# Guide Contents

**Overview**

- Momentary 5
- Toggle 6
- 5-Pad Momentary 6

**Assembly and Wiring**

- Installing the Headers: 8
- Position the header strips 8
- Position the breakout 8
- And Solder 9

**Wiring for Toggle and Momentary**

- Toggle Operation 11
- Momentary Operation 12

**Other Options:**

- LED Control 13
- Speed vs. Power (Momentary Only) 13
- Timer (Toggle Only) 14
- Connecting to your Circuit. 14
- Simple Motor Control 15

**Wiring for 5-pad Momentary**

**Adding Custom Touch Pads**

- Wire, Thread, Foil, Fabric, Paint 19
- Connections: 21
- Sensor Pads 22

**Build a Touch Control Panel**

- Design your panel 23
- Cut the touch-pads 24
- Attach the touch-pads 24
- Attach the Wires 25
- Adjust the Wires 26

© Adafruit Industries  https://learn.adafruit.com/adafruit-capacitive-touch-sensor-breakouts
Install the Panel
And Test
Connect to your Circuit
Downloads
Overview

These breakout boards are a simple way to add capacitive touch to your project. Just power with 1.8 to 5.5VDC and touch the pad to activate the sensor. These touch switches interface easily to any project - with or without a microcontroller.

When a capacitive load (such as a human hand) is in close proximity to the sense-pad, the sensor detects the change in capacitance and activates the switch. Custom sense-pads can be made from nearly any conductive material and these sensors can detect touch through thin layers of non-conductive materials such as glass, plastic, fabric or even wood.

The breakouts come in three styles:
Momentary
This sensor has a built-in sense-pad and is active for as long as the sensor area is touched.

The sense-pad can be extended with wire and almost any conductive material.
Toggle
This sensor also has a built-in sense-pad. It becomes active when touched and remains active until it is touched again.

As with the momentary sensor, the sense-pad can be extended with wire and almost any conductive material.

5-Pad Momentary
This version combines 5 momentary switches into one breakout. There are 5 pins for attaching wires to up to 5 external sensor pads.
We also have a new 12-Key version with it's own tutorial over here! (http://adafruit.it/dKH)
Assembly and Wiring

These breakouts come fully assembled. For use in a breadboard, you may want to take a couple minutes to install the included header strips:

Installing the Headers:
Install the headers by following these 3 easy steps. The photographs below show one of each sensor type.

Position the header strips
Plug them long-pins down into a breadboard to stabilize them for soldering.

Position the breakout
Place the breakout board over the header pins.
And Solder

Solder each pin for solid electrical contact.
Wiring for Toggle and Momentary

These two breakouts are very similar and can be powered by anything from 1.8V to 5.5V DC. Simply connect Ground to GND and the positive voltage to VDD. The standalone sensors are fully functional without further connections.
**Toggle Operation**

The Toggle version of the sensor turns on when you touch it once, then turns off when you touch it again. The on-board LED indicates the state of the switch.
Momentary Operation
The momentary touch sensor works just like a momentary switch. It is on when you touch it and off when you move away. The on-board LED indicates the state of the switch.

Other Options:
These sensors have several jumper configurable operating modes as described below:
LED Control
The LED indicators can be disabled for ultra-low power applications. To disable the LED, simply cut the jumper between the pads where indicated on the back of the breakout board.

With the jumper cut, the LED can be controlled externally via the LED pin on the header.

Speed vs. Power (Momentary Only)
The Momentary version can be configured for "Fast" mode (default) or low-power mode. Fast mode requires 0.5mA. Low Power mode requires just 50uA. To switch between the two, cut the jumper on one side of the "mode" pads and bridge the other side with a drop of solder.
Timer (Toggle Only)
By default, the toggle sensor is configured for infinite time-out. It will stay on until you touch the sensor to turn it off. It also supports a configurable time-out to turn off the output automatically after a delay. To select this mode, cut the 'TIMER' jumper and connect a resistor & capacitor to the TIME pin. For a circuit diagram and resistor/capacitor calculations, see page 13 of the datasheet (http://adafruit.it/cgW).

You can also just connect TIME to Vdd and the chip will turn off approx 15 minutes after being turned on. Connect TIME to OUT and the chip will time-out approx one hour after being turned on.

Connecting to your Circuit.
The outputs of these touch switches are 'active high'. Use them like a positive logic signal, or a pushbutton that shorts to VCC.
Simple Motor Control
You can use it just like a pushbutton or logic signal with a transistor or MOSFET to drive high-current loads like a DC motor.
The 5-pin momentary breakout can be powered with anything from 1.8V to 5.5VDC. Just connect ground to GND and the positive voltage to VDD. This sensor does not have built-in touch pads, but you can create your own pads in any size or shape from wire, foil or any other conductive material. Simply connect your touch-pads to each of the 5 sense pins. When you touch the pad, the corresponding LED on the other side will light up. The chip only detects one touch-pad at a time to prevent false readings.

5 touch inputs in one device makes this the perfect component for building your own custom capacitive touch panel!
Adding Custom Touch Pads

Custom touch pads are easy to make. You can use almost any conductive material:
Wire, Thread, Foil, Fabric, Paint
If it will conduct electricity, it will work as a touch sensor!
Connections:
The Toggle and Momentary boards have a solder hole located just below the sensor pad for attaching a wire to an external sensor.

The 5-pad breakout has pins numbered 0-4 on the left side of the board.
Sensor Pads
Attach the connecting wire to any conductive object or surface. That surface will become touch sensitive. Larger surfaces tend to be more sensitive. You will be able to sense through fabric, plastic glass and many other non-conductive materials.

Note that the wire will be touch sensitive too! Be sure to route any connecting wires away from areas where they might create an accidental touch input.
Build a Touch Control Panel

Capacitive touch sensors are a great way to add external controls to a waterproof enclosure. There is no need to drill holes or worry about gaskets and O-rings. These sensors will detect your touch right through the plastic case!

Design your panel
You can draw it by hand, or with your favorite drawing tool and print it on some heavy card-stock.
Cut the touch-pads
Cut pads from copper tape. About 1/2" square is a good size for buttons on a touch-pad.

Attach the touch-pads
Peel the release paper from the back of the copper tape and stick the touch-pads to be back of the panel so that they align with the buttons on the front.
Attach the Wires
Solder wires to the copper touch-pads. For this example, I used a 6-conductor 0.1" socket cable (http://adafru.it/206) with one end cut off to simplify connections to the breakout.
Adjust the Wires
Bend the wires away from the panel. The wires will be touch-sensitive too. To prevent accidental false touches, we want to keep them away from the panel surface.
Install the Panel
Tape the panel to the inside of the polycarbonate cover using clear packing tape.
And Test
Connect the cable to the breakout. Power it up and test your control panel. Touching each button should cause a different LED to light up.
Connect to your Circuit
The output signals are 'active low', so they can replace any pushbutton that shorts to ground - such as the buttons on the RGB LCD shield. You can leave off the buttons and solder directly to the circled pads, or (if your shield is already built), just 'tack-solder' the wires to the legs of the buttons.

