autorepeat the selected number if the user continues to press on the displayed number.

- Flick

  A flick occurs when the user quickly touches the touchscreen, moves their finger a short distance across the surface and releases touch. It is characterized by a short touch duration. This could be used, for example, to display the next in a sequence of images.

- Drag

  A drag occurs when the user touches the touchscreen, moves their finger across the surface, and releases touch. It is characterized by a large movement across the touchscreen. Depending on the application, multiple drag events may be generated as the user moves their finger. This could be used, for example, to select a sentence in a displayed document.

The gesture events reported to the host for each touch are independent of each other which makes it easy to combine them in host software, into higher order and more complex gestures. For example, a pinch gesture is a simple combination of the drag gestures occurring on both touches with directions that are converging. Example host source code is available from Atmel on request, for these types of gestures.

The overall gesture performance from the chip relies on configuration settings that minimise the cycle time. Generally, an overall cycle of less than 20 ms yields good results.

5.8.2 Press Events

When a touchscreen is touched, the touch held, and then released, it generates a series of events. The most general case is shown in Figure 5-3.

- When a touchscreen is touched and the touch held, after a short press delay it generates a short press event.
- After a further long press delay, it generates a long press event.
- After a further repeat press delay, it generates a repeat press event.
- Finally, if the key continues to be pressed, it generates a repeat press event at regular intervals.
- When the key is released, it generates a release event.

Note: All of the delays shown in Figure 5-3 are configurable (see Section 6.2.45 on page 46).

All touches always generate release events. The generation of the short, long and repeat press events can be enabled or disabled (see Section 6.2.45 on page 46). This means that a touch could, for example, have long press events disabled. In this case pressing, holding, and releasing the touch would generate a short press event, a series of a repeat press events, and a release event, as shown in Figure 5-4 on page 19.

Figure 5-3. Sequence of Press Events

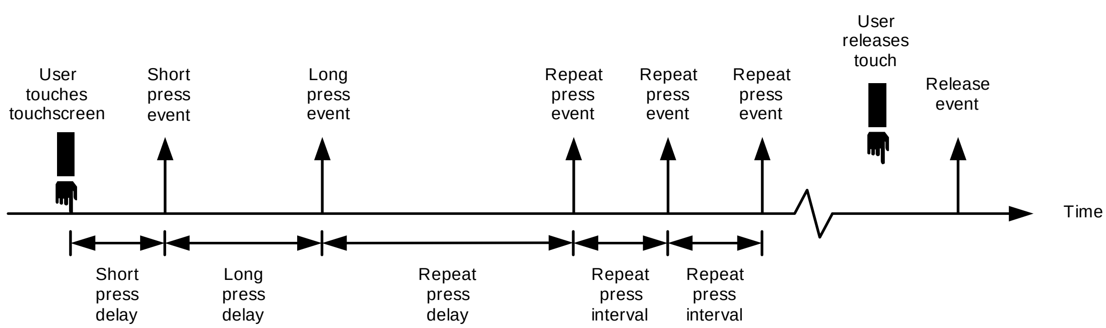
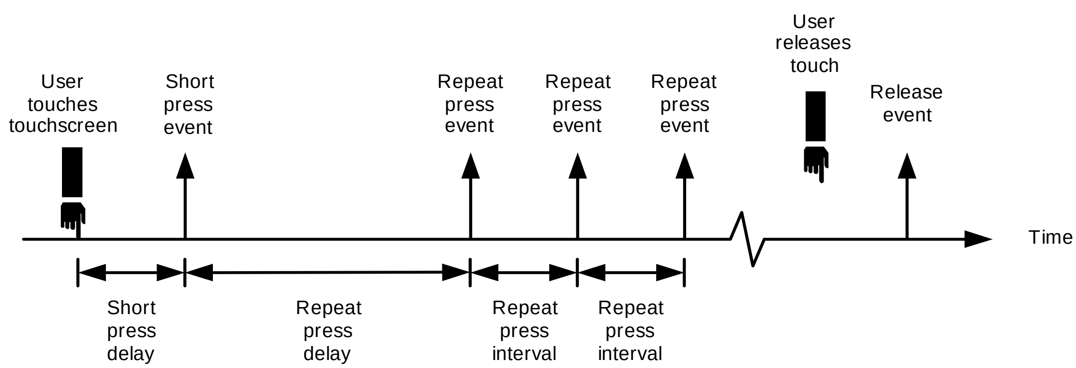


Figure 5-4. Sequence of Press Events With Long Press Disabled



5.9 Touch Size Reporting

During touch detection and touch tracking the chip computes an approximate measure of the size of the touches applied to a touchscreen (not applicable for sliders/wheels or keys).

There are two figures computed and reported:

- The touch size
- The touch area

The size measurement relates to the strength of the touch in terms of the magnitude of the touch signals and the area measurement relates to the number of nodes that are found to belong to each touch.

Changes in either of these values cause a message event to be generated unless the touch size reporting is disabled. See Section 6.2.39 on page 44 and Section 6.2.11 on page 30 for more details.

The nature of the calculations used to generate each of these measurements means that both are fairly basic approximations so expect them to fluctuate considerably with time, touch position and of course user touch pressure. The variability with finger choice and with the user's touch style also makes these measurements best suited to relative use only.

5.10 Two Touch™ Operation

5.10.1 Overview

The QT5480 can detect and track two simultaneous touches on a touchscreen (using Two Touch technology). This does not relate to simultaneous touches on two or more different objects or keys; there is no restriction on the number of such touches.

The performance of Two Touch is strongly driven by the design of the touch sensor. Higher node densities give rise to smaller touchscreens but allow closer approach of the two touches before they merge and become indistinguishable from each other.

Touchscreens using Two Touch that require the best spatial resolution in both axes should use all available channels. It is also possible to use lower Y line or X line counts in one axis and higher counts in the other axis if Two Touch is only required along that other axis.

5.10.2 Two Touch Logical Behavior

In Two Touch mode, the chip detects and tracks two touches at the same time. The first touch detected reports as touch 1 and a second touch reports as touch 2. If the first touch is then removed, touch 1 ceases reporting and the second touch continues to report as touch 2. On removal of this second (and only) touch, the next applied touch reports as touch 1 again.

5.10.3 Touch Convergence

If two touches are applied to a touchscreen and are then moved towards each other, there comes a point where the touches become indistinguishable and converge. This is because the capacitive influence of each touch extends far enough to overlap the other touch. There are a number of parameters than can be adjusted in the chip to influence the way the two touches behave as this occurs. This is to allow optimization of the behavior for different applications.

For example, a hysteresis effect can be applied so that once the touches have merged and a single touch is reported, the distance needed for separation before two touches are reported again is slightly larger than for convergence. This can help to stop oscillations between one and two touches being reported at the boundary of convergence.

5.10.4 Two Touch Accuracy

The tracking accuracy of two touches degrades as the touches get closer together. Touch separation is a function of the sensor design and node density. For best reported touch position accuracy the touches should be at least two clear nodes apart.

5.11 Touchscreen Modes

5.11.1 Y Line Wrap Mode

This feature allows a single touch touchscreen that uses the same electrode pattern as a 6 Y line design to be created, but it aliases the two outer edge Y electrodes and connects them both to Y0. A regular 6 Y line design has a Y line order Y0, Y1 ,Y2, Y3, Y4, Y5. In the wrapped configuration, the order becomes Y0, Y1, Y2, Y3, Y4, Y0, leaving Y5 spare, for use with keys or a slider (or a mix of both).

When using this mode, configure the touchscreen object width to be one less than the physical number of Y lines used in the sensor, and set the Y wrap enable bit (see Section 6.2.25 on page 36).

In this mode, the two touch enable bit must be cleared (see Section 6.2.25 on page 36).

5.11.2 Strongest Touch Mode

The logic in the chip constantly tracks two touches, even when configured for single touch operation. To allow flexibility in the behavior of the chip in one-touch mode, a configuration bit called Strongest Touch forces the touch with the biggest signal peak to report as touch 1.

With the chip configured for single touch operation:

1. The user touches the touchscreen once and keeps their finger there.
2. They then touch elsewhere on the touchscreen with their thumb and keep it there.
3. If the finger touch is removed and then reapplied the following results occur:
   a. **With Strongest Touch off**: the finger touch causes a detection status as touch 1. On applying the thumb, the finger still reports as touch 1 (internally the thumb is touch 2). On removing the finger touch, touch 1 disappears and the thumb stays as touch 2. On reapplying the finger touch it reports as touch 1 again.
   b. **With Strongest Touch on**: the finger touch causes a detection status as touch 1. On applying the thumb, the finger loses its status as touch 1 (internally the finger is touch 2). On removing the finger touch, the thumb stays as touch 1. Reapplying the finger touch will not cause a detection status at all.

5.12 Circuit Components

5.12.1 Cs Sample Capacitor Operation

Cs capacitors (Cs0 – Cs5) collect charge from the Y electrodes on the rising edge of each X pulse. A series of X pulses is known as a burst.

The Cs capacitors are charged in parallel by the X line and then measured sequentially. While waiting to be measured the capacitors are held at Vdd and measured in the order:

- Y0 (Cs0), Y1 (Cs1), Y2 (Cs2), Y3 (Cs3),Y4 (Cs4), Y5 (Cs5)

The Cs capacitors should be connected as shown in Figure 1-1 on page 5. The value of these capacitors is not critical but 4.7nF is recommended for most cases. They should be 10 percent X7R ceramics. 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 B.2 on page 64 for more information.

If a Y line is not used its corresponding Cs capacitor can be omitted and the pins left floating. Any associated channels must be disabled (see Section 6.2.19 on page 33).

5.12.2 Sample Resistors

The sample resistors (Rs0 – Rs5) 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. For most applications Rs should normally be 1MΩ in low speed mode or 500 kΩ if configured in high speed mode. Values larger than 1MΩ are not recommended. Unused Y lines do not require an Rs resistor.

5.12.3 Signal Levels

The signal values should normally be approximately 200 to 1500 counts, with properly designed electrodes and values of Rs. However, long adjacent runs of X and Y lines can also artificially boost the signal values, and induce signal saturation; avoid this. The X-to-Y coupling should come mostly from intra-electrode coupling, not from stray X-to-Y trace coupling. An error status bit is set if any channel has a signal less than 100 or greater than 4095 counts. This error bit can be useful for production testing.

Increasing the burst length (BL) parameter increases the signal strength, as does increasing the sampling resistor (Rs) values.

5.12.4 Oscillator

The QT5480 requires an external 16 MHz crystal or a 16 MHz ceramic resonator, or it can be driven by an external square wave onto the XT1 pin (in this case, a frequency of between 16 – 20 MHz, although the programmed timings will need recalculating). The resonator device only needs an initial frequency tolerance of ±1 percent or better. Resonators with integral load capacitors are suitable. In all cases the load capacitors must be approximately 12 – 22 pF.

If the chip is clocked via an external signal (F_{ak}) at greater than 16 MHz (maximum 20 MHz), then the timings such as LP mode, minimum cycle time, NRD, NDRIFT, PDRIFT and Dwell time, need scaling by the factor: 16 MHz/F_{ak}. For example, at 20 MHz; an LP mode setting of 20 ms results in a true cycle time of 16 ms (20 x 16/20).

5.12.5 Matrix Series Resistors

The X and Y matrix scan lines use series resistors (Rx0 – Rx7 and Ry0 – Ry5 respectively) for improved EMC performance (Figure 1-1 on page 5).

X drive lines require Rx in most cases to reduce edge rates and thus reduce RF emissions. Typical values range from 100Ω to 10 kΩ. For touchscreen objects 100Ω is recommended. In special cases these resistors may be omitted completely but as a general rule try to include them to reduce chances of later emissions effects.

Y lines need Ry to reduce EMC susceptibility and in some extreme cases, ESD. Typical Y values are 1kΩ – 10 kΩ. Y resistors act to reduce noise susceptibility by forming a natural low-pass filter with the Cs capacitors. For touchscreen objects 1kΩ is recommended.

It is essential that the Rx and Ry resistors and Cs capacitors are placed very close to the chip. Placing these parts more than a few millimeters away can open the circuit up to high frequency interference problems.

Dwell time is the duration in which charge coupled from X to Y is captured (Figure B-4 on page 66). Increasing Rx values causes the leading edge of the X pulses to increasingly roll off, causing the loss of captured charge (and hence loss of signal strength) from the channel.

The dwell time is programmable. If the X pulses have not settled within the chosen dwell time, channel gain is reduced; if this happens, either the dwell time should be increased or the stray capacitance on the X line(s) should be reduced. This can be done 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).

5.12.6 PCB Cleanliness

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

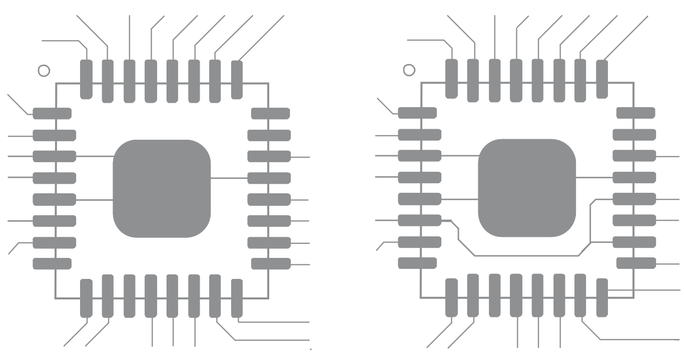
> **CAUTION:** If a PCB is reworked in any way, it is almost guaranteed that the behavior of the no-clean flux will change. This can mean that the flux changes from an inert material to one that can absorb moisture and dramatically affect capacitive measurements due to additional leakage currents. If 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.

5.12.7 MLF Package Restrictions

The central pad on the underside of the MLF chip should be connected to ground. Do not run any tracks underneath the body of the chip, only ground. Figure 5-5 shows an example of good/bad tracking.

Figure 5-5. Examples of Good and Bad Tracking



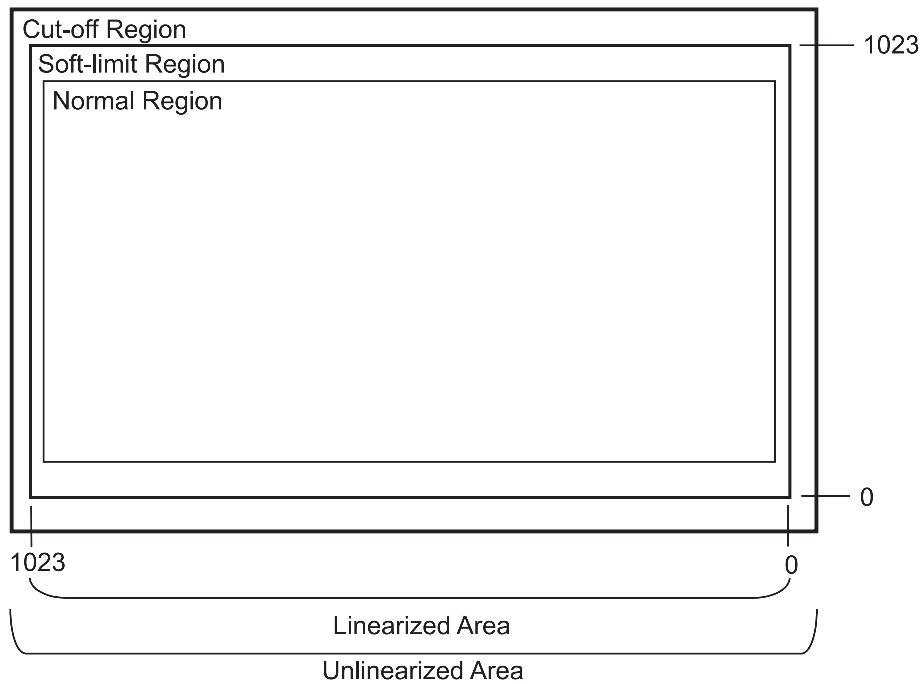
Example of good tracking   Example of bad tracking

5.13 Touchscreen Linearization

The chip contains a correction table that can be used to improve the behavior of various styles of two-dimensional sensors. See Appendix D on page 74 for more information.

The linearization controls also allow a cut-off zone to be defined around the borders of a touchscreen. The chip uses this zone to force a touch out of detect giving a nice sharp clipping effect. This means that as a touch moves over an inherently nonlinear region e.g. near to the sensor's edge wiring, then rather than report X Y coordinates that are hard for the host to interpret, the chip instead drops that touch out of detect until it moves definitively back into the non-clipped region. A hysteresis control allows this transition into and out-of the clipping region to be jitter free. See Section 6.2.27 on page 37.