When you put it all together, you will have a completely sealed, touch sensitive control panel!
Downloads

Schematics for '1010 and '1012 breakouts (click to enlarge)

Higher val C1 increases sensitivity but more power and slower reaction time. Value should be in the 0.1-50uF range.
Features

• Configurations:
  – Comms mode
  – Standalone mode

• Number of Keys:
  – Comms mode – 1 to 7 keys (or 1 to 6 keys plus a Guard Channel)
  – Standalone mode – 1 to 4 keys plus a fixed Guard Channel on key 0

• Number of I/O Lines:
  – Standalone mode – 5 outputs

• Technology:
  – Patented spread-spectrum charge-transfer

• Key Outline Sizes:
  – 6 mm x 6 mm or larger (panel thickness dependent); widely different sizes and shapes possible

• Layers Required:
  – One

• Electrode Materials:
  – Etched copper
  – Silver
  – Carbon
  – Indium Tin Oxide (ITO)

• Panel Materials:
  – Plastic
  – Glass
  – Composites
  – Painted surfaces (low particle density metallic paints possible

• Panel Thickness:
  – Up to 10 mm glass (electrode size dependent)
  – Up to 5 mm plastic (electrode size dependent)

• Key Sensitivity:
  – Comms mode – individually settable via simple commands over I²C-compatible interface
  – Standalone mode – settings are fixed

• Interface:
  – I²C-compatible slave mode (400 kHz). Discrete detection outputs

• Power:
  – 1.8V to 5.5V

• Package:
  – 14-pin SOIC RoHS compliant IC
  – 20-pin VQFN RoHS compliant IC

• Signal Processing:
  – Self-calibration
  – Auto drift compensation
  – Noise filtering
  – Adjacent Key Suppression® (AKS® – up to three groups possible)
1. Pinouts and Schematics

1.1 Pinout Configuration – Comms Mode (14-pin SOIC)

1.2 Pinout Configuration – Standalone Mode (14-pin SOIC)
1.3 Pinout Configuration – Comms Mode (20-pin VQFN)

1.4 Pinout Configuration – Standalone Mode (20-pin VQFN)
## 1.5 Pin Descriptions

### Table 1-1. Pin Listings (14-pin SOIC)

<table>
<thead>
<tr>
<th>Pin</th>
<th>Name (Comms Mode)</th>
<th>Name (Standalone Mode)</th>
<th>Type</th>
<th>Description</th>
<th>If Unused, Connect To...</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Vdd</td>
<td>Vdd</td>
<td>P</td>
<td>Power</td>
<td>–</td>
</tr>
</tbody>
</table>
| 2   | MODE (Vss)       | MODE (Vdd)             | I    | Mode selection pin  
Comms Mode – connect to Vss  
Standalone Mode – connect to Vdd | – |
| 3   | SDA              | OUT0                   | OD   | Comms Mode – I2C-compatible data line  
Standalone Mode – open drain output for guard channel | Open |
| 4   | RESET            | RESET                  | I    | RESET – has internal pull-up 60 kΩ resistor | Open |
| 5   | CHANGE           | OUT4                   | OD   | CHANGE line for controlling the communications flow  
Comms Mode – connect to CHANGE line  
Standalone Mode – connect to output | Open |
| 6   | SCL              | OUT3                   | OD   | Comms Mode – connect to I2C-compatible clock  
Standalone Mode – connect to output | Open |
| 7   | KEY6             | OUT2                   | O/OD | Comms Mode – connect to Key 6  
Standalone Mode – connect to output | Open |
| 8   | KEY5             | OUT1                   | O/OD | Comms Mode – connect to Key 5  
Standalone Mode – connect to output | Open |
| 9   | KEY4             | KEY4                   | O    | Key 4       | Open |
| 10  | KEY3             | KEY3                   | O    | Key 3       | Open |
| 11  | KEY2             | KEY2                   | O    | Key 2       | Open |
| 12  | KEY1             | KEY1                   | O    | Key 1       | Open |
| 13  | KEY0             | KEY0                   | O    | Key 0       | Open |
| 14  | Vss              | Vss                    | P    | Ground      | –                       |

<table>
<thead>
<tr>
<th>Type Description</th>
<th>I</th>
<th>O</th>
<th>OD</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>I Input only</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O Output only, push-pull</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OD Open drain output</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P Ground or power</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 1-2. Pin Listings (20-pin VQFN)