Figure 5-6. Touchscreen Areas



6. Configuration Settings

6.1 Memory Map

The QT5480 processes 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 touch sensing solutions.

User-defined Setups (which are loaded into the device over the I^2C-compatible serial interfaces) are used to alter these algorithms to suit each application. The Setups are stored in an onboard EEPROM array on command from the host. The settings held in EEPROM are used to configure the chip every time it initializes after power-on or reset.

Table 6-1. Memory Map

Address	Use	Access
0	Chip ID	Read
1	Major/minor code version	Read
2	Calibrate	Read/Write
3	Reset	Read/Write
4	EEPROM backup request	Read/Write
5	Read address pointer	Read/Write
6	EEPROM checksum LSByte	Read
7	EEPROM checksum MSByte	Read
8 – 13	Key status bytes	Read
14	General status 1	Read
15	General status 2	Read
16	Touchscreen touch 0 X position MSByte	Read
17	Touchscreen touch 0 X position LSByte	Read
18	Touchscreen touch 0 Y position MSByte	Read
19	Touchscreen touch 0 Y position LSByte	Read
20	Touchscreen touch 1 X position MSByte / slider 0 position	Read
21	Touchscreen touch 1 X position LSByte / slider 1 position	Read
22	Touchscreen touch 1 Y position MSByte / slider 2 position	Read
23	Touchscreen touch 1 Y position LSByte / slider 3 position	Read
24	Slider 4 position	Read
25	Slider 5 position	Read
26	Force sensor measurement	Read
27	Channel gating input status	Read
28 – 31	Touch 0 gesture status byte 1 – 4	Read
32 – 35	Touch 1 gesture status byte 1 – 4	Read
36 – 255	Reserved	Read

Table 6-1. Memory Map (Continued)

Address	Use	Access
256	Channel 0 delta (LSByte)	Read
257	Channel 0 delta (MSByte)	Read
258 – 351	Channels 1 – 47 delta (LSByte and MSByte)	Read
352	Channel 0 reference (LSByte)	Read
353	Channel 0 reference (MSByte)	Read
354 – 447	Channels 1 – 47 reference (LSByte and MSByte)	Read
448 – 511	Reserved	Read
512 – 559	Channel control	Read/Write
560 – 607	Channel negative threshold (NTHR)	Read/Write
608 – 655	Channel burst length	Read/Write
656	LP mode	Read/Write
657	Minimum cycle time	Read/Write
658	Awake timeout	Read/Write
659	Trigger control	Read/Write
660	Guard channel enable	Read/Write
661	Touchscreen setup	Read/Write
662	Touchscreen length / slider setup 0	Read/Write
663 – 667	Slider setup 1 – 5	Read/Write
668 – 673	Touchscreen / slider 0 – 5 hysteresis	Read/Write
674	GPO control	Read/Write
675	NDRIFT setting	Read/Write
676	PDRIFT setting	Read/Write
677	Normal detect integrator limit (NDIL) setting	Read/Write
678	Short detect integrator limit (SDIL) setting	Read/Write
679	Negative recalibration delay	Read/Write
680	Drift hold time	Read/Write
681	Force sensor thresholds	Read/Write
682 – 683	Position clipping limits	Read/Write
684	Linearization table X offset LSByte	Read/Write
685	Linearization table X offset MSByte	Read/Write
686 – 701	Linearization table X segments 1 – 16	Read/Write
702	Linearization table Y offset LSByte	Read/Write
703	Linearization table Y offset MSByte	Read/Write
704 – 719	Linearization table Y segments 1 – 16	Read/Write
720	Burst control	Read/Write
721	Status mask	Read/Write

Table 6-1. Memory Map (Continued)

Address	Use	Access
722	Position filter control	Read/Write
723	Touch size resolution control	Read/Write
724	Touchscreen plateau control	Read/Write
725	Slew rate filter control	Read/Write
726	Median filter length	Read/Write
727	IIR filter control	Read/Write
728	Touchdown hysteresis	Read/Write
734 – 747	Gesture configuration registers	Read/Write

6.2 Register Descriptions

6.2.1 Address 0: Chip ID

Table 6-2. Chip ID

Address	b7	b6	b5	b4	b3	b2	b1	b0
0	CHIP ID							

The 8-bit chip ID is set at 0x40.

6.2.2 Address 1: Code Version

Table 6-3. Code Version

Address	b7	b6	b5	b4	b3	b2	b1	b0
1	MAJOR VERSION				MINOR VERSION			

This is the 8-bit major and minor version of software code revision. The top nibble of the software version register contains the major version (e.g. **5**.0) and the bottom nibble contains the minor version (e.g. 5.**0**).

6.2.3 Address 2: Calibrate

Table 6-4. Calibrate

Address	b7	b6	b5	b4	b3	b2	b1	b0
2	CALIBRATE							

Writing any nonzero value into this address triggers the QT5480 to start a calibration cycle on all enabled channels.

6.2.4 Address 3: Reset

Table 6-5. Reset

Address	b7	b6	b5	b4	b3	b2	b1	b0
3	RESET							

Any nonzero value written to this register triggers the device to perform a soft reset.

6.2.5 Address 4: EEPROM Backup Request

Table 6-6. EEPROM Backup Request

Address	b7	b6	b5	b4	b3	b2	b1	b0
4	EEPROM BACKUP REQUEST							

Write 0x55 to the backup request for the setup values to be copied to the EEPROM. This is cleared once the backup is complete. The backed-up area consists of addresses 512 - 747.

Note: Data cannot be backed up when the Low Power (LP) value is 0 because the device needs to be awake to execute the back-up command and it is never awake when LP = 0.

6.2.6 Address 5: Read Address Pointer

Table 6-7. Read Address Pointer

Address	b7	b6	b5	b4	b3	b2	b1	b0
5	READ ADDRESS POINTER							

Writing to the location of the read address pointer causes a forced-read data packet to be generated. The data location of the message is four times the address location. See the memory map (Table 6-1 on page 25) for data locations.

6.2.7 Address 6 – 7: EEPROM Checksum

Table 6-8. EEPROM Checksum

Address	b7	b6	b5	b4	b3	b2	b1	b0
6 – 7	EEPROM CHECKSUM							

This is the 16-bit checksum of the backed-up area. It is calculated at power-up and after performing an EEPROM back-up. See Appendix E on page 76 for the checksum calculation.

6.2.8 Address 8 – 13: Key Status

Table 6-9. Bits for Key Status and Numbering

Address	b7	b6	b5	b4	b3	b2	b1	b0
8	7	6	5	4	3	2	1	0
9	15	14	13	12	11	10	9	8
10	23	22	21	20	19	18	17	16
11	31	30	29	28	27	26	25	24
12	39	38	37	36	35	34	33	32
13	47	46	45	44	43	42	41	40

These addresses show the detect status for all keys. See Table 6-9 to see which keys are related to which address.

Each location indicates keys in detection, as a bitfield; touched keys report as 1, untouched or disabled keys report as 0. Channels configured as part of an object do not report their status here.

6.2.9 Address 14: General Status 1

Table 6-10. General Status 1

Address	b7	b6	b5	b4	b3	b2	b1	b0
14	RESET	CYCLE OVERRUN	SL5 DET	SL4 DET	SL3 DET	SL2 DET	SL1 DET	SL0 DET

These bits indicate the general status of the device.

RESET: this bit is set after a reset.

CYCLE OVERRUN: this bit is set if the minimum cycle time or LP mode is too short (see Section 6.2.21 on page 34 and Section 6.2.20 on page 34). If this bit is set it does not mean that any data has been lost, or measurements corrupted. It is only an indicator that the total acquisition and processing time is longer than the programmed cycle time.

SL5 – 0 DET: the appropriate bit is set if a touch is detected on sliders 0 – 5.

6.2.10 Address 15: General Status 2

Table 6-11. General Status 2

Address	b7	b6	b5	b4	b3	b2	b1	b0
15	TSCR DET	CAL	ERR	TMT			TS 1 DET	TS 0 DET

These bits indicate the general status of the device.

TSCR DET: this bit is set if any touches are detected on the touchscreen

CAL: indicates a calibration is in progress

ERR: indicates a signal error on a channel. This bit is set if a channel's signal is below 100 or above 4095, and can be used to indicate if there is a component, connection or sensor electrode problem

TMT: this bit is set if more than two touches are detected on the touchscreen. If set, this bit indicates that the two reported XY positions may not be reliable.

TS 1 DET: this bit is set if a secondary touch is detected on the touchscreen

TS 0 DET: this bit is set if a touch is detected on the touchscreen

6.2.11 Address 16 – 25: XY Positions/Touch Size

Table 6-12. XY Positions/Touch Size

<table>
<tr><th>Address</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>16</td><td colspan="8">Touchscreen touch 0, X position bits 9 – 2</td></tr>
<tr><td>17</td><td colspan="2">Touchscreen touch 0, X bits 1 – 0</td><td colspan="6">Touch 0 size</td></tr>
<tr><td>18</td><td colspan="8">Touchscreen touch 0, Y position bits 9 – 2</td></tr>
<tr><td>19</td><td colspan="2">Touchscreen touch 0, Y bits 1 – 0</td><td colspan="6">Touch 0 area</td></tr>
<tr><td rowspan="2">20*</td><td colspan="8">Touchscreen touch 1, X position bits 9 – 2</td></tr>
<tr><td colspan="8">Slider 0, X position</td></tr>
<tr><td rowspan="2">21*</td><td colspan="2">Touchscreen touch 1, X bits 1 – 0</td><td colspan="6">Touch 1 size</td></tr>
<tr><td colspan="8">Slider 1, X position</td></tr>
<tr><td rowspan="2">22*</td><td colspan="8">Touchscreen touch 1, Y position bits 9 – 2</td></tr>
<tr><td colspan="8">Slider 2, X position</td></tr>
<tr><td rowspan="2">23*</td><td colspan="2">Touchscreen touch 1, Y bits 1 – 0</td><td colspan="6">Touch 1 area</td></tr>
<tr><td colspan="8">Slider 3, X position</td></tr>
<tr><td>24</td><td colspan="8">Slider 4 X position</td></tr>
<tr><td>25</td><td colspan="8">Slider 5 X position</td></tr>
</table>

Touch size: this indicates a value that is proportional to the signal delta of the channels within a touch, and can be used, for example, to detect the size or pressure of the user's finger touch. The resolution of this data may be modified using the resolution control (Section 6.2.39 on page 44).

Touch area: this value indicates the number of channels forming a touch and can be used, for example, to reject grips or head detections.

6.2.12 Address 26: Force Sensor Measurement

Table 6-13. Force Sensor Measurement

<table>
<tr><th>Address</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>26</td><td colspan="8">FORCE SENSOR MEASUREMENT</td></tr>
</table>

This value is read from the A/D converter connected to the force sensor input. Its value is updated once per measurement cycle.

6.2.13 Address 27: Channel Gating Input Status

Table 6-14. Channel Gating Input Status

Address	b7	b6	b5	b4	b3	b2	b1	b0
27	TRIG	ADC			TRIG	ADC		

TRIG/ADC: Bits 6 and 7 show the state of the gates and read as 0 if the respective gate is disabled. Bits 2 and 3 show the true state of the inputs irrespective of whether the gate is enabled. Only the gated status is used to allow associated channels to report touch.

6.2.14 Address 28 – 31: Touch 0 Gesture Status

Table 6-15. Touch 0 Gesture Status

Address	b7	b6	b5	b4	b3	b2	b1	b0
28	Res'd	Direction			Gesture Event			
29	Distance				X position bits 3 – 0			
30	Y position bits 1 – 0		X position bits 9 – 4					
31	Y position bits 9 – 2							

The Touch 0 Gesture Status bytes report the last recognized gesture type on Touch 0.

Gesture Event: this field determines what kind of gesture is being reported, and it is decoded using Table 6-16. More information about the Gesture-processing can be found in Section 6.2.45 on page 46.

Table 6-16. Types of Gesture

Event	Description	Event	Description	Event	Description
0000	Reserved	0100	Double-tap	1000	Long Press
0001	Press	0101	Flick	1001	Repeat Press
0010	Release	0110	Drag	1010	Tap and Press
0011	Tap	0111	Short Press	1011 – 1111	Reserved

Direction: this field contains the flick direction when a flick event is generated; it is decoded using Table 6-17. If the user's finger moves in the X and Y axis then the axis of movement reported in the direction field is the axis with the greater displacement.

Table 6-17. Flick Direction

Direction	Description	Direction	Description
000	Positive Movement in X axis	100	Negative Movement in X axis
001	Reserved	101	Reserved
010	Positive Movement in Y axis	110	Negative Movement in Y axis
011	Reserved	111	Reserved

Distance: The distance field is used when a flick event is generated. It contains the distance between the start and end touch locations. The absolute distance is calculated as a 10-bit value, and bits 9 - 6 of this value are reported in the distance field. This could be used by the host as an indication of the flick speed.

X Position: this field contains the 10-bit X location at which the gesture was recognized.

Y Position: this field contains the 10-bit Y location at which the gesture was recognized.

6.2.15 Address 32 – 35: Touch 1 Gesture Status

The Touch 1 Gesture Status bytes report the last recognized gesture type on Touch 1. The description is the same as for the Touch 0 Gesture status except that the status bytes are located at addresses 32 - 35.

6.2.16 Address 256 – 447: Debug Data

Addresses 256 – 351 allow delta data to be read for each channel. There are two bytes of data for each channel. These are the channel's 16-bit signed delta value which is accessed as two 8-bit bytes, stored LSByte first. For ease of display, deltas are shown with a touch showing a positive delta. In reality, the sign of the delta is reversed; hence the normal detection threshold is referred to as the negative detection threshold.

Addresses 352 – 447 allow reference data to be read for each channel. There are two bytes of data for each channel. These are the channel's 16-bit unsigned reference value which is accessed as two 8-bit bytes, stored LSByte first.

There are a total of 48 channels and 4 bytes of data per channel, yielding a total of 192 addresses. These addresses are read-only.

Table 6-18. Debug Data

Address	Key #	Use	Address	Key #	Use
256	0	Delta* LSByte	352	0	Reference LSByte
257	0	Delta* MSByte	353	0	Reference MSByte
258	1	Delta* LSByte	354	1	Reference LSByte
259	1	Delta* MSByte	355	1	Reference MSByte
260 – 351	2 – 47		356 – 447	2 – 47	

Note: * Delta = Reference - Signal

6.2.17 Address 512 – 559: Channel Control

Table 6-19. Channel Control

Address	b7	b6	b5	b4	b3	b2	b1	b0
512 – 559	TRIG	FORCE					AKS GROUP	

TRIG/FORCE: These bits are for channel 0 – 47 and are associated with the gate control. Setting the trigger and/or ADC bit means these have to be asserted in order for the key to detect.

AKS GROUP: These bits configure which AKS group a channel is within (0 (off), 1, 2 or 3). If the AKS feature is not required on a channel it should be turned off for that channel as this allows faster cycle times and consequently a quicker response time.

Default: 0x0

6.2.18 Address 560 – 607: Channel Negative Threshold (NTHR)

Table 6-20. Channel Negative Threshold (NTHR)

Address	b7	b6	b5	b4	b3	b2	b1	b0
560 – 607	CHANNEL NEGATIVE THRESHOLD (NTHR)							

The channel negative threshold (NTHR) defines how much a channel's signal must drop below its reference level (determined during calibration and adjusted using drift compensation) to qualify as a potential touch detect. The final detection confirmation uses the Detection Integration as described in Section 6.2.30 on page 40. Larger values of threshold desensitize channels since the signal must change more in order to exceed the threshold level. Conversely, lower thresholds make channels more sensitive.

As Cx and Cs drift, the reference is automatically compensated with these changes at a user-configurable rate.

The setting for NTHR for each channel depends on the amount of signal swing that occurs when a channel is touched. Thicker panels or smaller electrode geometries reduce channel gain, i.e. signal swing from touch, thus requiring smaller NTHR values to detect touch.

Once a channel has registered as being touched, its internal detection threshold is reduced a small amount to effect hysteresis so as to stop channels dithering in and out of detect. The out-of-detect threshold is 25 percent below the detection threshold.

NTHR Typical values: Keys and Slider Channels: 10 – 30, Touchscreen Channels: 15 – 20

NTHR Default value: 0

6.2.19 Address 608 – 655: Channel Burst Length

Table 6-21. Channel Burst Length

Address	b7	b6	b5	b4	b3	b2	b1	b0
608 – 655	CHANNEL BURST LENGTH							

The burst-length setting controls the number of charge-transfer cycles (pulses of the corresponding X line) that are executed in order to measure the capacitance of a channel. This number of pulses generated is the value written to this register rounded down to the nearest multiple of 4; that is, if 9 is entered it is rounded down to 8. A value of zero disables the channel and it will not be measured.

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

Apparent touch sensitivity is also controlled by the Negative Threshold level (NTHR). Burst length and NTHR interact; normally burst lengths should be kept as short as possible to reduce scan time, 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.

Default: 0 (channel off)

6.2.20 Address 656: LP Mode

Table 6-22. LP Mode

Address	b7	b6	b5	b4	b3	b2	b1	b0
656	LP MODE							

LP mode sets the overall cycle time (burst time + sleep time between bursts) when the awake-timeout is not active. A higher value causes more sleep time between acquisitions resulting in lower power consumption, but slower response time. The permitted values are 0 – 255 ms.

A value of zero causes the device to enter an ultra-low power mode (SLEEP), where no measurements are carried out. The chip enters SLEEP mode after the expiry of the awake-timeout, if it has recently been touched. The QT5480 is designed to sleep as much as possible to conserve power. Do not set the LP mode or minimum cycle time to be less than the actual burst time.

Note: Settings cannot be backed up to EEPROM when LP = 0 because the device needs to be awake to execute the back-up command and it is never awake when LP = 0.

Default value: 0

6.2.21 Address 657: Minimum Cycle Time

Table 6-23. Minimum Cycle Time

Address	b7	b6	b5	b4	b3	b2	b1	b0
657	MINIMUM CYCLE TIME							

This sets the minimum time for the completion of a burst cycle while the awake-timeout is active. The permitted values are 0 – 254 ms.

A setting of zero forces the chip to go to SLEEP mode immediately, irrespective of any active touches or the awake-timeout.

A setting of 255 makes the chip run in free-run mode during the awake timeout (where it does not sleep, giving the fastest response time).

Note: Settings cannot be backed up to EEPROM when minimum cycle time = 0 because the device needs to be awake to execute the back-up command and it is not awake when minimum cycle time = 0.

Default value: 16 ms

6.2.22 Address 658: Awake Timeout

Table 6-24. AWAKE Timeout

Address	b7	b6	b5	b4	b3	b2	b1	b0
658	AWAKE TIMEOUT							

After each matrix scan, the device automatically goes to sleep whenever possible to conserve power, unless there has been a channel state change or an object is being touched. The AWAKE timeout feature determines how long the device remains in this mode from the last state change.

Subsequent channel state changes, touched objects or trigger activations reinitialize the AWAKE timeout. Once the device has been awakened by a change, the touch response time is fast for as long as the sensor remains in use. Once channel activity lapses for a period longer than the AWAKE timeout, the device returns to LP mode.

The AWAKE period can be configured to a value between 0 and 51s, in increments of 0.2s.

Default value: 50 (10s)

6.2.23 Address 659: Trigger Control

Table 6-25. Trigger Control

Address	b7	b6	b5	b4	b3	b2	b1	b0
659	TRIG	ADC			SYNC_EN	SYNC _MODE	SYNC_EDGE	

TRIG/ADC: These bits determine the global enables on the trigger condition to allow them to be switched on or off. If a trigger is disabled then channels that are configured as gated by that trigger never register a detection.

SYNC_EN: This bit enables the chip's synchronization feature, useful for synchronizing with external noise sources (for example, LCDs). Setting this bit forces the chip to wait for an edge on the TRIGGER pin before performing a measurement. Clearing this bit disables this feature and the chip will run as configured by the LP and minimum cycle time settings.

Note: If this feature is enabled the TRIGGER gating feature (see Section 6.2.17 on page 32) should not be used.

SYNC_MODE: This bit selects how the chip behaves when a synchronization edge is detected. Clearing this bit forces the chip to perform the measurement on a single X line after the edge and wait until the next edge before measuring the next X line. Setting this bit forces the chip to perform a measurement on all channels after the edge is detected and wait until the next edge before measuring all of the channels again. When this bit is set the chip ignores its LP and minimum cycle time settings and is timed entirely by the SYNC input. When clear, the chip runs at its programmed cycle times but will attempt to synchronize the X line measurements to the SYNC input.

SYNC_EDGE: These bits select whether a rising and/or falling edge is used for the synchronization feature.

SYNC_EDGE	Synchronization
00	Rising edge
01	Falling edge
10	Either edge
11	Reserved

Note: If synchronization edges occur more frequently than the chip is able to synchronize to, then the IC will wait for the next occurring edge before continuing.

Default: 0 (all triggers off and synchronization off)

6.2.24 Address 660: Guard Channel Enable

Table 6-26. Guard Channel Enable

Address	b7	b6	b5	b4	b3	b2	b1	b0
660	GUARD CHANNEL NUMBER							

This option allows any channel to be configured as a guard channel to prevent false detection (see Section 5.6.2 on page 15). Write the desired guard channel number into the register. An invalid channel number disables the guard channel completely. With the guard channel not enabled, all the channels work normally.

When the guard channel is enabled and active, no channel with the AKS option enabled can be activated. It forces any AKS-enabled channels that are in detect, out of detect. The guard channel itself will not report its detection status.

Default: 48 (disabled)

6.2.25 Address 661: Touchscreen Setup

Table 6-27. Touchscreen Setup

Address	b7	b6	b5	b4	b3	b2	b1	b0
661	Y WRAP	GESTURE ENABLE	STRONGEST TOUCH	TWO TOUCH	TOUCHSCREEN HEIGHT			

This enables touchscreen processing, and sets the touchscreen width and operating mode.

Bit 7: Y wrap enable. See Section 5.11.1 on page 21.

Bit 6: Gesture Enable. This bit enables the on-chip gesture processing module. When set, any detected gestures are indicated in the gesture status bytes (see Section 6.2.45 on page 46). When gesture processing is enabled the host may wish to disable normal touchscreen messages using the mask control (see Section 6.2.37 on page 43).

Bit 5: Strongest touch. This bit controls how the QT5480 tracks touches in single touch mode. This bit must be cleared when two touch is set.

Bit 4: Two touch enable. If set, the position and detection status of secondary touches is reported.

Bits 3 – 0: Touchscreen height. This sets the number of Y lines used in the touchscreen. The touchscreen starts at Y0 and uses the number of Y lines given by this setting. A value of 0 disables the touchscreen. A minimum of 3 Y lines (4 Y lines for Two Touch) are required for a touchscreen. See Section 5.11.1 on page 21.

Default: 0

6.2.26 Address 662 – 667: Touchscreen Length / Slider (0 – 5) Setup

Table 6-28. Touchscreen Length / Slider (0 – 5) Setup

Address	b7	b6	b5	b4	b3	b2	b1	b0
662 – 667	WHEEL				TOUCHSCREEN/SLIDER LENGTH			

Wheel: if this bit is set the slider is processed as if it were a wheel, with the first and last channels in the slider located next to each other (not valid for touchscreen use).

Touchscreen/Slider Length: this sets the number of X lines used in each slider. All sliders start at X0. If a touchscreen is enabled, its length is set using the first slider length register only. A minimum of 3 X lines are required for a touchscreen or a wheel, and two for a slider.

Default: 0

6.2.27 Address 668 – 673: Touchscreen/Slider (0 – 5) Hysteresis

Table 6-29. Touchscreen/Slider (0 – 5) Hysteresis

Address	b7	b6	b5	b4	b3	b2	b1	b0
668	TOUCHSCREEN: MOVEMENT HYSTERESIS SLIDER 0: MOVEMENT HYSTERESIS							
669	TOUCHSCREEN: CLIPPING HYSTERESIS SLIDER 1: MOVEMENT HYSTERESIS							
670	TOUCHSCREEN: TWO TOUCH HYSTERESIS SLIDER 2: MOVEMENT HYSTERESIS							
671 – 673	SLIDER 3 – 5: MOVEMENT HYSTERESIS							

These registers configure the hysteresis applied to the coordinates reported by the chip. The use of these registers is slightly different depending on whether they are controlling a slider's behavior or a touchscreen's behavior.

Slider Use

- **Slider 0 – 5 Movement Hysteresis:** When the user first touches the slider this value sets the amount that the user’s touch must move by before a position update is signaled. Once the user’s touch has started moving, this value then controls the amount that the user’s touch must move in the opposite direction before a direction change is indicated in the reported position. This control allows a simple jitter filter to be implemented. Higher hysteresis settings result in a stronger filter.

Touchscreen Use

- **Movement Hysteresis:** When the user first touches the touchscreen, the value configured by the Touchdown Hysteresis (see Section 6.2.44 on page 45) is applied to any movement before a position update is signaled. Once this limit has been crossed in any direction then the Movement Hysteresis is applied to the calculated position before a direction change in an axis is reported to the host. This control allows a simple jitter filter to be implemented. Higher hysteresis settings result in a stronger filter.
- **Clipping hysteresis:** This control is to ensure a smooth cutoff when the touch position reaches the clipping region as configured using the linearization controls (see Section 5.13 on page 23). This setting specifies the distance that the calculated position must move into the clipping region before the touch is no longer reported as present, and then the distance back into the active region that the touch must then move before being reported as present. A setting of zero applies no hysteresis.
- **Two Touch hysteresis:** This control ensures smooth touchscreen behavior when two touches are close to the boundary where the QT5480 can no longer track two touches independently because they are too close together. This setting is applied on top of the Touchscreen Plateau Control (see Section 6.2.40 on page 44) level when the QT5480 decides how many touches are present on the screen. A setting of zero applies no hysteresis; higher values apply more hysteresis so that the required touch separation to move from 1 to 2 touches becomes progressively larger than the separation before moving from 2 to 1 touches.

Default: 0

6.2.28 Address 674: GPO Control

Table 6-30. GPO Control

Address	b7	b6	b5	b4	b3	b2	b1	b0
674					GPO3	GPO2	GPO1	GPO0

This register allows the user to control the state of the general purpose outputs (GPO's). They are updated when this register is written to, with an output state that reflects the corresponding bit's state. The outputs float for approximately 10 ms during the chip initialization and then initialize to zero after power-on. This register cannot be stored to EEPROM.

Default: 0x00 (all outputs are 0)

6.2.29 Address 675 – 676: Negative/Positive Drift Compensation

Table 6-31. Negative/Positive Drift Compensation

Address	b7	b6	b5	b4	b3	b2	b1	b0
675	0	NDRIFT						
676	0	PDRIFT						

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

Drift compensation (Figure 6-1 on page 39) 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 range of settings is 1 – 127. The parameters can be configured in increments of 0.2s.

The device drift compensates using a slew-rate limited change to the reference level; the threshold and hysteresis values are slaved to this reference.

When a finger is sensed, the signal falls since the human body acts to absorb charge from the cross-coupling between X and Y lines. An isolated, untouched foreign object (a coin, or a water film) 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 asymmetric; 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 Setup parameters. This is a global configuration.

Specifically, drift compensation should be set to compensate faster for increasing signals than for decreasing signals. Decreasing signals should not be compensated for quickly, since an approaching finger could be compensated for partially or entirely before even touching the channel (NDRIFT).

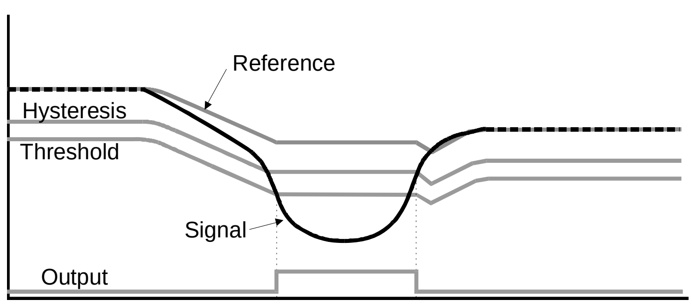
However, an object over the channel which does not cause a detection, and for which the sensor has already made full allowance (over some period of time), could suddenly be removed leaving the sensor with an artificially suppressed reference level and thus become insensitive to touch. In the latter case,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

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.

Default NDRIFT: 20 (4s per LSB)

Default PDRIFT: 5 (1s per LSB)

Figure 6-1. Thresholds and Drift Compensation



6.2.30 Address 677: Normal Detect Integrator Limit (NDIL)

Table 6-32. Normal Detect Integrator

Address	b7	b6	b5	b4	b3	b2	b1	b0
677	NDIL							

The Normal Detect Integrator Limit (NDIL) is used to provide signal filtering.

To suppress false detections caused by spurious events like electrical noise, the device incorporates a detection integrator (DI) counter mechanism. A per-key counter is incremented each time the channel has exceeded its threshold and stayed there for a number of acquisitions in succession, without going below the threshold level. When this counter reaches a preset limit the channel is finally declared to be touched. If on any acquisition the delta is not seen to exceed the threshold level, the counter is cleared and the process has to start from the beginning.

NDIL Default value: 4

6.2.31 Address 678: Short Detect Integrator Limit (SDIL)

Table 6-33. Short Detect Integrator

Address	b7	b6	b5	b4	b3	b2	b1	b0
678	SDIL							

The Short Detect Integrator Limit (SDIL) is used in combination with NDIL to provide optimized signal filtering on sliders and touchscreen. It has two purposes:

- The internal slider digital filters start to operate after SDIL acquisitions, even though a touch is not reported until NDIL acquisitions; this ensures a better filtered position is reported when a touch is first reported.
- On a touchscreen or a slider, once the object has reported as touched then subsequent channels only require SDIL acquisitions with sufficient touch delta to qualify as in-detect. This ensures a fast response during rapid touch movements.

SDIL Default value: 2

6.2.32 Address 679: Negative Recalibration Delay

Table 6-34. Negative Recalibration Delay

Address	b7	b6	b5	b4	b3	b2	b1	b0
679	NRD							

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

The Negative Recalibration Delay (NRD) timer monitors such detections; if a detection event exceeds the timer's setting, the channel is automatically recalibrated. After a recalibration has taken place, the affected channel once again functions normally even if it is still being contacted by the foreign object. This feature is set globally. The NRD does not apply to guard channels.

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 can range in value from 0 – 255, with 0 = infinite. NRD above 0 is expressed in 0.2s increments.

Default: 150 (30s)

6.2.33 Address 680: Drift Hold Time

Table 6-35. Drift Hold Time

Address	b7	b6	b5	b4	b3	b2	b1	b0
680	DHT							

Drift Hold Time (DHT) is used to restrict drift on all channels while one or more channel are activated. It defines the length of time the drift is halted after a key detection.

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 any touch detection.

DHT can be configured to a value of between 2 – 255 (0.2s and 51s), in increments of 0.2s. Values of 0 and 1 are invalid and should not be used.

Default: 20 (4s)

6.2.34 Address 681: Force Sensor Thresholds

Table 6-36. Force Sensor Thresholds

Address	b7	b6	b5	b4	b3	b2	b1	b0
681	ON_THR				OFF_THR			

This applies to the force sensor trigger and specifies the on and off thresholds (see Section 5.6.1 on page 15). Values are from 2 percent to 25 percent of Vdd. The OFF_THR is added to the ON_THR.

0 = 2%	4 = 8%	8 = 14%	12 = 20%
1 = 3%	5 = 9%	9 = 16%	13 = 22%
2 = 5%	6 = 11%	10 = 17%	14 = 23%
3 = 6%	7 = 12%	11 = 19%	15 = 25%

Default: 0xFF (on threshold = 25 percent, off threshold = 50 percent Vdd)

6.2.35 Address 682 – 719: Linearization Controls

Table 6-37. Linearization Controls

Address	b7	b6	b5	b4	b3	b2	b1	b0
682	POSITION CLIPPING LIMIT X							
683	POSITION CLIPPING LIMIT Y							
684	OFFSET_X (LSB)							
685	OFFSET_X (MSB)							
686 – 701	SEGMENT_X							
702	OFFSET_Y (LSB)							
703	OFFSET_Y (MSB)							
704 – 719	SEGMENT_Y							

If the touchscreen produces nonlinear outputs, linear interpolation can be carried out on the outputs to improve linearity. Figure 5-6 on page 24 shows the linearization areas. There is no provision to use this for sliders or wheels.