<table>
<thead>
<tr>
<th>Pin</th>
<th>Name (Comms Mode)</th>
<th>Name (Standalone Mode)</th>
<th>Type</th>
<th>Description</th>
<th>If Unused, Connect To…</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>KEY4</td>
<td>KEY4</td>
<td>O</td>
<td>Key 4</td>
<td>Open</td>
</tr>
<tr>
<td>2</td>
<td>KEY3</td>
<td>KEY3</td>
<td>O</td>
<td>Key 3</td>
<td>Open</td>
</tr>
<tr>
<td>3</td>
<td>KEY2</td>
<td>KEY2</td>
<td>O</td>
<td>Key 2</td>
<td>Open</td>
</tr>
<tr>
<td>4</td>
<td>KEY1</td>
<td>KEY1</td>
<td>O</td>
<td>Key 1</td>
<td>Open</td>
</tr>
<tr>
<td>5</td>
<td>KEY0</td>
<td>KEY0</td>
<td>O</td>
<td>Key 0</td>
<td>Open</td>
</tr>
<tr>
<td>6</td>
<td>N/C</td>
<td>N/C</td>
<td>–</td>
<td>Not connected</td>
<td>–</td>
</tr>
<tr>
<td>7</td>
<td>N/C</td>
<td>N/C</td>
<td>–</td>
<td>Not connected</td>
<td>–</td>
</tr>
<tr>
<td>8</td>
<td>Vss</td>
<td>Vss</td>
<td>P</td>
<td>Ground</td>
<td>–</td>
</tr>
<tr>
<td>9</td>
<td>Vdd</td>
<td>Vdd</td>
<td>P</td>
<td>Power</td>
<td>–</td>
</tr>
<tr>
<td>10</td>
<td>N/C</td>
<td>N/C</td>
<td>–</td>
<td>Not connected</td>
<td>–</td>
</tr>
<tr>
<td>11</td>
<td>MODE (Vss)</td>
<td>MODE (Vdd)</td>
<td>I</td>
<td>Mode selection pin</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Comms Mode – connect to Vss</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Standalone Mode – connect to Vdd</td>
<td>–</td>
</tr>
<tr>
<td>12</td>
<td>SDA</td>
<td>OUT0</td>
<td>OD</td>
<td>Comms Mode – I2C-compatible data line</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Standalone Mode – open drain output for guard channel</td>
<td>Open</td>
</tr>
<tr>
<td>13</td>
<td>RESET</td>
<td>RESET</td>
<td>I</td>
<td>RESET – has internal pull-up 60 kΩ resistor</td>
<td>Open</td>
</tr>
<tr>
<td>14</td>
<td>CHANGE</td>
<td>OUT4</td>
<td>OD</td>
<td>CHANGE line for controlling the communications flow</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Comms Mode – connect to CHANGE line</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Standalone Mode – connects to output</td>
<td>Open</td>
</tr>
<tr>
<td>15</td>
<td>SCL</td>
<td>OUT3</td>
<td>OD</td>
<td>Comms Mode – connect to I2C-compatible clock</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Standalone Mode – connect to output</td>
<td>Open</td>
</tr>
<tr>
<td>16</td>
<td>KEY6</td>
<td>OUT2</td>
<td>O/OD</td>
<td>Comms Mode – connect to Key 6</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Standalone Mode – connect to output</td>
<td>Open</td>
</tr>
<tr>
<td>17</td>
<td>KEY5</td>
<td>OUT1</td>
<td>O/OD</td>
<td>Comms Mode – connect to Key 5</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Standalone Mode – connect to output</td>
<td>Open</td>
</tr>
<tr>
<td>18</td>
<td>N/C</td>
<td>N/C</td>
<td>–</td>
<td>Not connected</td>
<td>–</td>
</tr>
<tr>
<td>19</td>
<td>N/C</td>
<td>N/C</td>
<td>–</td>
<td>Not connected</td>
<td>–</td>
</tr>
<tr>
<td>20</td>
<td>N/C</td>
<td>N/C</td>
<td>–</td>
<td>Not connected</td>
<td>–</td>
</tr>
</tbody>
</table>

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

Figure 1-1. Typical Circuit – Comms (14-pin SOIC)

Figure 1-2. Typical Circuit – Standalone (14-pin SOIC)
Figure 1-3. Typical Circuit – Comms (20-pin VQFN)

1) The central pad on the underside of the chip is a Vss pin and should be connected to ground. Do not put any other tracks underneath the body of the chip.
2) It is important to place all Rs components physically near to the chip.

Figure 1-4. Typical Circuit – Standalone (20-pin VQFN)

1) The central pad on the underside of the chip is a Vss pin and should be connected to ground. Do not put any other tracks underneath the body of the chip.
2) It is important to place all Rs components physically near to the chip.
Re Figure 1-1, 1-2, 1-3 and 1-4, check the following sections for component values:

- **Section 3.1 on page 13**: Series resistors (Rs0 – Rs6 for comms mode and Rs0 – Rs4 for standalone mode)
- **Section 3.2 on page 13**: LED traces
- **Section 3.4 on page 14**: Power Supply (voltage levels)
- **Section 4.4 on page 17**: SDA, SCL pull-up resistors
2. Overview

2.1 Introduction

The AT42QT1070 (QT1070) is a digital burst mode charge-transfer (QT™) capacitive sensor driver. The device can sense from one to seven keys, dependent on mode.

The QT1070 includes all signal processing functions necessary to provide stable sensing under a wide variety of changing conditions, and the outputs are fully debounced. Only a few external parts are required for operation and no external Cs capacitors are required.

The QT1070 modulates its bursts in a spread-spectrum fashion in order to heavily suppress the effects of external noise, and to suppress RF emissions. The QT1070 use a dual-pulse method of acquisition. This provides greater noise immunity and eliminates the need for external sampling capacitors, allowing touch sensing using a single pin.

2.2 Modes

2.2.1 Comms Mode

The QT1070 can operate in comms mode where a host can communicate with the device via an I²C-compatible bus. This allows the user to configure settings for Threshold, Adjacent Key Suppression® (AKS®), Detect Integrator, Low Power (LP) Mode, Guard Channel and Max Time On for keys.

2.2.2 Standalone Mode

The QT1070 can operate in a standalone mode where an I²C-compatible interface is not required. To enter standalone mode, connect the Mode pin to Vdd before powering up the QT1070.

In standalone mode, the start-up values are hard coded in firmware and cannot be changed. The default start-up values are used. This means that key detection is reported via their respective IOs. The Guard channel feature is automatically implemented on key 0 in standalone mode. This means that this channel gets priority over all other keys going into touch.

2.3 Keys

Dependent on mode, the QT1070 can have a minimum of one key and a maximum of seven keys. These can be constructed in different shapes and sizes. See “Features” on page 1 for the recommended dimensions.

- Comms mode – 1 to 7 keys (or 1 to 6 keys plus Guard Channel)
- Standalone mode – 1 to 4 keys plus a Guard Channel

Unused keys should be disabled by setting the averaging factor to zero (see Section 5.9 on page 21).

The status register can be read to determine the touch status of the corresponding key. It is recommended using the open-drain CHANGE line to detect when a change of status has occurred.

2.4 Input/Output (IO) Lines

There are no IO lines in comms mode.

In Standalone mode pins OUT0 – OUT4 can be used as open drain outputs for driving LEDs.
2.5 Acquisition/Low Power Mode (LP)

There are 255 different acquisition times possible. These are controlled via the LP mode byte (see Section 5.11 on page 22) which can be written to via I²C-compatible communication.

LP mode controls the intervals between acquisition measurements. Longer intervals consume lower power but have an increased response time. During calibration, touch and during the detect integrator (DI) period, the LP mode is temporarily set to LP mode 1 for a faster response.

The QT1070 operation is based on a fixed cycle time of approximately 8 ms. The LP mode setting indicates how many of these periods exist per measurement cycle. For example, if LP mode = 1, there is an acquisition every cycle (8 ms). If LP mode = 3, there is an acquisition every 3 cycles (24 ms). If a high Averaging Factor (see Section 5.9 on page 21) setting is selected then the acquisition time may exceed 8 ms.

LP settings above mode 32 (256 ms) result in slower thermal drift compensation and should be avoided in applications where fast thermal transients occur.

2.6 Adjacent Key Suppression (AKS) Technology

The device includes Atmel’s patented Adjacent Key Suppression (AKS) technology, to allow the use of tightly spaced keys on a keypad with no loss of selectability by the user.