If linearization is not required, setting the POSITION CLIPPING LIMITs to 0, OFFSETs to 0, and SEGMENTS to 64 disables linearization and allows the output to be passed through unmodified.

Refer to Section D on page 74 for the process of calculating the linearization coefficients.

POSITION CLIPPING LIMITS: These limits define a soft boundary for the touchscreen. The reported coordinate in each axis will never go below (0 + LIMIT) or above (1023 - LIMIT). Touching outside the bounded region, but over the active electrode area, still results in a detect status and a coordinate update but with one or both coordinates set to the boundary limit.

Default: 0

OFFSET_X / OFFSET_Y: The unlinearized coordinates that after linearization cause X and Y to report 0.

Default: 0

SEGMENT_X / SEGMENT_Y: Linearization coefficients for the X and Y axes respectively. Up to 16 coefficients can be used for each axis.

Note: Unused SEGMENT X/Y must be set to 64 and not 0.

Default: 64

6.2.36 Address 720: Burst Control

Table 6-38. Burst Control

Address	b7	b6	b5	b4	b3	b2	b1	b0
720						DWELL		

This setting controls the charge transfer dwell time. It should normally be set to 1 µs but can be reduced if the sensor allows (see Section 5.12.5 on page 22), or increased if required (at the expense of a slower scan time).

Table 6-39. Burst Control Timing

Burst Control Setting	Dwell Time	
	High Speed Mode I^2C-compatible address 0x30 or 0x40	Low Speed Mode I^2C-compatible address 0x11 or 0x20
0	1 µs	1 µs
1	0.5 µs	1 µs
2	1 µs	2 µs
3	1.5 µs	3 µs
4	2.0 µs	4 µs
5	2.5 µs	5 µs

Default: 0x00 (1 µs)

6.2.37 Address 721: Status Mask

Table 6-40. Status Mask

Address	b7	b6	b5	b4	b3	b2	b1	b0
721	–	32 – 35	31 – 28	27 – 24	23 – 20	19 – 16	15 – 12	11 – 8

This mask allows the host to disable the generation of individual event messages by the QT5480. Each bit controls the automatic generation of status event messages at certain addresses. If a message is masked by clearing the corresponding bit then the QT5480 will not signal an interrupt via the $\overline{\text{CHANGE}}$ line when the data at these addresses change value.

When an event message is masked off, the host can still read the values in the masked status registers using the Read Address Pointer (See Section 7.4 on page 49).

This control might be used when touchscreen-gesture processing is enabled so that the host can disable the reporting of normal XY coordinates to reduce communications traffic.

Default: 0xFF (all status packets enabled)

6.2.38 Address 722: Position Filter Control

Table 6-41. Position Filter Control

Address	b7	b6	b5	b4	b3	b2	b1	b0
722	–				POSITION FILTER CONTROL			

This setting allows the host to change the level of filtering applied to the calculated touchscreen or slider position. A setting of zero disables the filter, a setting of 15 applies the filter at its strongest level. This value can be tuned, based on the device's programmed cycle time and the level of system noise, but under normal circumstances the default setting is recommended.

Default: 8

6.2.39 Address 723: Touch Size Resolution Control

Table 6-42. Touch Size Resolution Control

Address	b7	b6	b5	b4	b3	b2	b1	b0
723	–					TOUCH SIZE RESOLUTION		

This setting allows the host to modify the resolution/dynamic range of the reported touch size (see Section 6.2.11 on page 30). A setting of 7 disables reporting of the touch size and touch area. A setting of 0 makes the QT5480 calculate the touch size at the maximum possible resolution but gives the smallest dynamic range (i.e. the maximum size that can be reported). Subsequent settings halve the resolution but double the dynamic range; for example, 1 = ½ resolution, 2 = ¼ resolution.

Note that natural fluctuations in the calculated touch size and area cause messages to be generated to the host. This can happen even in cases where the resolution has been set to very coarse values. If such messages are not desirable then using a setting of 7 can be useful to suppress all the size and area values.

Default: 3

6.2.40 Address 724: Touchscreen Plateau Control

Table 6-43. Touchscreen Plateau Control

Address	b7	b6	b5	b4	b3	b2	b1	b0
724	TOUCHSCREEN PLATEAU CONTROL							

This setting allows the host to tune the amount of unevenness, in a measured area, that the device still considers as one individual touch, beyond which it considers there to be multiple touches present on the screen.

This value may, in certain circumstances, be used to allow the grouping of a large and uneven touch, such as a user's palm, into a single tracked touch. Larger values allow more uneven touches to be tracked as a single touch, but reduce the minimum separation of two true touches.

Contact Atmel's Touch Technology Division for further information.

Default: 0

6.2.41 Address 725: Slew Rate Filter Control

Table 6-44. Slew Rate Filter Control

Address	b7	b6	b5	b4	b3	b2	b1	b0
725	SLEW RATE FILTER CONTROL							

When nonzero this value is the amount that the signal level on a channel may change per cycle.

Default: 0 (off)

6.2.42 Address 726: Median Filter Length

Table 6-45. Median Filter Length

Address	b7	b6	b5	b4	b3	b2	b1	b0
726	MEDIAN FILTER LENGTH							

This setting controls the median filter that is applied to the channel signals; a setting of 0 bypasses this filter. A nonzero value sets the length of the median filter that is applied to the measured signals, before they are fed into the detection and position calculation processing.

The maximum median length configurable is 6. Odd settings generate a true median filter, while even settings result in the middle two values being averaged to generate the filter output. See Section 5.7.4 on page 17

Default: 0 (off)

6.2.43 Address 727: IIR Filter Control

Table 6-46. IIR Filter Control

Address	b7	b6	b5	b4	b3	b2	b1	b0
727	IIR FILTER CONTROL							

This setting controls the final digital filter that is applied to the channel signals; a setting of 0 disables this filter. A nonzero value (maximum of 5) sets the filter coefficient that is applied to the measured signals before they are fed into the detection and position calculation processing. Higher coefficient settings result in a stronger filtering effect at the expense of a slower response time and greater lag in the reported touch position

Default: 0 (off)

6.2.44 Address 728: Touchdown Hysteresis

Table 6-47. Touchdown Hysteresis

Address	b7	b6	b5	b4	b3	b2	b1	b0
728	TOUCHDOWN HYSTERESIS							

This hysteresis setting is applied to the reported position when a user has just touched down onto the touchscreen. Once the position of the user's finger moves from its initial position beyond this setting in any direction, then position updates are signaled and the normal touchscreen positional hysteresis setting is used to filter changes in direction in either axis.

This setting allows the host to differentiate between an intended drag and a simple press. If this control is not required, set it to the same value as the normal touchscreen position hysteresis setting.

Default: 0

6.2.45 Address 734 – 747: Gesture Configuration Registers

Table 6-48. Gesture Configuration Registers

Address	b7	b6	b5	b4	b3	b2	b1	b0
734	Reserved					Press Processing Controls		
						ENABLE REPEAT PRESS	ENABLE LONG PRESS	ENABLE SHORT PRESS
735	TAP TIMEOUT							
736	DRAG TIMEOUT							
737	FLICK TIMEOUT							
738	SHORT TIMEOUT							
739	LONG TIMEOUT							
740	REPEAT TIMEOUT							
741	RESERVED							
742	MINIMUM FLICK MOVEMENT LSByte							
743	MINIMUM FLICK MOVEMENT MSByte							
744	DRAG THRESHOLD LSByte							
745	DRAG THRESHOLD MSByte							
746 – 747	RESERVED							

These registers configure the on-chip touchscreen gesture processing, which is enabled by setting the Gesture Enable bit in the touchscreen setup register (see Section 6.2.25 on page 36).

Press Processing Controls: These bits (b0 – b2) control the type of press events that can be signaled by the chip by enabling the various press gesture messages. Setting the relevant control bit enables the reporting of the event type.

Default: 0

Tap Timeout

This value specifies the maximum duration of a tap. If the user presses the touchscreen, does not move their finger, and then releases the touchscreen, and this sequence takes less time than the tap timeout, then a tap event is generated. The tap timeout is specified in units of 32 ms.

This value also controls the generation of double tap events. A double tap event consists of a press, release, press, and final release. If the time between each of these actions is less than the tap timeout, a double tap event is generated.

The user must move their finger less than the Drag Threshold (see "Drag Threshold" on page 48) during a tap to generate a tap or double tap gesture. Also, the distance between consecutive taps must be less than the Drag Threshold to generate a double tap gesture.

Default: 6 (192 ms)

Drag Timeout

This value specifies the maximum time between two drag events. If the user presses the touchscreen and moves their finger, a drag event is generated. If the user then moves their finger again within the drag timeout period, another drag event is generated. If the user's finger remains stationary for longer than the drag timeout period, drag processing stops, and any enabled press events start to be generated. See also "Drag Threshold" on page 48. The drag timeout is specified in units of 32 ms.

Default: 16 (512 ms)

Flick Timeout

This value specifies the maximum duration of a flick gesture. If the user presses the touchscreen, moves their finger, and then releases the touchscreen, and this sequence takes less time than the flick timeout, then a flick event is generated. See also "Minimum Flick Movement" on page 47. The flick timeout is specified in units of 32 ms.

Default: 6 (192 ms)

Short Timeout

This value specifies the time between starting press processing, and the generation of a short press event. It is only used if short press processing is enabled using the Press Processing Controls parameter on page 46.The short timeout is specified in units of 32 ms.

Default: 16 (512 ms)

Long Timeout

This value specifies the time between a short press event, and the generation of a long press event. It is only used if long press processing is enabled using the Press Processing Controls parameter on page 46. The long timeout is specified in units of 32 ms.

Default: 16 (512 ms)

Repeat Timeout

This value specifies the time between a long press event, and the generation of a repeat press event. It also specifies the time between the generation of consecutive repeat press events. It is only used if repeat press processing is enabled using the Press Processing Controls parameter on page 46. The repeat timeout is specified in units of 32 ms.

Default: 16 (512 ms)

Minimum Flick Movement

This value specifies how far the user must move their finger across the touchscreen to generate a flick gesture. If they do not move their finger by at least this distance, a flick is not generated. See also "Flick Timeout" on page 47.

The minimum flick movement is specified in units of touchscreen distance. The touchscreen distance between two points is the sum of the absolute difference between the X and Y coordinates of the points.

Default: 75

Drag Threshold

This value specifies how far the user must move their finger across the touchscreen to generate a drag event. See also "Drag Timeout" on page 47. The drag threshold is specified in units of touchscreen distance (see "Minimum Flick Movement" on page 47).

Default: 50

7. Getting Started

7.1 Using the I²C-compatible Bus

The QT5480 is based around an event driven data packet system (see Section 5.3.1 on page 12).

7.2 Establishing Contact

To establish that the device is present and running, reset the device and read the event based data packet it generates on coming out of reset. This should contain two zero bytes followed by the two general status bytes. The first of these should have its reset bit set. If this is the case the device is present and running.

7.3 Writing to the Device

A WRITE cycle to the device consists of a START condition followed by the I²C-compatible address of the device. The next two bytes are the address of the location (LSByte first) into which the writing starts. This address is then stored as the address pointer.

Subsequent bytes in a multibyte transfer are written to the location of the address pointer, location of the address pointer +1, location of the address pointer +2 etc. This ends with the STOP condition on the I²C-compatible bus. A new write cycle involves sending another address pointer.

7.4 Reading From the Device

Reads are done by writing the desired address/4 to the READ address pointer. This generates a read data packet (see Section 5.3.3 on page 12).

7.5 Backing Up User Settings to EEPROM

The configurations stored in address locations 512 onwards are read from an EEPROM store on power-up or after a reset. These settings are nonvolatile and are not lost on power-down.

7.6 Channels

The default setting of the QT5480 is for all 48 channels to be disabled and in AKS group 0. This is the default setting when the device first powers up. A coin placed over any channel does not pick up the burst signal until they are enabled. See the activity on the channels as explained in Appendix B on page 64.

The $\overline{\text{CHANGE}}$ line goes low (default setting for this line is active low) indicating there is new data to be read at power-up. In this case it is the reset bit in the general status register.

If a key is enabled and touched, the $\overline{\text{CHANGE}}$ line becomes active again, indicating that there is new data again. The $\overline{\text{CHANGE}}$ line remains active until the status location containing the status for that key is read. If the $\overline{\text{CHANGE}}$ line does not go low then it is likely the sensitivity of the key is not high enough. The burst length should be increased to increase the sensitivity.

A change in burst length should be followed by a calibration command (set the calibration byte to a nonzero value) to ensure reliable operation. It is also possible to adjust the sensitivity using the negative threshold for that key. Note that thresholds below 6 counts may cause sensitivity to noise and thresholds above 12 counts require longer burst lengths than strictly necessary.

All unused channels should be switched off by setting their burst lengths to zero. This reduces the power requirements of the device.

7.7 Sliders

Groups of channels on the same Y line can be configured as a slider. These have to be placed in numerical order starting with X0 and with no missing channels in the sequence. The channels should be 5 – 7 mm wide along the length of the slider for good linearity. The number of channels needed in a slider are simply the number required to form the desired slider length.

The slider can now be enabled by setting the slider setup configuration to the number of channels which are used in the slider. This can be from 2 – 8 channels. For example, to enable a slider of 5 keys on Y line 1, set slider setup 1 to 5.

Note: Sliders must start with X line X0.

Now the slider is enabled, touching it results in a slider position being reported in the X status byte for the Y line being used. In the previous example this is slider X1.

Note: The channels forming the slider will not report their status in the key status registers.

If the slider position is noisy, the slider hysteresis setting can be increased to provide some positional hysteresis. This updates the slider value as long as the slide continues in the same direction. If there is a change in direction, the slide must move by more than the positional hysteresis before a change is registered. This is a simple noise filter.

Keys within the same slider are normally in the same AKS group and have the same burst length and threshold.

7.8 Touchscreens

A group of channels can be configured as a touchscreen, reporting the position of touch in 10-bit resolution. The touchscreen is made up of a programmable number of channels; spanning across 3 – 6 Y lines (for best linearity use as many Y lines as possible). As with other objects the keys used must start from X0.

The touchscreen is configured by setting *slider setup 0* to the desired X length of the touchscreen (i.e. the number of X lines to use) and *touchscreen enable* to the desired Y width of the touchscreen (i.e. the number of Y lines to use).

For example, if slider setup 0 is set to 4 and touchscreen enable set to 3, then a 4 x 3 touchscreen is enabled using channels 0, 1, 2, 3, 8, 9, 10, 11, 16, 17, 18 and 19, with the remaining channels functioning as normal keys.

If the touchscreen position is noisy, the touchscreen hysteresis (configured by slider 0 hysteresis) can be increased to provide some positional hysteresis. This updates the touchscreen value as long as the touch continues in the same direction. If there is a change in direction, the touch must move by more than the positional hysteresis before a change is registered. This is a simple noise filter.

Channels within a touchscreen are normally in the same AKS group and have the same burst length and threshold.

8. Specifications

8.1 Absolute Maximum Specifications

Vdd	-0.5 to +6V
Max continuous pin current, any control or drive pin	±10 mA
Voltage forced onto any pin	-0.6V to (Vdd + 0.6) Volts
EEPROM setups maximum writes	100,000 write cycles

CAUTION: Stresses beyond those listed under *Absolute Maximum Specifications* 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.