There can be up to three AKS groups, implemented so that only one key in the group may be reported as being touched at any one time. Once a key in a particular AKS group is in detect no other key in that group can go into detect. Only when the key in detect goes out of detection can another key go into detect state.

The keys which are members of the AKS groups can be set (see Section 5.9 on page 21). Keys outside the group may be in detect simultaneously.

2.7 CHANGE Line (Comms Mode Only)

The CHANGE line is active low and signals when there is a change in state in the Detection or Input status bytes. It is cleared (allowed to float high) when the host reads the status bytes.

If the status bytes change back to their original state before the host has read the status bytes (for example, a touch followed by a release), the CHANGE line will be held low. In this case, a read to any memory location will clear the CHANGE line.

The CHANGE line is open-drain and should be connected via a 47 kΩ resistor to Vdd. It is necessary for minimum power operation as it ensures that the QT1070 can sleep for as long as possible. Communications wake up the QT1070 from sleep causing a higher power consumption if the part is randomly polled.

Note: The CHANGE line is pulled low 100 ms after power-up or reset.
2.8 Proximity Sensing

The QT1070 is capable of detecting near-proximity or touch. By increasing the sensitivity, the QT1070 can be used as a very effective proximity sensor, allowing the presence of a nearby object to be detected.

As the object being sensed is typically a hand, very large electrode sizes can be used (see Section 2.12.3 on page 12 for Cx limitations), which is extremely effective in increasing the sensitivity of the detector. Note that, although this affects the responsiveness of the sensor, it is less of an issue in proximity sensing applications; in such applications it is only necessary to detect the presence of a large object, rather than a small, precise touch.

Proximity sensing technology enables users to interact with consumer electronics without even touching them. With this powerful technology, sensors in a device such as a PC notebook, PC peripheral, or digital photo frame, sense the presence of a user’s hand and take action. These sensors can illuminate LEDs for discoverable buttons, wake devices from power-saving mode immediately, or activate other functionality.

Refer to “Associated Documents” on page 37 for information about design guidelines.

2.9 Types of Reset

2.9.1 External Reset
An external reset logic line can be used if desired, fed into the RESET pin. However, under most conditions it is acceptable to tie RESET to Vdd.

2.9.2 Soft Reset
The host can cause a device reset by writing a nonzero value to the RESET byte. This soft reset triggers the internal watchdog timer on a 125 ms interval. After 125 ms the device resets and wakes again.

The device NACKs any attempts to communicate with it during the first 30 ms of its initialization period.

2.10 Calibration
Writing a nonzero value to the calibration byte can force a recalibration at any time. This can be useful to clear out a stuck key condition after a prolonged period of uninterrupted detection.

Note: A calibrate command clears all key status bits and the overflow bit (until it is checked on the next cycle).

2.11 Guard Channel
A guard channel to help prevent false detection is available in both modes. This is fixed on key 0 for standalone mode and programmable for comms mode.

Guard channel keys should be more sensitive than the other keys (physically bigger). Because the guard channel key is physically bigger it becomes more susceptible to noise so it has a higher Averaging Factor (see Section 5.9 on page 21) and a lower Threshold (see Section 5.8 on page 21) than the other keys. In standalone mode it should have an Averaging Factor of 16 and a Threshold of 10 counts.
A channel set as the guard channel (there can only be one) is prioritised when the filtering of keys going into detect is taking place. So if a normal key is filtering into touch (touch present but DI has not been reached) and the key set as the guard key begins filtering in, then the normal key’s filter is reset and the guard key filters in first.

The guard channel is connected to a sensor pad which detects the presence of touch and overrides any output from the other keys.

**Figure 2-1.** Guard Channel Example

2.12 Signal Processing

2.12.1 Detect Threshold

The device detects a touch when the signal has crossed a threshold level and remained there for a specified number of counts (see Section 5.10 on page 21). This can be altered on a key-by-key basis using the key threshold I²C-compatible commands.

In standalone mode the detect threshold is set to a fixed value of 10 counts of change with respect to the internal reference level for the guard channel and 20 counts for the other four keys. The reference level has the ability to adjust itself slowly in accordance with the drift compensation mechanism.

The drift mechanism will drift toward touch at a rate of 160 ms x 18 = 2.88 seconds and away from touch at a rate of 160 ms x 6 = 0.96 seconds. The 160 ms is based on 20 x 8 ms cycles. If the cycle time exceeds 8 ms then the overall times will be extended to match.

2.12.2 Detect Integrator

The device features a fast detection integrator counter (DI filter), which acts to filter out noise at the small expense of a slower response time. The DI filter requires a programmable number of consecutive samples confirmed in detection before the key is declared to be touched. The minimum number for the DI filter is 2. Settings of 0 and 1 for the DI also default to 2. The DI is also implemented when a touch is removed.

2.12.3 Cx Limitations

The recommended range for key capacitance Cx is 1 pF – 30 pF. Larger values of Cx will give reduced sensitivity.
2.12.4 Max On Duration

If an object or material obstructs the sense pad the signal may rise enough to create a detection, preventing further operation. To prevent this, the sensor includes a timer which monitors detections. If a detection exceeds the timer setting the sensor performs a key recalibration. This is known as the Max On duration feature and is set to approximately 30s in standalone mode.

In comms mode this feature can be changed by setting a value in the range 1 – 255 (160 ms – 40800 ms) in steps of 160 ms. A setting of 0 disables the Max On Duration recalibration feature.

**Note:** If bit 4 of address 53 is clear then a recalibration of all keys occurs on Max On Duration instead of the individual key recalibration.

2.12.5 Positive Recalibration

If a keys signal jumps in the negative direction (with respect to its reference) by more than the Positive Recalibration setting (4 counts), then a recalibration of that key takes place.

2.12.6 Drift Hold Time

Drift Hold Time (DHT) is used to restrict drift on all keys while one or more keys are activated. DHT restricts the drifting on all keys until approximately four seconds after all touches have been removed.

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 touch detection.

2.12.7 Hysteresis

Hysteresis is fixed at 12.5 percent of the Detect Threshold. When a key enters a detect state once the DI count has been reached, the NTHR value is changed by a small amount (12.5 percent of NTHR) in the direction away from touch. This is done to affect hysteresis and so makes it less likely a key will dither in and out of detect. NTHR is restored once the key drops out of detect.

3. Wiring and Parts

3.1 Rs Resistors

Series resistors Rs (Rs0 – Rs6 for comms mode and Rs0 – Rs4 for standalone mode) are in line with the electrode connections and should be used to limit electrostatic discharge (ESD) currents and to suppress radio frequency interference (RFI). Series resistors are recommended for noise reduction. They should be approximately 4.7 kΩ to 20 kΩ each.