8.2 Recommended Operating Conditions

Operating temp	-40°C to +85°C
Storage temp	-55°C to +125°C
Vdd	1.8V to 5.5V in low speed mode (2.7V to 5.5V in high speed mode)
Supply ripple + noise	±5 mV
Cx transverse load capacitance per key	2 to 20 pF
GPO current	500 µA

8.3 DC Specifications

Vdd = 5.0V, Cs = 4.7 nF, Rs = 1 MΩ, Ta = recommended range, unless otherwise noted

Parameter	Description	Min	Typ	Max	Units	Notes
Iddr	Average supply current, running	–	–	–	–	See Table 8-1 on page 52
Vil	Low input logic level	–	–	0.2 Vdd	V	1.8V <Vdd <5V
Vhl	High input logic level	0.6Vdd	–	–	V	1.8V <Vdd <5V
Vol	Low output voltage	–	–	0.2	V	
Voh	High output voltage	4.2	–	–	V	
Iil	Input leakage current	–	–	1	µA	
Ar	Acquisition resolution	–	10	–	bits	
Rrst	Internal $\overline{\text{RST}}$ pull-up resistor	–	–	60	kΩ	

8.4 Timing Specifications

Parameter	Description	Min	Typ	Max	Units	Notes
Fc	Burst center frequency	–	–	–	kHz	Programmable, see Section 6.2.36 on page 42
T_{DW}	Dwell time	–	–	–	ns	Programmable, see Section 6.2.36 on page 42
T_{PW}	Pulse width	–	–	–	ns	Programmable, see Section 6.2.19 on page 33

8.5 I²C-compatible Bus Specifications

Parameter	Operation
Address space	7-bit
Maximum bus speed (SCL)	100 kHz (400 kHz in high speed mode)
Hold time START condition	4 µs minimum (100 kHz), 0.6 µs minimum (400 kHz)
Setup time for STOP condition	4 µs minimum (100 kHz), 0.6 µs minimum (400 kHz)
Bus free time between a STOP and START condition (when the host interfaces with the QT5480)	140 µs minimum (100 kHz), 70 µs minimum (400 kHz)
SDA/SCL rise time	1 µs maximum (100 kHz), 300 ns maximum (400 kHz)

8.6 Power Consumption

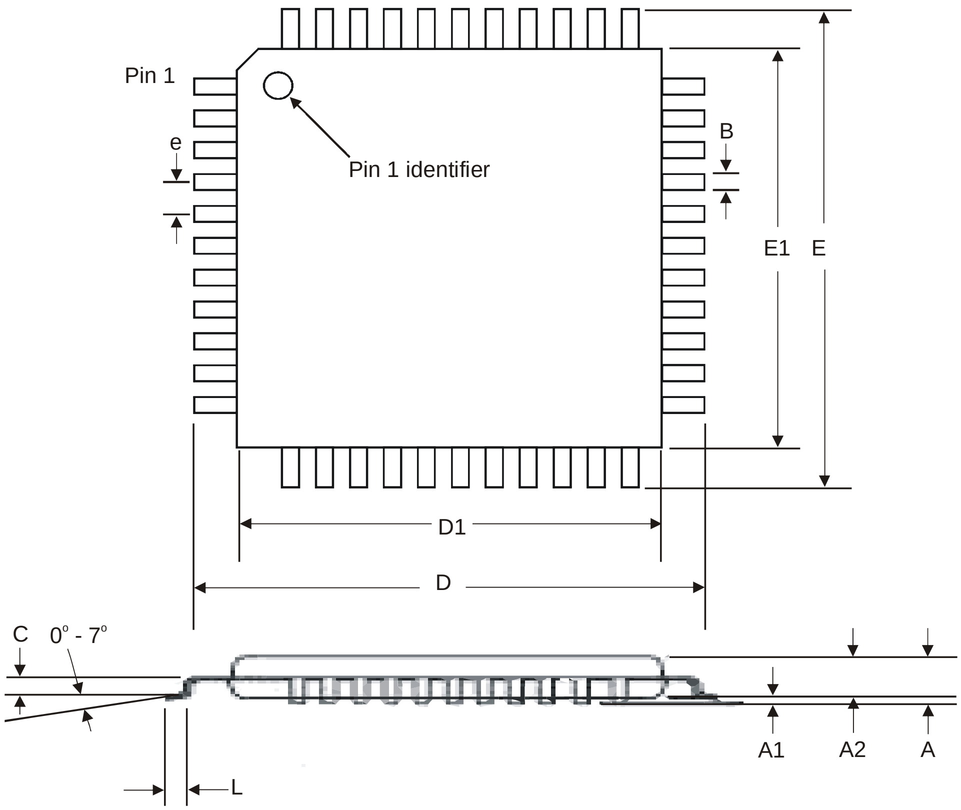
Test condition: 48 keys enabled, Cs = 4.7 nF, Rs = 1 MΩ, BL = 64				
QT5480				
	Low-speed Mode		**High-speed Mode**	
Vdd (V)	**LP Mode**	**Idd (µA)**	**LP Mode**	**Idd (µA)**
1.8V	–	–	–	–
	–	–	–	–
	–	–	–	–
	–	–	–	–
	–	–	–	–
	–	–	–	–
3.3V	SLEEP	<2	SLEEP	<2
	16 ms	3670	16 ms	4900
	32 ms	2680	32 ms	3300
	64 ms	2150	64 ms	2740
	128 ms	1770	128 ms	2460
	255 ms	1600	255 ms	2330

8.7 Mechanical Dimensions

8.7.1 Mechanical Dimensions

The central pad on the underside of the MLF/two-row staggered QFN chip should be connected to ground. Do not run any tracks underneath the body of the chip, only ground.

8.7.2 QT5480 TQFP Mechanical Dimensions

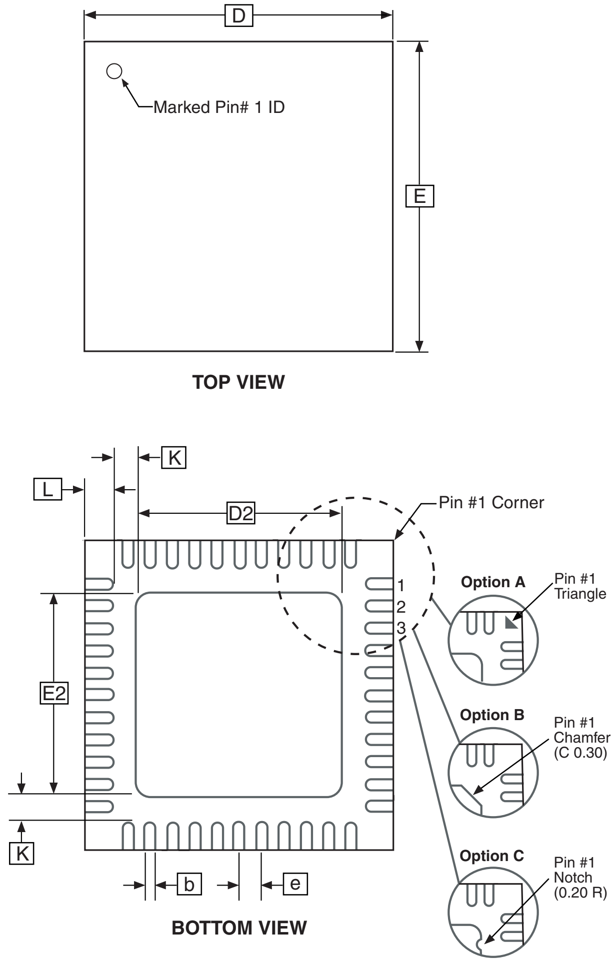


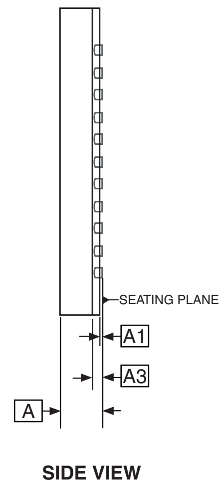
Dimensions for 44-pin TQFP (mm)

Symbol	Minimum	Nominal	Maximum	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	see Note
E	11.75	12.00	12.25	
E1	9.90	10.00	10.10	see Note
B	0.30		0.45	
C	0.09		0.20	
L	0.45		0.75	
e	0.80 Typical			

Note: Dimensions D1 and E1 are maximum plastic body size dimensions including mold mismatch.They do not include mold protrusion.
Allowable protrusion is 0.25mm per side.

8.7.3 QT5480 MLF Mechanical Dimensions





COMMON DIMENSIONS
(Unit of Measure = mm)

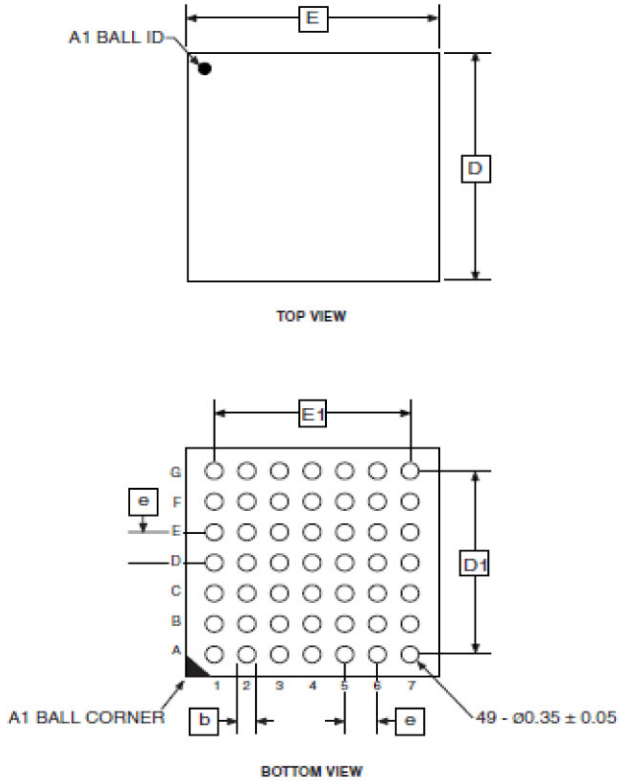
SYMBOL	MIN	NOM	MAX	NOTE
A	0.80	0.90	1.00	
A1	–	0.02	0.05	
A3		0.25 REF		
b	0.18	0.23	0.30	
D	6.90	7.00	7.10	
D2	5.00	5.20	5.40	
E	6.90	7.00	7.10	
E2	5.00	5.20	5.40	
e		0.50 BSC		
L	0.59	0.64	0.69	
K	0.20	0.26	0.41	

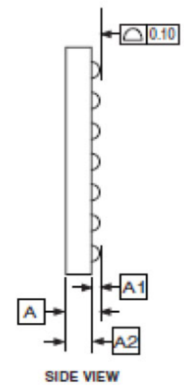
Note: JEDEC Standard MO-220, Fig. 1 (SAW Singulation) VKKD-3.

5/27/06

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**44M1**, 44-pad, 7 x 7 x 1.0 mm Body, Lead Pitch 0.50 mm, 5.20 mm Exposed Pad, Micro Lead Frame Package (MLF)	44M1	G

8.7.4 QT5480 BGA Mechanical Dimensions





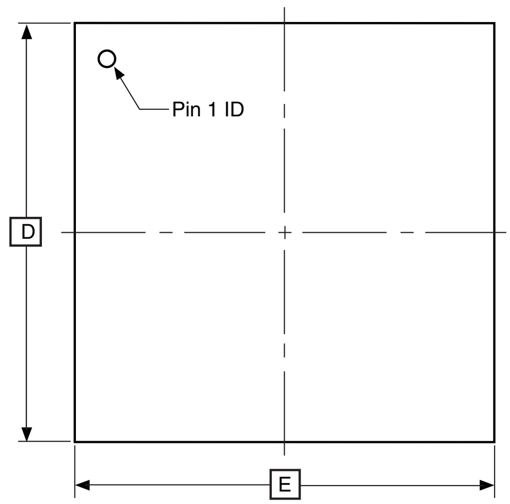
COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.00	
A1	0.20	–	–	
A2	0.65	–	–	
D	4.90	5.00	5.10	
D1	3.90 BSC			
E	4.90	5.00	5.10	
E1	3.90 BSC			
b	0.30	0.35	0.40	
e	0.65 BSC			

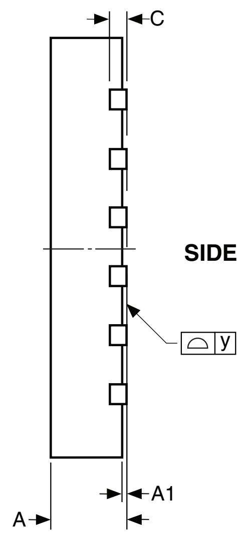
3/14/08

Package Drawing Contact: packagedrawings@atmel.com	TITLE	GPC	DRAWING NO.	REV.
	49C2, 49-ball (7 x 7 Array), 0.65 mm Pitch, 5.0 x 5.0 x 1.0 mm, Very Thin, Fine-Pitch Ball Grid Array Package (VFBGA)	CBD	49C2	A

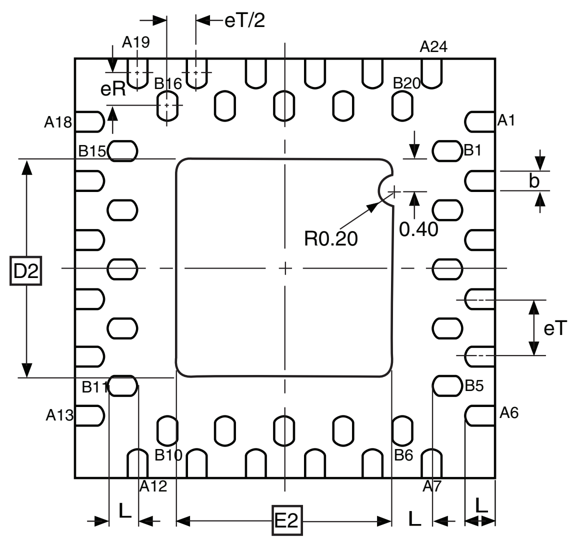
8.7.5 QT5480 Two-row Staggered QFN Mechanical Dimensions



TOP VIEW



SIDE VIEW



BOTTOM VIEW

Note: 1. The terminal #1 ID is a Laser-marked Feature.

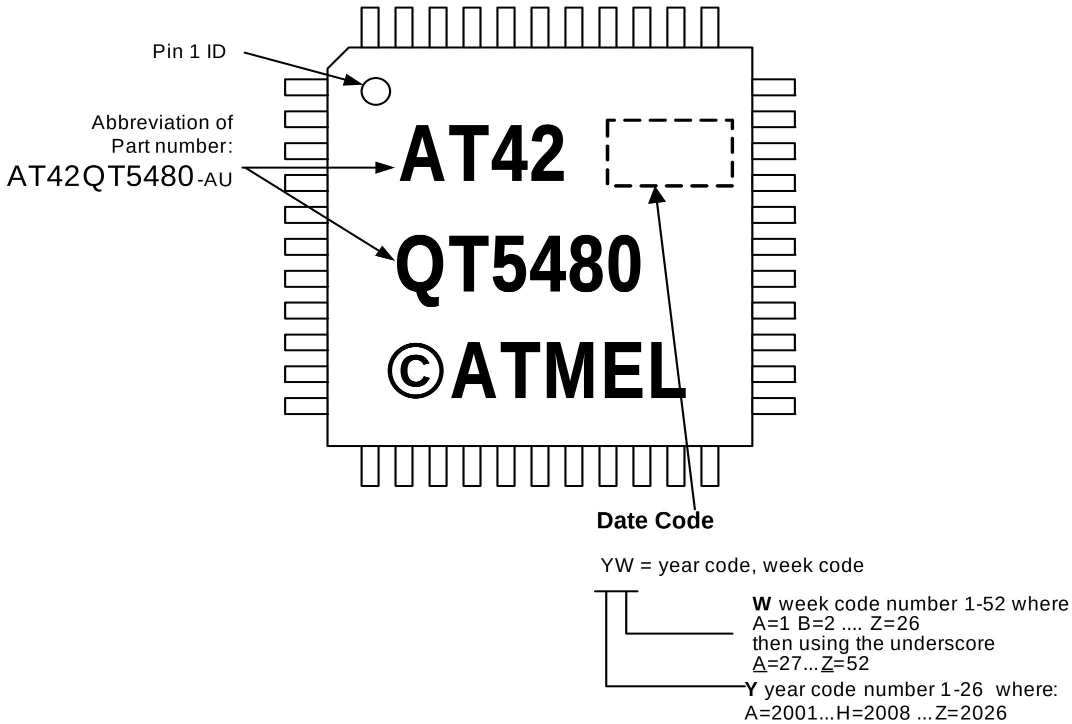
COMMON DIMENSIONS
(Unit of Measure = mm)

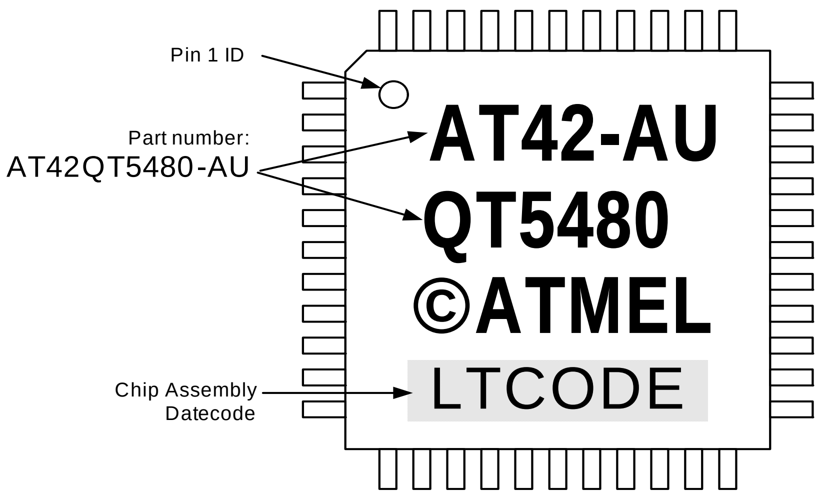
SYMBOL	MIN	NOM	MAX	NOTE
A	0.80	0.90	1.00	
A1	0.00	0.02	0.05	
b	0.18	0.23	0.30	
C		0.20 REF		
D	4.90	5.00	5.10	
D2	2.55	2.60	2.65	
E	4.90	5.00	5.10	
E2	2.55	2.60	2.65	
eT	–	0.70	–	
eR	–	0.40	–	
K	0.45	–	–	
L	0.30	0.35	0.40	
y	0.00	–	0.075	

9/13/07

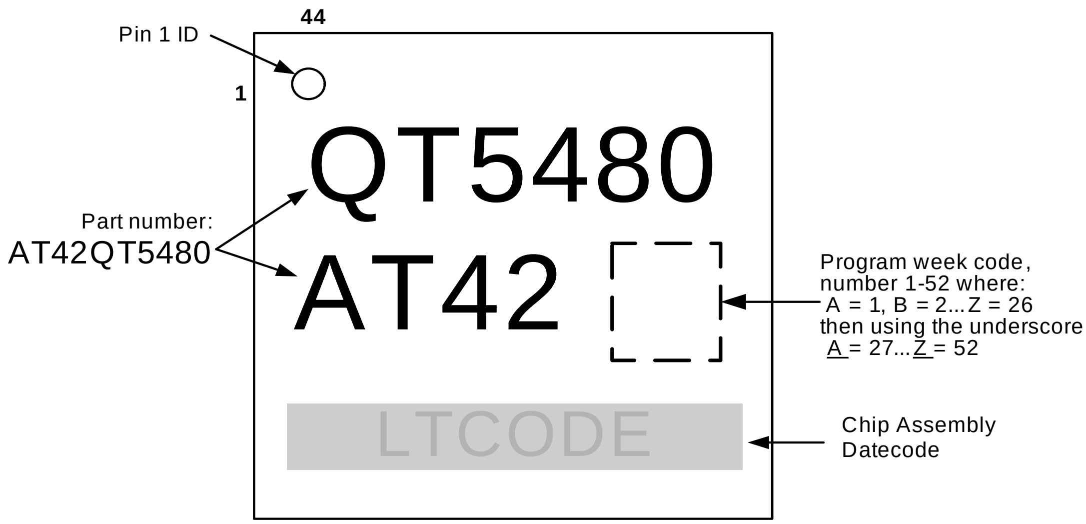
	TITLE	DRAWING NO.	REV.
Package Drawing Contact: packagedrawings@atmel.com	**44MC,** 44QFN (2-Row Staggered), 5 x 5 x 1.00 mm Body, 2.60 x 2.60 mm Exposed Pad, Quad Flat No Lead Package	44MC	A

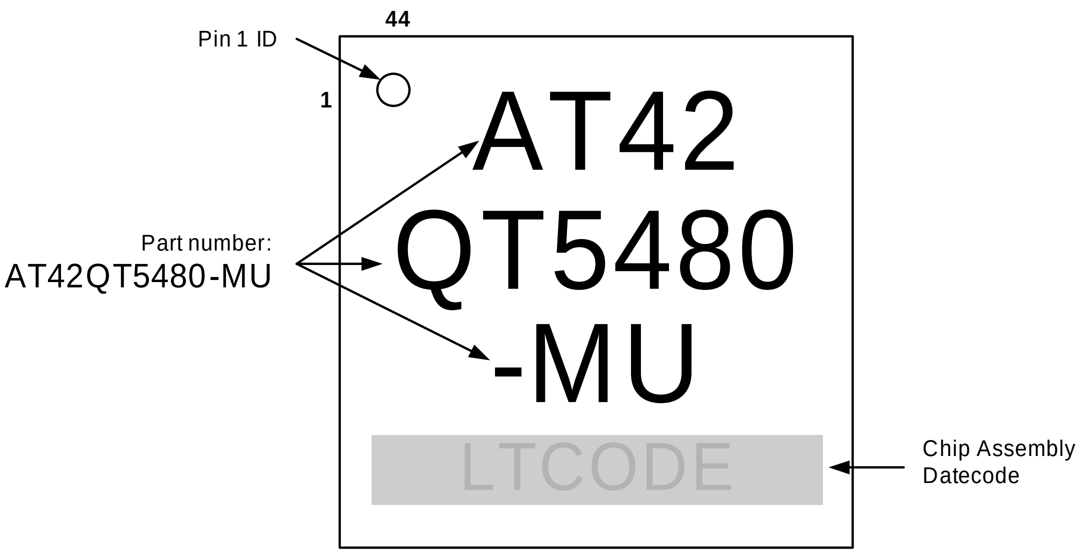
8.7.6 MarkingQT5480 Marking (TQFP IC)



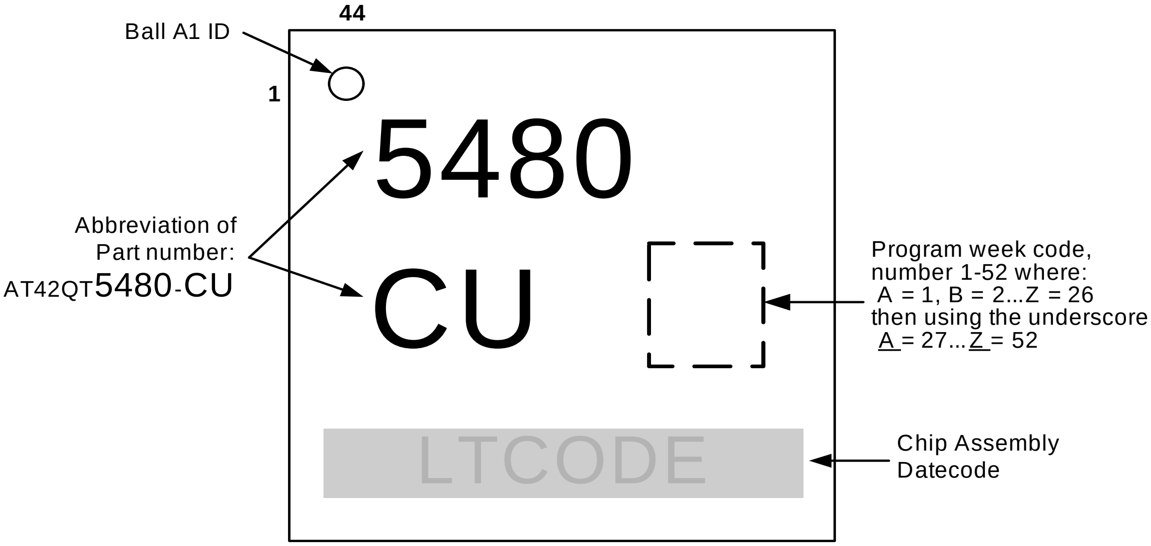


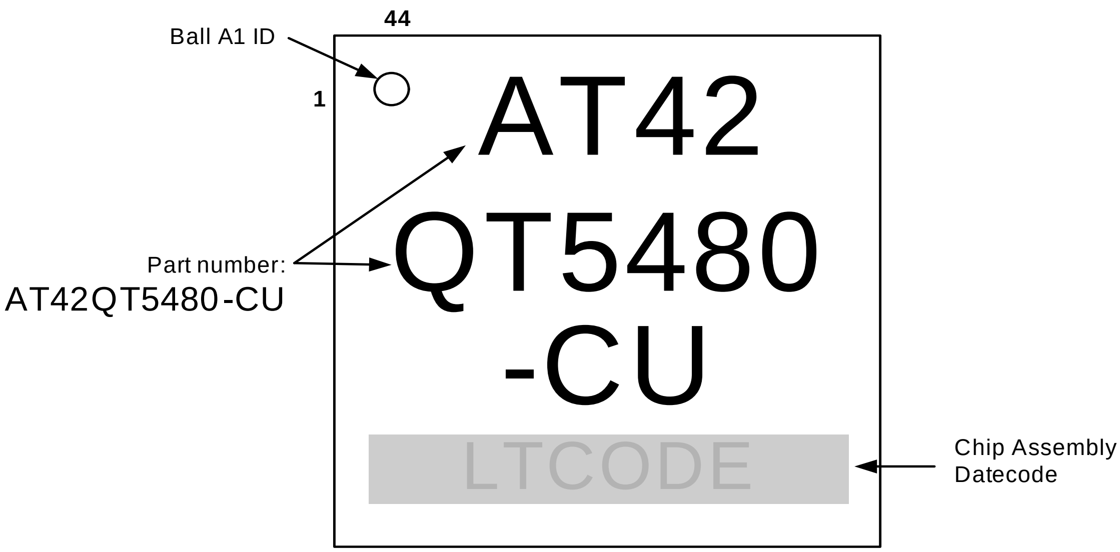
8.7.7 QT5480 Marking (MLF IC)



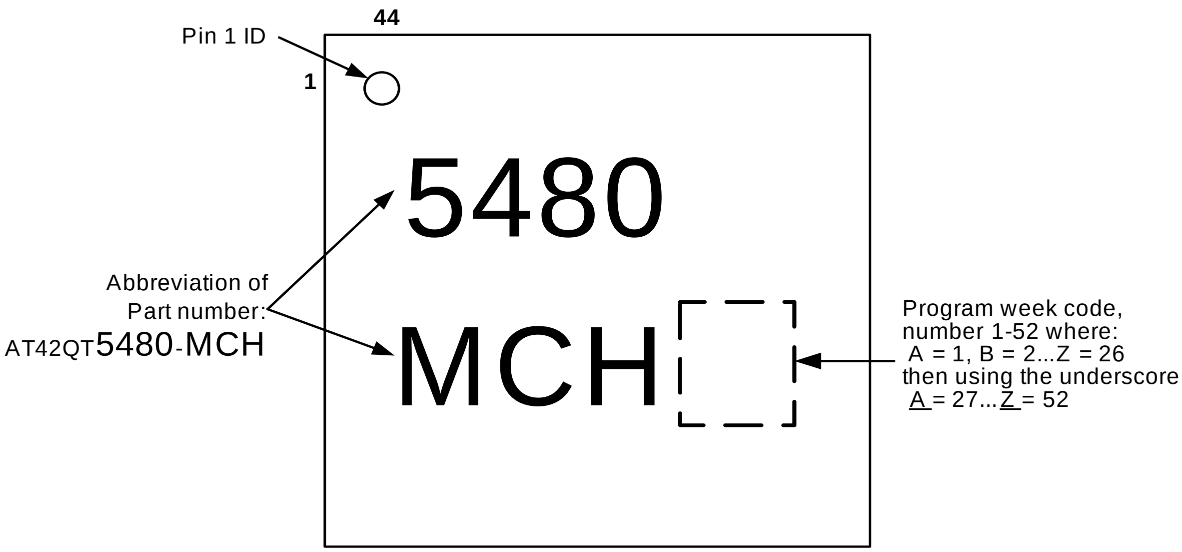


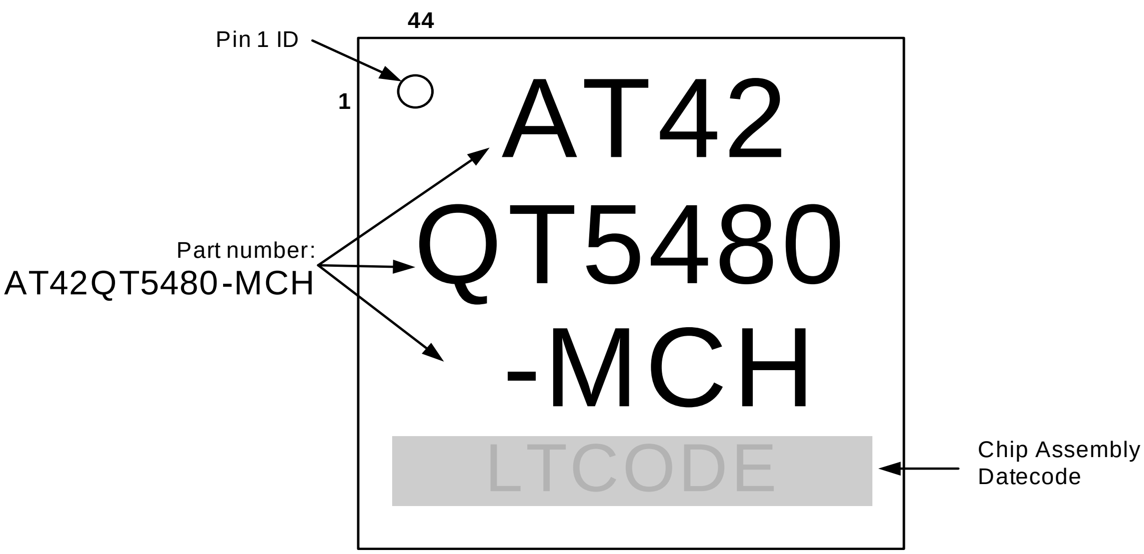
8.7.8 QT5480 Marking (BGA IC)





8.7.9 QT5480 Marking (Two-row Staggered QFN IC)





8.7.10 Part Number

Part Number	Description
AT42QT5480-MU	44-pin 7 x 7 mm MLF RoHS compliant
AT42QT5480-AU	44-pin 10 x 10 mm TQFP RoHS compliant
AT42QT5480-CU	49-ball 5 x 5 mm BGA RoHS compliant
AT42QT5480-MCH	44-pin 5 x 5 mm two-row staggered QFN RoHS compliant

8.8 Moisture Sensitivity Level (MSL)

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

Appendix A. Glossary of Terms

Touchscreen

A two-dimensional arrangement of electrodes whose capacitance changes when touched, allowing the location of touch to be computed in both X and Y axes. The output from the XY computation is a pair of numbers, typically 10-bits each, ranging from 0 to 1023, representing the extents of the touchscreen active region.

Slider

A one-dimensional arrangement of electrodes whose capacitance changes when touched, allowing the location of touch to be computed in one axis.

Key

A simple electrode arrangement whose capacitance changes when touched, allowing touched or not-touched status (on or off) detection.

Node/Channel

This is one of the capacitive measurement points at which the IC can detect capacitive change. A node and a channel are really the same thing, but the term node is used in the context of a touchscreen or slider as it conveys the sense of spatial distribution better.

Object

The IC can combine a group of channels and process them as a slider or touchscreen. These groupings are known as objects.

Adjacent Key Suppression (AKS) Technology

Adjacent Key Suppression (AKS) technology is a method patented by Atmel to help suppress accidental key activations on densely packed keypads. AKS technology can also be used between objects so that, for example, it could prevent a touchscreen from activating when a key is touched.

Interpolation

This is the process of generating intermediate data points from a given discrete subset of known data points. In the context of this datasheet, interpolation can be used in three ways:

- In the computation of the X and Y coordinates for the screen, the numerical algorithm has the effect of interpolating the raw computation result from a starting bit-depth to a higher bit-depth, so increasing the resolution.
- The electrodes are shaped in such a way that the transition from one node to the next node happens in a smooth and progressive way, rather than just a step change (also known as spatial interpolation). Alternatively, the electrodes are connected so that between the nodes there are intermediate electrodes held at intermediate potentials to help smooth and interpolate the output response (also known as resistive interpolation).
- To improve the overall output linearity of the XY response, a process known as piecewise linear interpolation is used that remaps the X and Y coordinates (separately) over a number of discrete intervals. This assumes that the response approximates a straight line in these intervals (also known as PWL interpolation).

Two Touch

The ability of a touchscreen to report two concurrent touches. The touches are reported as two separate sets of XY coordinates.

Jitter

This is the peak-to-peak variance in the reported location for an axis when a fixed touch is applied. Typically jitter is random in nature and has a Gaussian [1] distribution, therefore measurement of peak-to-peak jitter must be conducted over some period of time, typically a few seconds. Jitter is typically measured as a percentage of the axis in question.