3.2 LED Traces and Other Switching Signals

Digital switching signals near the sense lines induce transients into the acquired signals, deteriorating the signal-to-noise (SNR) performance of the device. Such signals should be routed away from the sensing traces and electrodes, or the design should be such that these lines are not switched during the course of signal acquisition (bursts).
LED terminals which are multiplexed or switched into a floating state, and which are within, or physically very near, a key (even if on another nearby PCB) should be bypassed to either Vss or Vdd with at least a 10 nF capacitor. This is to suppress capacitive coupling effects which can induce false signal shifts. The bypass capacitor does not need to be next to the LED, in fact it can be quite distant. The bypass capacitor is noncritical and can be of any type.

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

3.3 PCB Cleanliness

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

**CAUTION:** If a PCB is reworked in any way, it is 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.

3.4 Power Supply

See Section 6.2 on page 24 for the power supply range. If the power supply fluctuates slowly with temperature, the device tracks and compensates for these changes automatically with only minor changes in sensitivity. If the supply voltage drifts or shifts quickly, the drift compensation mechanism is not able to keep up, causing sensitivity anomalies or false detections.

The usual power supply considerations with QT parts apply to the device. The power should be clean and come from a separate regulator if possible. However, this device is designed to minimize the effects of unstable power, and except in extreme conditions should not require a separate Low Dropout (LDO) regulator.

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

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

It is assumed that a larger bypass capacitor (like 1 µF) is somewhere else in the power circuit; for example, near the regulator.

To assist with transient regulator stability problems, the QT1070 waits 500 µs any time it wakes up from a sleep state (in LP modes) before acquiring, to allow Vdd to fully stabilize.
4. I^2^C-compatible Communications (Comms Mode Only)

4.1 I^2^C-compatible Protocol

4.1.1 Protocol

The I^2^C-compatible protocol is based around access to an address table (see Table 5-1 on page 18) and supports multibyte reads and writes. The maximum clock rate is 400 kHz. See Section A on page 31 for an overview of I^2^C-compatible bus operation.

4.1.2 Signals

The I^2^C-compatible interface requires two signals to operate:

- **SDA** - Serial Data
- **SCL** - Serial Clock

A third line, **CHANGE**, is used to signal when the device has seen a change in the status byte:

- **CHANGE**: Open-drain, active low when any capacitive key has changed state since the last I^2^C-compatible read. After reading the two status bytes, this pin floats (high) again if it is pulled up with an external resistor. If the status bytes change back to their original state before the host has read the status bytes (for example, a touch followed by a release), the **CHANGE** line is held low. In this case, a read to any memory location clears the **CHANGE** line.

4.2 I^2^C-compatible Address

There is one preset I^2^C-compatible address of 0x1B. This is not changeable.

4.3 Data Read/Write

4.3.1 Writing Data to the Device

The sequence of events required to write data to the device is shown next.

1. The host initiates the transfer by sending the START condition
2. The host follows this by sending the slave address of the device together with the WRITE bit.

Table 4-1. Description of Write Data Bits

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>START condition</td>
</tr>
<tr>
<td>SLA+W</td>
<td>Slave address plus write bit</td>
</tr>
<tr>
<td>A</td>
<td>Acknowledge bit</td>
</tr>
<tr>
<td>MemAddress</td>
<td>Target memory address within device</td>
</tr>
<tr>
<td>Data</td>
<td>Data to be written</td>
</tr>
<tr>
<td>P</td>
<td>Stop condition</td>
</tr>
</tbody>
</table>

1. The host initiates the transfer by sending the START condition
2. The host follows this by sending the slave address of the device together with the WRITE bit.
3. The device sends an ACK.
4. The host then sends the memory address within the device it wishes to write to.
5. The device sends an ACK if the write address is in the range 0x00 – 0x7F, otherwise it sends a NACK.
6. The host transmits one or more data bytes; each is acknowledged by the device (unless trying to write to an invalid address).
7. If the host sends more than one data byte, they are written to consecutive memory addresses.
8. The device automatically increments the target memory address after writing each data byte.
9. After writing the last data byte, the host should send the STOP condition.

Note: The host should not try to write to addresses outside the range 0x20 to 0x39 because this is the limit of the device’s internal memory address.

4.3.2 Reading Data From the Device

The sequence of events required to read data from the device is shown next.

1. The host initiates the transfer by sending the START condition
2. The host follows this by sending the slave address of the device together with the WRITE bit.
3. The device sends an ACK.
4. The host then sends the memory address within the device it wishes to read from.
5. The device sends an ACK if the address to be read from is less than 0x80 otherwise it sends a NACK.
6. The host must then send a STOP and a START condition followed by the slave address again but this time accompanied by the READ bit.
   Note: Alternatively, instead of step 6 a repeated START can be sent so the host does not need to relinquish control of the bus.
7. The device returns an ACK, followed by a data byte.
8. The host must return either an ACK or NACK.
   a. If the host returns an ACK, the device subsequently transmits the data byte from the next address. Each time a data byte is transmitted, the device automatically increments the internal address. The device continues to return data bytes until the host responds with a NACK.
   b. If the host returns a NACK, it should then terminate the transfer by issuing the STOP condition.
9. The device resets the internal address to the location indicated by the memory address sent to it previously. Therefore, there is no need to send the memory address again when reading from the same location.

Note: Reading the 16-bit reference and signal values is not an automatic operation; reading the first byte of a 16-bit value does not lock the other byte. As a result glitches in the reported value may be seen as values increase from 255 to 256, or decrease from 256 to 255.
4.4 SDA, SCL

The I²C-compatible bus transmits data and clock with SDA and SCL respectively. They are open-drain; that is I²C-compatible master and slave devices can only drive these lines low or leave them open. The termination resistors pull the line up to Vdd if no I²C-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 I²C-compatible specifications (1 µs maximum).

Standalone mode: if I²C-compatible communications are not required, then standalone mode can be enabled by connecting the MODE pin to Vdd. See Section 2.4 on page 9 for more information.
5. Setups

5.1 Introduction

The device calibrates and processes signals using a number of algorithms specifically designed to provide for high survivability in the face of adverse environmental challenges. User-defined Setups are employed to alter these algorithms to suit each application. These Setups are loaded into the device over the I²C-compatible serial interfaces. In standalone these settings are fixed to predetermined values.