For example a 100 x 100 mm touchscreen that shows ±0.5 percent jitter in X and ±1 percent jitter in Y would show a peak deviation from the average reported coordinate of ±0.5 mm in X and ±1 mm in Y. Note that by defining the jitter relative to the average reported coordinate, effects of linearity are ignored.

Resolution

This is a measure of the smallest movement on a slider or touchscreen in an axis that causes a change in the reported coordinate for that axis. Resolution is normally expressed in bits and tends to refer to resolution across the whole axis in question. For example, a slider of length 100 mm and a resolution of 10 bits could resolve a movement of 0.0977 mm. Jitter in the reported position degrades usable resolution.

Linearity

This is a measurement of the peak-to-peak deviation of the reported touch coordinate in one axis relative to the absolute position of touch on that axis. This is often referred to as the nonlinearity. Nonlinearities in either X or Y axes manifest themselves as regions where the perceived touch motion along that axis (alone) is not reflected correctly in the reported coordinate giving the sense of moving too fast or too slow. Linearity is measured as a percentage of the axis in question.

For each axis, a plot of the true coordinate versus the reported coordinate should be a perfect straight line at 45°. A non linearity makes this plot deviate from this ideal line. It is possible to correct modest nonlinearities using on-chip linearization tables, but this correction trades linearity for resolution in regions where stronger corrections are needed (because there is a stretching or compressing effect to correct the nonlinearity, so altering the resolution in these regions). Linearity is typically measured using data that has been sufficiently filtered to remove the effects of jitter. For example, a 100 mm slider with a nonlinearity of ±1 percent reports a position that is, at most, 1 mm away in either direction from the true position.

1. Sometimes called Bell-shaped or Normal distribution.

Appendix B. QMatrix Primer

B.1 Acquisition Technique

QMatrix capacitive acquisition uses a series of pulses to deposit charge into a sampling capacitor, Cs. The pulses are driven on X lines from the controller. The rising edge of the pulse causes current to flow in the mutual capacitance, Cx, formed between the X line and a neighboring receiver electrode or Y line. While one X line is being pulsed, all others are grounded. This leads to excellent isolation of the particular mutual capacitances being measured [1], a feature that makes for good inherent touchscreen performance.

After a fixed number of pulses (known as the burst length) the sampling capacitor's voltage is measured to determine how much charge has accumulated. This charge is directly proportional to Cx and therefore changes if Cx [2] changes. The transmit-receive charge transfer process between the X lines and Y lines causes an electric field to form that loops from X to Y. The field itself emanates from X and terminates on Y. If the X and Y electrodes are fixed directly [3] to a dielectric material like plastic or glass, then this field tends to channel through the dielectric with very little leakage of the field out into free-space i.e. above the panel. However, some proportion of the field does escape the surface of the dielectric and so can be influenced during a touch.

When a finger is placed in close proximity (a few millimeters) or directly onto the dielectric's surface, some of this stray field and some of the field that would otherwise have propagated via the dielectric and terminated onto the Y electrode, is diverted into the finger and is conducted back to the controller chip via the human body rather than via the Y line.

This means that less charge is accumulated in Cs, and hence the terminal voltage present on Cs, after all the charge transfer pulses are complete, becomes less. In this way, the controller can measure changes in Cx during touch. This means that the measured capacitance Cx goes down during touch, because the coupled field is partly diverted by the touching object.

The spatial separation between the X and Y electrodes is significant to make the electric field to propagate well in relation to the thickness of the dielectric panel.

For touchscreen electrode designs the situation is more complex.

B.2 Sample Capacitor Saturation

Cs voltage saturation at a pin YnB is shown in Figure B-1 on page 65. Saturation begins to occur when the voltage at a YnB pin becomes more negative than -0.25V 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. 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, more separation of the X and Y lines on the PCB, increasing Cs, and decreasing the burst length.

1. A common problem with other types of capacitive acquisition technique when used for touchscreens, is that this isolation is not so pronounced. This means that when touching one region of the screen, the capacitive signals also tend to change slightly in nearby channels too, causing small but often significant errors in the reported touch position.
2. To a first approximation.
3. Air gaps in front of QMatrix sensors massively reduce this field propagation and kill sensitivity. Normal optically clear adhesives work well to attach QMatrix touchscreens to their dielectric front panel.

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 more negative than -0.25 volts during any burst (not counting overshoot spikes which are probe artifacts), there is a potential saturation problem.

Figure B-2 on page 65 shows a defective waveform similar to that of Figure B-1 on page 65, 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. This phenomenon is more subtle; it can be best detected by increasing the burst length (BL) to a high count and watching what the waveform does as it descends towards and below -0.25V. The waveform appears deceptively straight, but it slowly starts to flatten even before the -0.25V level is reached.

A correct waveform is shown in Figure B-3 on page 66. Note that the bottom edge of the bottom trace is substantially straight (ignoring the downward spikes).

Unlike other QT circuits, the Cs capacitor values on QT5480 devices have no effect on conversion gain. However, they do affect conversion time.

Unused Y lines should be left open.

Figure B-1. VCs – Nonlinear During Burst
(Burst too long, or Cs too small, or X-Y transcapacitance too large)

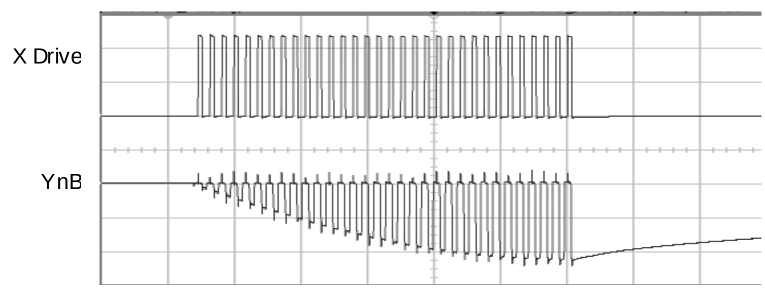


Figure B-2. VCs – Poor Gain, Nonlinear During Burst (Excess capacitance from Y line to Gnd)

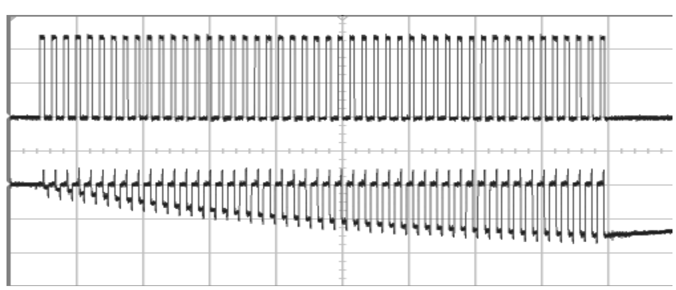


Figure B-3. VCs – Correct

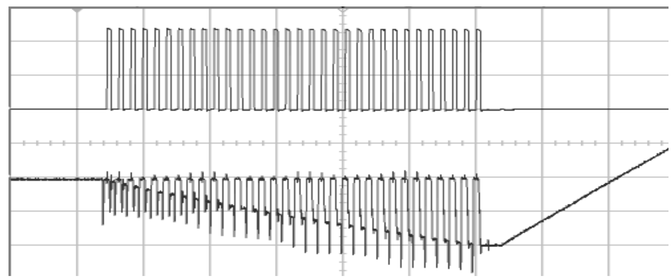
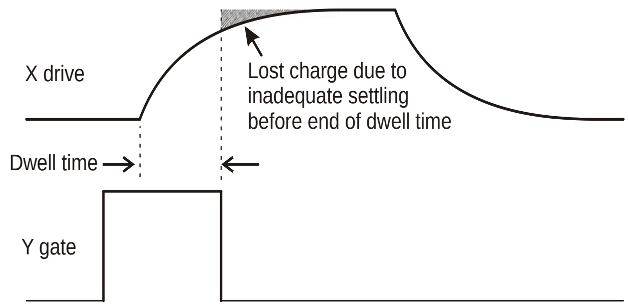


Figure B-4. Drive Pulse Roll-off and Dwell Time



Note: The Dwell time is a minimum of ~250 ns - see Section 5.12.5 on page 22

B.3 Moisture Resistance

A useful side effect of the QMatrix acquisition method is that placing a floating conductive element between the X and Y lines tends to increase the field coupling and so increases the capacitance Cx. This is the opposite change direction to normal touch, and so can be quite easily be ignored or compensated for by the controller. An example of such floating conductive elements is the water droplets caused by condensation.

As a result, QMatrix-based touchscreens tend not to go into false detect when they are covered in small non-coalesced water droplets. Once the droplets start to merge however, they can become large enough to bridge the field across to nearby ground return paths (for example, other X lines not currently driven, or ground paths in mechanical chassis components). When this happens, the screen's behavior can become erratic.

There are some measures used in these controllers to help with this situation, but in general there comes a point where the screen is so contaminated by moisture that false detections become inevitable. It should also be noted that uniform condensation soon becomes non-uniform once a finger has spread it around. Finger grease renders the water highly conductive, making the situation worse overall.

In general, QMatrix has industry leading moisture tolerance but there comes a point when even the best capacitive touchscreen suffers due to moisture on the dielectric surface.

B.4 Series Resistance

If the total accumulated series resistance in the X line gets large (typically over the high tens of kΩ) then the RC time constant formed in combination with the parasitic capacitance of the electrode by virtue of its proximity to say a ground layer or LCD [1], can become so large that the rising edge used to cause current to flow can become very slow and so require excessively long durations to settle (think of a settled X line as having transferred all possible charge via Cx once approximately five time constants have elapsed).

The resistance in the Y lines also causes an RC time constant that can be equally problematic in capturing all charge from the X line. Again, resistance build up above high tens of kΩ can be problematic.

In general, for keys and sliders, this is not an issue. For touchscreens, the electrode pattern designs used by Atmel take this resistance build-up into account and seek to minimize it when possible. While acquisition times can be increased to mitigate the RC time constant effects, this can lead to poor response times. Hence reducing resistance is a goal in all of Atmel's designs. Adding deliberate extra series resistance in Y lines can have a beneficial effect on conducted noise susceptibility, but care must be exercised when doing this and Atmel should always be consulted during design.

B.5 Typical Waveforms

During testing of a QMatrix design it is very important to check the waveforms present on the Y lines near the controller. This is to ensure that the final configuration of the touch sensor, circuit components and layout allows for correct charge-transfer operation. See Appendix B on page 64 for some typical waveforms.

The charge transfer process can be affected by the series resistance of the electrode and connections (for example, the electrodes themselves, tracking, series resistors) and also by parasitic capacitances on the Y lines and Y electrodes.

A secondary effect that can occur (rarely) is that enough voltage is accumulated on the sampling capacitor that the internal ESD protection diode on the controller's IO pin starts to partially conduct and so corrupt the capacitive measurements. This happens if the terminal voltage on Cs becomes larger than around -0.25V. For most touchscreen applications, this is almost impossible to achieve, but for keys and sliders with very high gains, it is more of a concern. When this effect happens, the apparent gain of the sensor reduces, rendering keys etc totally insensitive.

B.6 Measurement Variance

There is some level of chip-to-chip variance in the measurements obtained. This does not mean that sensitivity changes from chip-to-chip, but it can mean that if the measurement references are used as a production test method, then part of the reference distribution observed is attributed to the controller chip itself. For a normal setup, this variance normally accounts for less than ±20 percent variance in the references values. It should also be noted that in general, operating at a higher Vdd serves to reduce this variance source.

Other factors also add to the overall variance, such as the sample capacitor tolerance (although the effect of Cs on references is a second order effect). More information is available on request.

1. And all the parallel Cx mutual capacitances that are at any instant not being sampled.

B.7 Component Sensitivity

In general, the passive components used as part of the measuring circuit are non-critical in terms of absolute tolerance and temperature coefficient with one exception; the sample capacitors. Always use good quality sample capacitors; X7R dielectric or better.

All resistors used can be simple low cost SMT types of 5 percent tolerance or better and generally 100 ppm/0°C or better.

A good quality voltage regulator on Vdd is recommended; one with good transient response on transitioning from very low Idd (<100 µA) to normal Idd (a few mA). The regulator should settle when presented with this load change in under 100 µs. The tolerance of the regulator used should ideally be 5 percent or better.

B.8 Interference Sources

B.8.1 Power Supply

See Section 8.2 on page 51 for the power supply range. The device can tolerate short-term power supply fluctuations. If the power supply fluctuates slowly with temperature, the device tracks and compensate 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 the 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 LEDs, 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.

See the text underneath Figure 1-1 on page 5 for suggested regulator manufacturers.

Noise on Vdd can appear directly in the measurement results. Vdd should be checked to ensure that it stays within specification in terms of noise, across a whole range of product operating conditions.

Ceramic bypass capacitors on Vdd, placed very close (<5 mm) to the chip are recommended. A bulk capacitor of at least 1 µF and a higher frequency capacitor of around 10 nF to 100 nF in parallel are recommended.

B.8.2 LCD Drive

Switching noise from some LCDs can have a strong effect on the measurement results. Further information can be obtained on request.

B.8.3 LED Drive

LEDs and their drive circuitry can change impedance between on and off states and so present varying loads to capacitive sensors if those circuits are in close proximity. Further information can be obtained on request.

B.8.4 Other Noise Sources

One of the most challenging noise sources for capacitive sensors comes from line-induced voltages on non-earthed products. This noise type is sometimes referred to as earth-referred noise and is generally caused by power supplies that have a fully isolated primary-to-secondary design with very low leakage impedance across this isolation boundary. The net result is that highly distorted 50 or 60Hz voltage noise, sometimes of hundreds of volts peak-to-peak, can appear on the connected product relative to earth.

This noise is fully common mode to the capacitive sensor (and hence not an issue) until it is touched, whereupon part of it becomes differential and can be seen in the measurement results. A combination of techniques are used in these sensor chips to deal with this noise source. A level of parameter tuning is normally required to optimize the system's performance. Further information can be obtained on request.

B.9 Typical QMatrix Waveforms

Figure B-5. Probing the X Line and YnB Pin

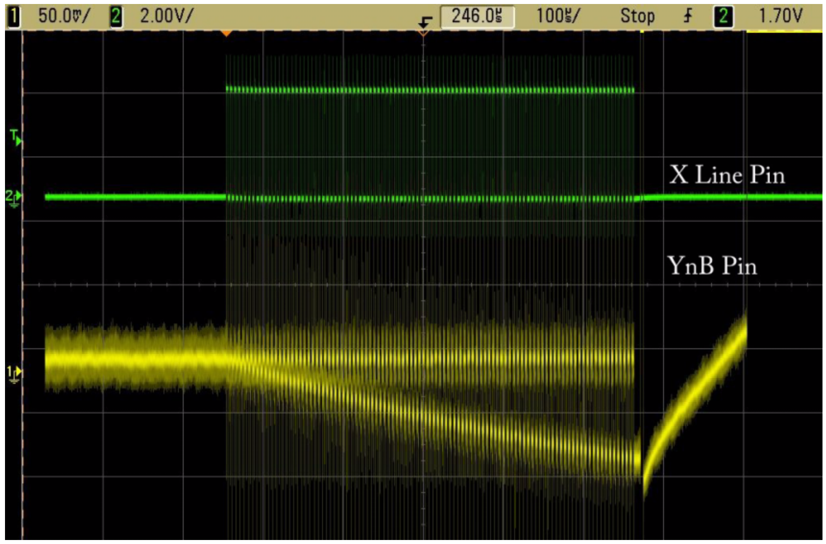
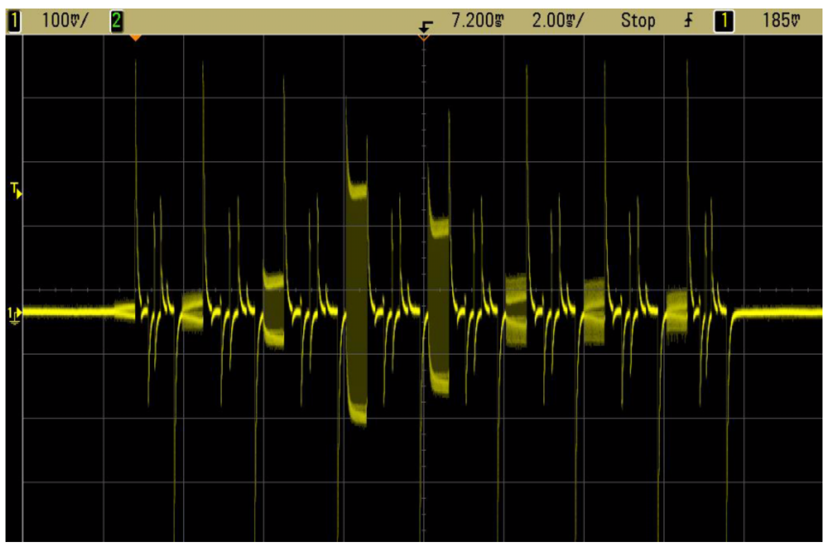


Figure B-6. Probing on Top of Screen With a Coin

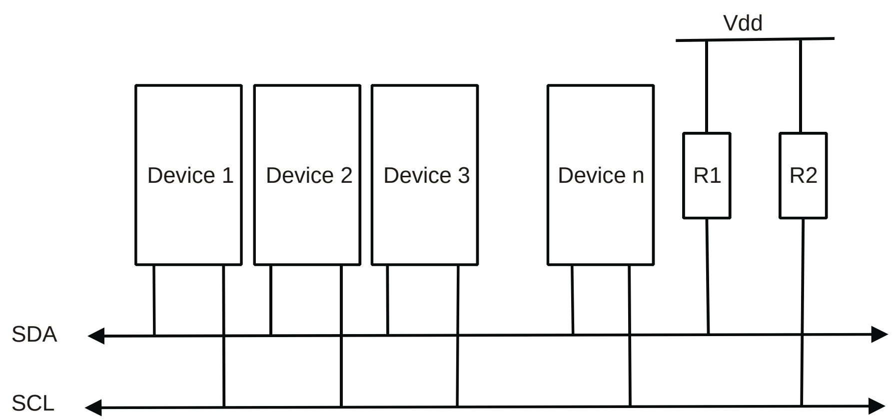


Appendix C. I²C Basics (I²C-compatible Operation)

C.1 Interface Bus

The device communicates with the host over an I²C-compatible bus. The following sections give an overview of the bus; more detailed information is available from www.i2C-bus.org. Devices are connected to the I²C-compatible bus as shown in Figure C-1. Both bus lines are connected to Vdd via pull-up resistors. The bus drivers of all I²C-compatible devices must be open-drain type. This implements a wired "AND" function that allows any and all devices to drive the bus, one at a time. A low level on the bus is generated when a device outputs a zero.