Table 5-1. Internal Register Address Allocation

<table>
<thead>
<tr>
<th>Address</th>
<th>Use</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>R/W</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Chip ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>R</td>
</tr>
<tr>
<td>1</td>
<td>Firmware Version</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>R</td>
</tr>
<tr>
<td>2</td>
<td>Detection status</td>
<td>CALIBRATE OVERFLOW</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>TOUCH</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Key status</td>
<td>Reserved</td>
<td>Key 6</td>
<td>Key 5</td>
<td>Key 4</td>
<td>Key 3</td>
<td>Key 2</td>
<td>Key 1</td>
<td>Key 0</td>
<td>R</td>
</tr>
<tr>
<td>4 – 5</td>
<td>Key signal 0</td>
<td></td>
<td></td>
<td></td>
<td>Key signal 0 (MSByte) – Key signal 0 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 – 7</td>
<td>Key signal 1</td>
<td></td>
<td></td>
<td></td>
<td>Key signal 1 (MSByte) – Key signal 1 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8 – 9</td>
<td>Key signal 2</td>
<td></td>
<td></td>
<td></td>
<td>Key signal 2 (MSByte) – Key signal 2 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 – 11</td>
<td>Key signal 3</td>
<td></td>
<td></td>
<td></td>
<td>Key signal 3 (MSByte) – Key signal 3 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12 – 13</td>
<td>Key signal 4</td>
<td></td>
<td></td>
<td></td>
<td>Key signal 4 (MSByte) – Key signal 4 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14 – 15</td>
<td>Key signal 5</td>
<td></td>
<td></td>
<td></td>
<td>Key signal 5 (MSByte) – Key signal 5 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16 – 17</td>
<td>Key signal 6</td>
<td></td>
<td></td>
<td></td>
<td>Key signal 6 (MSByte) – Key signal 6 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18 – 19</td>
<td>Reference data 0</td>
<td></td>
<td></td>
<td></td>
<td>Reference data 0 (MSByte) – Reference data 0 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20 – 21</td>
<td>Reference data 1</td>
<td></td>
<td></td>
<td></td>
<td>Reference data 1 (MSByte) – Reference data 1 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22 – 23</td>
<td>Reference data 2</td>
<td></td>
<td></td>
<td></td>
<td>Reference data 2 (MSByte) – Reference data 2 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24 – 25</td>
<td>Reference data 3</td>
<td></td>
<td></td>
<td></td>
<td>Reference data 3 (MSByte) – Reference data 3 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 – 27</td>
<td>Reference data 4</td>
<td></td>
<td></td>
<td></td>
<td>Reference data 4 (MSByte) – Reference data 4 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30 – 31</td>
<td>Reference data 6</td>
<td></td>
<td></td>
<td></td>
<td>Reference data 6 (MSByte) – Reference data 6 (LSByte)</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>NTHR key 0</td>
<td></td>
<td></td>
<td></td>
<td>Negative Threshold level for key 0</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>NTHR key 1</td>
<td></td>
<td></td>
<td></td>
<td>Negative Threshold level for key 1</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>NTHR key 2</td>
<td></td>
<td></td>
<td></td>
<td>Negative Threshold level for key 2</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>NTHR key 3</td>
<td></td>
<td></td>
<td></td>
<td>Negative Threshold level for key 3</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>NTHR key 4</td>
<td></td>
<td></td>
<td></td>
<td>Negative Threshold level for key 4</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>NTHR key 5</td>
<td></td>
<td></td>
<td></td>
<td>Negative Threshold level for key 5</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>NTHR key 6</td>
<td></td>
<td></td>
<td></td>
<td>Negative Threshold level for key 6</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>AVE/AKS key 0</td>
<td></td>
<td></td>
<td></td>
<td>Adjacent key suppression level for key 0</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>AVE/AKS key 1</td>
<td></td>
<td></td>
<td></td>
<td>Adjacent key suppression level for key 1</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>AVE/AKS key 2</td>
<td></td>
<td></td>
<td></td>
<td>Adjacent key suppression level for key 2</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>AVE/AKS key 3</td>
<td></td>
<td></td>
<td></td>
<td>Adjacent key suppression level for key 3</td>
<td>R/W</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
5.2 Address 0: Chip ID

Table 5-2. Chip ID

<table>
<thead>
<tr>
<th>Address</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>MAJOR ID</td>
</tr>
</tbody>
</table>

MAJOR ID: Reads back as 2
MINOR ID: Reads back as E

5.3 Address 1: Firmware Version

Table 5-3. Firmware Version

<table>
<thead>
<tr>
<th>Address</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FIRMWARE VERSION: this shows the 8-bit firmware version 1.5 (0x15).
5.4 Address 2: Detection Status

Table 5-4. Detection Status

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>2</td>
<td>CALIBRATE</td>
<td>OVERFLOW</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>TOUCH</td>
</tr>
</tbody>
</table>

**CALIBRATE:** This bit is set during a calibration sequence.

**OVERFLOW:** This bit is set if the time to acquire all key signals exceeds 8 ms.

**TOUCH:** This bit is set if any keys are in detect.

5.5 Address 3: Key Status

Table 5-5. Key Status

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Reserved</td>
<td>KEY6</td>
<td>KEY5</td>
<td>KEY4</td>
<td>KEY3</td>
<td>KEY2</td>
<td>KEY1</td>
<td>KEY0</td>
</tr>
</tbody>
</table>

**KEY0 – 6:** bits 0 to 6 indicate which keys are in detection, if any. Touched keys report as 1, untouched or disabled keys report as 0.

5.6 Address 4 – 17: Key Signal

Table 5-6. Key Signal

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>4</td>
<td></td>
<td>MSByte OF KEY SIGNAL FOR KEY 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>LSByte OF KEY SIGNAL FOR KEY 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 – 17</td>
<td></td>
<td>MSByte/LSByte OF KEY SIGNAL FOR KEYS 1 – 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**KEY SIGNAL:** addresses 4 – 17 allow key signals to be read for each key, starting with key 0. There are two bytes of data for each key. These are the key’s 16-bit key signals which are accessed as two 8-bit bytes, stored MSByte first. These addresses are read-only.

5.7 Address 18 – 31: Reference Data

Table 5-7. Reference Data

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20 – 31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**REFERENCE DATA:** addresses 18 – 31 allow reference data to be read for each key, starting with key 0. There are two bytes of data for each key. These are the key’s 16-bit reference data which is accessed as two 8-bit bytes, stored MSByte first. These addresses are read-only.
5.8 Address 32 – 38: Negative Threshold (NTHR)

Table 5-8. NTHR

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>32 – 38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

NTHR Keys 0 – 6: these 8-bit values set the threshold value for each key to register a detection.

Default: 20 counts

Note: Do not use a setting of 0 as this causes a key to go into detection when its signal is equal to its reference.

5.9 Address 39 – 45: Averaging Factor/Adjacent Key Suppression (AVE/AKS)

Table 5-9. AVE/AKS

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>39 – 45</td>
<td>AVE5</td>
<td>AVE4</td>
<td>AVE3</td>
<td>AVE2</td>
<td>AVE1</td>
<td>AVE0</td>
<td>AKS1</td>
<td>AKS0</td>
</tr>
</tbody>
</table>

AVE 0 – 5: The Averaging Factor (AVE) is the number of pulses which are added together and averaged to get the final signal value for that channel.

For example, if AVE = 8 then 8 ADC samples are taken and added together. The result is divided by the original number of pulses (8). If sixteen pulses are used then the result is divided by sixteen.

This provides a better signal-to-noise ratio but requires longer acquire times. Values for AVE are restricted internally to 1, 2, 4, 8, 16 or 32.

Default: 8 (In standalone mode key 0 is 16)

AKS 0 – 1: these bits control which keys are included in an AKS group. There can be up to three groups, each containing any number of keys (up to the maximum allowed for the mode).

Each key can have a value between 0 and 3, which assigns it to an AKS group of that number. A key may only go into detect when it has the largest signal change of any key in its group. A value of 0 means the key is not in any AKS group.

Default: 0x01

5.10 Address 46 – 52: Detection Integrator (DI)

Table 5-10. Detection Integrator

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>46 – 52</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DETECTION INTEGRATOR: addresses 46 – 52 allow the DI level to be set for each key. This 8-bit value controls the number of consecutive measurements that must be confirmed as having passed the key threshold before that key is registered as being in detect. The minimum value for the DI filter is 2. Settings of 0 and 1 for the DI also default to 2 because a minimum of two consecutive measurements must be confirmed.

Default: 4
5.11 Address 53: FastOutDI/Max Cal/Guard Channel

Table 5-11. Max On/Guard Channel

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>53</td>
<td>–</td>
<td>FO</td>
<td>MAX CAL</td>
<td></td>
<td>GUARD CHANNEL</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FO: Fast Out DI – when bit 5 is set then a key filters out with an integrator of 4. Could have a DI in of 100 but filter out with DI of 4 (global setting for all keys).

MAX CAL: if this bit is clear then all keys recalibrate after a Max On Duration timeout, otherwise only the key with the incorrect timing gets recalibrated.

GUARD CHANNEL: bits 0 – 3 are used to set a key as the guard channel (which gets priority filtering). Valid values are 0 – 6, with any larger value disabling the guard key feature.

5.12 Address 54: Low Power (LP) Mode

Table 5-12. LP Mode

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>54</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LOW POWER MODE</td>
</tr>
</tbody>
</table>

LP MODE: this 8-bit value determines the number of 8 ms intervals between key measurements. Longer intervals between measurements yield a lower power consumption but at the expense of a slower response to touch.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>8 ms</td>
</tr>
<tr>
<td>1</td>
<td>8 ms</td>
</tr>
<tr>
<td>2</td>
<td>16 ms</td>
</tr>
<tr>
<td>3</td>
<td>24 ms</td>
</tr>
<tr>
<td>4</td>
<td>32 ms</td>
</tr>
<tr>
<td>...254</td>
<td>2.032s</td>
</tr>
<tr>
<td>255</td>
<td>2.040s</td>
</tr>
</tbody>
</table>

Default: 2 (16 ms between key acquisitions)
5.13 **Address 55: Max On Duration**

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MAX ON DURATION</td>
</tr>
</tbody>
</table>

**MAX ON DURATION:** this is a 8-bit value which determines how long any key can be in touch before it recalibrates itself.

A value of 0 turns Max On Duration off.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Off</td>
</tr>
<tr>
<td>1</td>
<td>160 ms</td>
</tr>
<tr>
<td>2</td>
<td>320 ms</td>
</tr>
<tr>
<td>3</td>
<td>480 ms</td>
</tr>
<tr>
<td>4</td>
<td>640 ms</td>
</tr>
<tr>
<td>255</td>
<td>40.8s</td>
</tr>
</tbody>
</table>

Default: 180 (160 ms x 180 = 28.8s)

5.14 **Address 56: Calibrate**

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>56</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Writing a nonzero value forces a calibration</td>
</tr>
</tbody>
</table>

Writing any nonzero value into this address triggers the device to start a calibration cycle. The CALIBRATE flag in the detection status register is set when the calibration begins and clears when the calibration has finished.

5.15 **Address 57: RESET**

<table>
<thead>
<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>
</thead>
<tbody>
<tr>
<td>57</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Writing a nonzero value forces a reset</td>
</tr>
</tbody>
</table>

Writing any nonzero value to this address triggers the device to reset.
6. Specifications

6.1 Absolute Maximum Specifications

<table>
<thead>
<tr>
<th>Vdd</th>
<th>-0.5 to +6V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max continuous pin current, any control or drive pin</td>
<td>±10 mA</td>
</tr>
<tr>
<td>Short circuit duration to ground, any pin</td>
<td>infinite</td>
</tr>
<tr>
<td>Short circuit duration to Vdd, any pin</td>
<td>infinite</td>
</tr>
<tr>
<td>Voltage forced onto any pin</td>
<td>-0.5V to (Vdd + 0.5) Volts</td>
</tr>
</tbody>
</table>

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.

6.2 Recommended Operating Conditions

<table>
<thead>
<tr>
<th>Operating temperature</th>
<th>-40°C to +85°C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage temperature</td>
<td>-55°C to +125°C</td>
</tr>
<tr>
<td>Vdd</td>
<td>+1.8V to 5.5V</td>
</tr>
<tr>
<td>Supply ripple+noise</td>
<td>±25 mV</td>
</tr>
<tr>
<td>Cx load capacitance per key</td>
<td>1 to 30 pF</td>
</tr>
</tbody>
</table>

6.3 DC Specifications

Vdd = 3.3V, Cs = 10nF, load = 5 pF, 32 ms default sleep, Ta = recommended range, unless otherwise noted

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Minimum</th>
<th>Typical</th>
<th>Maximum</th>
<th>Units</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vil</td>
<td>Low input logic level</td>
<td>–</td>
<td>–</td>
<td>0.2Vdd</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>Vih</td>
<td>High input logic level</td>
<td>0.7Vdd</td>
<td>–</td>
<td>Vdd + 0.5</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>Vol</td>
<td>Low output voltage</td>
<td>–</td>
<td>–</td>
<td>0.6</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>Voh</td>
<td>High output voltage</td>
<td>Vdd - 0.7V</td>
<td>–</td>
<td>–</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>Iil</td>
<td>Input leakage current</td>
<td>–</td>
<td>–</td>
<td>±1</td>
<td>µA</td>
<td></td>
</tr>
</tbody>
</table>
6.4 Power Consumption Measurements