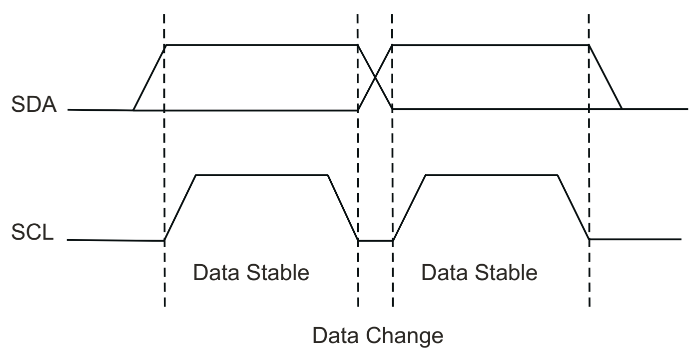
Figure C-1. I²C-compatible Interface Bus



C.2 Transferring Data Bits

Each data bit transferred on the bus is accompanied by a pulse on the clock line. The level of the data line must be stable when the clock line is high; the only exception to this rule is for generating START and STOP conditions.

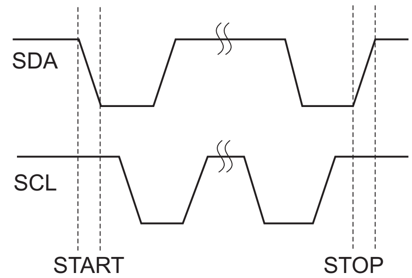
Figure C-2. Data Transfer



C.3 START and STOP Conditions

The host initiates and terminates a data transmission. The transmission is initiated when the host issues a START condition on the bus, and is terminated when the host issues a STOP condition. Between the START and STOP conditions, the bus is considered busy. As shown in Figure C-3, START and STOP conditions are signaled by changing the level of the SDA line when the SCL line is high.

Figure C-3. START and STOP Conditions

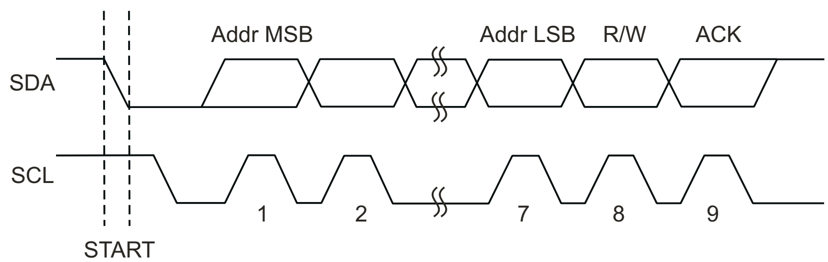


C.4 Address Byte Format

All address bytes are 9 bits long, consisting of 7 address bits, one READ/WRITE control bit and an acknowledge bit. If the READ/WRITE bit is set, a read operation is performed, otherwise a write operation is performed. When the device recognizes that it is being addressed, it acknowledges by pulling SDA low in the ninth SCL (ACK) cycle. An address byte consisting of a slave address and a READ or a WRITE bit is called SLA+R or SLA+W, respectively.

The most significant bit of the address byte is transmitted first. The address sent by the host must be consistent with that selected with the option jumpers.

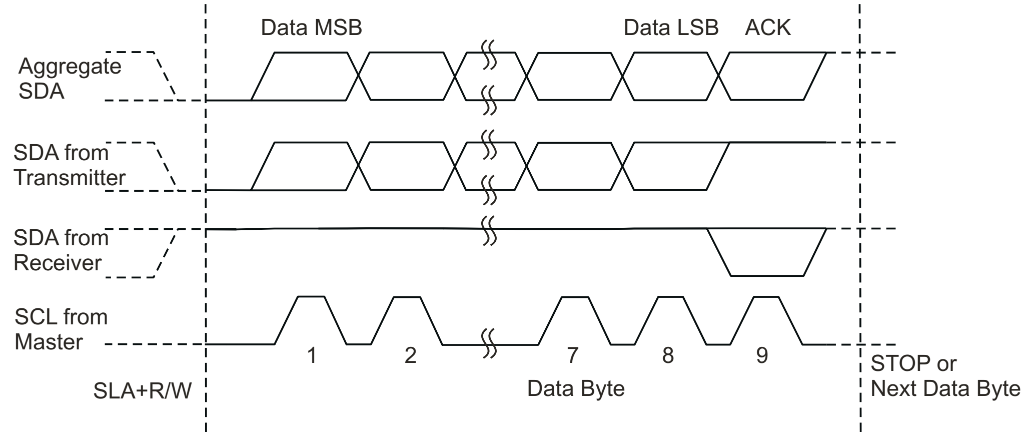
Figure C-4. Address Byte Format



C.5 Data Byte Format

All data bytes are 9 bits long, consisting of 8 data bits and an acknowledge bit. During a data transfer, the host generates the clock and the START and STOP conditions, while the Receiver is responsible for acknowledging the reception. An acknowledge (ACK) is signaled by the Receiver pulling the SDA line low during the ninth SCL cycle. If the Receiver leaves the SDA line high, a NACK is signaled.

Figure C-5. Data Byte Format



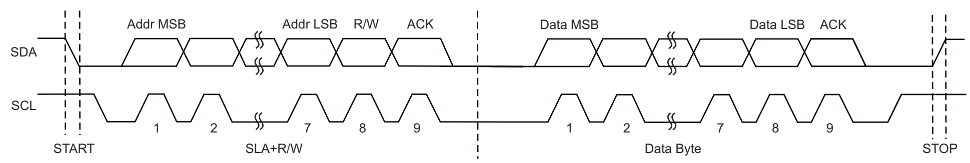
C.6 Combining Address and Data Bytes into a Transmission

A transmission consists of a START condition, an SLA+R/W, one or more data bytes and a STOP condition. The wired “ANDing” of the SCL line is used to implement handshaking between the host and the device. The device extends the SCL low period by pulling the SCL line low whenever it needs extra time for processing between the data transmissions.

Note: Each write or read cycle must end with a stop condition. The device may not respond correctly if a cycle is terminated by a new start condition.

Figure C-6 shows a typical data transmission. Note that several data bytes can be transmitted between the SLA+R/W and the STOP.

Figure C-6. Byte Transmission



Appendix D. Touchscreen Linearization

The raw data is obtained from the device using a plotter. The screen is scanned across the axis to be linearized, and this is repeated through the whole screen.

Raw data obtained are the plotter position and reported position. The reported position bytes are combined to get a 10-bit value.

1. Scale the plotter position to a 10-bit value to match the 10-bit reported position using the following formula:

   Scaled Plotter Position = (PlotterPosition / number of measurement points) x 1023
2. Average the reported position for each data measurement point, to get an average reported position curve over the entire screen.
3. Determine the offset value which will be the new zero point from which the screen's touch position will be reported. The offset value will be a point from the reported position that is stable and above the minimum reported value.
4. Work out the interval points, starting from the offset value. Using an interval width of 64, the value of the interval points will be:
   a. offset
   b. offset + 64
   c. offset + (64 x 2)...
   d. offset + (64 x n)

   The last interval point (n) should be below the maximum reported position.
5. Get the corresponding plotter positions of these interval points.

   As the interval point might not fall on the measurement points, a little interpolation is required to determine the corresponding plotter value. The measurement points that the corresponding plotter position will fall into are determined by the following formula:

$$Y = mX + c$$

$$m = \frac{PlotterPosition(H) - PlotterPosition(L)}{ReportedPosition(H) - ReportedPosition(L)}$$

$$X = ReportedPosition(n) - ReportedPosition(L)$$

$$c = PlotterPosition(L)$$

   Therefore, the corresponding plotter positions can be calculated as follows:

$$CorrespondingPlotterPosition(n) = Y = mX + c$$

   where:

   n = interval point
   H = upper measurement point
   L = lower measurement point
6. Get the difference of the plotter positions between these interval points by using the following formula:

$$Difference = PlotterPosition(n) - PlotterPosition(n-1)$$

   where **n** is interval point.

7. Scale the difference of the plotter position between the intervals to 10 bits (0 to 1023). These values will be the coefficients for the linearization algorithm.

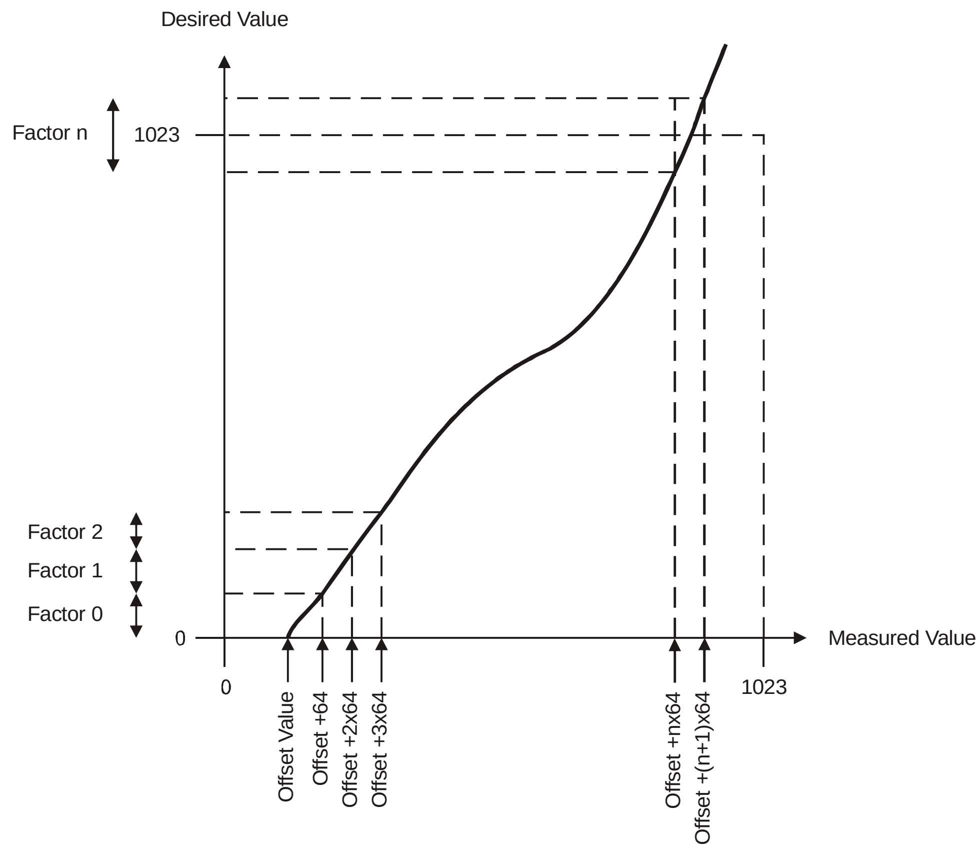
   Use the following formula:

$$Coefficient(n) = \frac{DiffPlotterPosition(n)}{SumOfDiffPlotterPosition} \times 1023$$

   where:

   n= coefficient number
   DiffPlotterPosition = Difference of plotter position between intervals
   SumofDiffPlotterPosition = Sum of all difference of plotter position between intervals.

Figure D-1. Linearization Coefficients



Appendix E. Checksum Calculation

When the QT5480 initializes after power-on/reset it configures itself using the configuration settings that it holds in its internal EEPROM, and also generates the checksum value of these stored settings. This checksum is generated so that the host controller is able to quickly identify if these EEPROM settings are not as intended. Example C code (shown below) allows the designer to generate the checksum of a set of configuration values so that this value may then be embedded into the host controller for comparison with that generated by the IC.

```
/* Function Prototype */
uint16_t calculate_crc(uint8_t ch, uint16_t oldCRC);



/* Function Implementation */
/* This function will generate the new value of the old checksum combined
 * with the new input byte */
uint16_t calculate_crc(uint8_t ch, uint16_t oldCRC)
{
  static const uint16_t crcPoly = 0x8005;
  uint16_t i;
  uint32_t result;

  result = ((uint32_t)oldCRC << 8) | ch;

  for (i = 0; i < 8; i++)
  {
    if ((result <<= 1) & 0x1000000)
    {
      result ^= ((uint32_t)crcPoly << 8);
    }
  }
  return (uint16_t)(result >> 8);
}
```

```
/* Example Calling Routine */
/* This function will take the default configuration settings of an
 * AT42QT5320/5480 and generate the same checksum as the IC would. This
 * is useful for generating a reference checksum value for embedding into
 * the host controller so that it is able to verify that the IC has the
 * correct settings stored - in this instance the checksum should be 0xEC2D
*/
uint16_t calculate_config_checksum(void)
{
 uint16_t i;
 uint16_t CRC_val = 0;
 uint8_t setup_data[217] = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,16,50,0,48,0,0,0,0,0,0,0,0,0,0,0,0,0,0,20,5,4,2,150,
20,255,0,0,0,0,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,64,0,0,64,64,
64,64,64,64,64,64,64,64,64,64,64,64,64,64,0,255,8,3,0,0,0,0,0};

  for(i = 0; i < sizeof(setup_data); i++)
  {
    CRC_val = calculate_crc(setup_data[i], CRC_val);
  }

  return(CRC_val);
}
```

Revision History

Revision No.	History
Revision AX – October 2008	• Initial release for chip revision 5.0
Revision BX – November 2008	• Added sync bits • Added two-row staggered QFN package
Revision CX – December 2008	• Event timestamp information added to addresses 16 – 25 • Changes to Sync mode in address 659 • Checksum calculation amended

Contents

Headquarters

Atmel Corporation
2325 Orchard Parkway
San Jose, CA 95131
USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

International

Atmel Asia
Unit 01-05 & 16, 19/F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong
Kowloon
Hong Kong
Tel: (852) 2245-6100
Fax: (852) 2722-1369

Atmel Europe
Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-
Yvelines Cedex
France
Tel: (33) 1-30-60-70-00
Fax: (33) 1-30-60-71-11

Atmel Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Touch Technology Division
1 Mitchell Point
Ensign Way
Hamble
Southampton
Hampshire SO31 4RF
United Kingdom
Tel: (44) 23-8056-5600
Fax: (44) 23-8045-3939

Product Contact

Web Site
www.atmel.com

Technical Support
touch@atmel.com

Sales Contact
www.atmel.com/contacts

Literature Requests
www.atmel.com/literature