$C_x = 5 \text{ pF, } R_s = 4.7 \text{ k}\Omega$

<table>
<thead>
<tr>
<th>LP Mode</th>
<th>$I_{dd} (\mu A) \text{ at } V_{dd}$ =</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5V</td>
</tr>
<tr>
<td>0 (8 ms)</td>
<td>1744</td>
</tr>
<tr>
<td>1 (16 ms)</td>
<td>1375</td>
</tr>
<tr>
<td>2 (24 ms)</td>
<td>1263</td>
</tr>
<tr>
<td>4 (32 ms)</td>
<td>1168</td>
</tr>
<tr>
<td>5 (40 ms)</td>
<td>1119</td>
</tr>
<tr>
<td>6 (48 ms)</td>
<td>1089</td>
</tr>
</tbody>
</table>

6.5 Timing Specifications

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Minimum</th>
<th>Typical</th>
<th>Maximum</th>
<th>Units</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_R$</td>
<td>Response time</td>
<td>DI setting x 8 ms</td>
<td>–</td>
<td>LP mode + (DI setting x 8 ms)</td>
<td>ms</td>
<td>Under host control</td>
</tr>
<tr>
<td>$F_{QF}$</td>
<td>Sample frequency</td>
<td>162</td>
<td>180</td>
<td>198</td>
<td>kHz</td>
<td>Modulated spread-spectrum (chirp)</td>
</tr>
<tr>
<td>$T_D$</td>
<td>Power-up delay to operate/calibration time</td>
<td>–</td>
<td>&lt;230</td>
<td>–</td>
<td>ms</td>
<td>Can be longer if burst is very long.</td>
</tr>
<tr>
<td>$F_{I2C}$</td>
<td>I$^2$C-compatible clock rate</td>
<td>–</td>
<td>–</td>
<td>400</td>
<td>kHz</td>
<td>–</td>
</tr>
<tr>
<td>$F_m$</td>
<td>Burst modulation, percentage</td>
<td>±8</td>
<td>–</td>
<td>–</td>
<td>%</td>
<td>–</td>
</tr>
<tr>
<td>RESET</td>
<td>pulse width</td>
<td>5</td>
<td>–</td>
<td>–</td>
<td>µs</td>
<td>–</td>
</tr>
</tbody>
</table>
6.6 Mechanical Dimensions

6.7 AT42QT1070X-SSU – 14-pin SOIC

---

**Notes:**

1. This drawing is for general information only; refer to JEDEC Drawing MS-012, Variation AB for additional information.
2. Dimension D does not include mold Flash, protrusions or gate burrs. Mold Flash, protrusion and gate burrs shall not exceed 0.15 mm (0.006") per side.
3. Dimension E does not include inter-lead Flash or protrusion. Inter-lead flash and protrusions shall not exceed 0.25 mm (0.010") per side.
4. L is the length of the terminal for soldering to a substrate.
5. The lead width B, as measured 0.36 mm (0.014") or greater above the seating plane, shall not exceed a maximum value of 0.61 mm (0.024") per side.

---

**COMMON DIMENSIONS**

(Unit of Measure = mm/inches)

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>MIN</th>
<th>NOM</th>
<th>MAX</th>
<th>NOTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1.35/0.0532</td>
<td>–</td>
<td>1.75/0.0688</td>
<td></td>
</tr>
<tr>
<td>A1</td>
<td>0.1/0.0040</td>
<td>–</td>
<td>0.25/0.0098</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>0.33/0.0130</td>
<td>–</td>
<td>0.5/0.02005</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>8.55/0.3367</td>
<td>–</td>
<td>8.74/0.3444</td>
<td>2</td>
</tr>
<tr>
<td>E</td>
<td>3.8/0.1497</td>
<td>–</td>
<td>3.99/0.1574</td>
<td>3</td>
</tr>
<tr>
<td>H</td>
<td>5.8/0.2284</td>
<td>–</td>
<td>6.19/0.2440</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>0.41/0.0160</td>
<td>–</td>
<td>1.27/0.0500</td>
<td>4</td>
</tr>
<tr>
<td>e</td>
<td>1.27/0.050 BSC</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

2325 Orchard Parkway
San Jose, CA 95131

**TITLE**
14S1, 14-lead, 0.150” Wide Body, Plastic Gull Wing Small Outline Package (SOIC)

**DRAWING NO.**
14S1

**REV.**
A

2/5/02
6.8 AT42QT1070X-MMH – 20-pin 3 x 3 mm VQFN

COMMON DIMENSIONS
(Unit of Measure = mm)

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>MIN</th>
<th>NOM</th>
<th>MAX</th>
<th>NOTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.75</td>
<td>0.80</td>
<td>0.85</td>
<td></td>
</tr>
<tr>
<td>A1</td>
<td>0.00</td>
<td>0.02</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>0.17</td>
<td>0.22</td>
<td>0.27</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td>0.152</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>2.90</td>
<td>3.00</td>
<td>3.10</td>
<td></td>
</tr>
<tr>
<td>D2</td>
<td>1.40</td>
<td>1.55</td>
<td>1.70</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>2.90</td>
<td>3.00</td>
<td>3.10</td>
<td></td>
</tr>
<tr>
<td>E2</td>
<td>1.40</td>
<td>1.55</td>
<td>1.70</td>
<td></td>
</tr>
<tr>
<td>e</td>
<td>–</td>
<td>0.45</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>0.35</td>
<td>0.40</td>
<td>0.45</td>
<td></td>
</tr>
<tr>
<td>K</td>
<td>0.20</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>y</td>
<td>0.00</td>
<td>–</td>
<td>0.08</td>
<td></td>
</tr>
</tbody>
</table>

10/24/08
6.9 Marking

6.9.1 AT42QT1070-SSU – 14-pin SOIC

Either part marking can be used.

- **Pin 1 ID**
- **Date Code**
- **Code revision 1.5, released**
- **Abbreviated part number**

**W** = Week code

- Week code number 1-52 where:
  - A=1 B=2 ..., Z=26
  - then using the underscore A=27..., Z=52

**YYWW** = Date code, variable text
6.9.2 AT42QT1070-MMH – 20-pin 3 x 3 mm VQFN

Either part marking can be used.

- **Pin 1 ID**
- **Code Revision 1.5, released**
- **Date Code, released**

**Shortened part number in hexadecimal**

42E = 1070

**Date Code Description**

W=Week code

W week code number 1-52 where:

A=1 B=2 .... Z=26

then using the underscore A=27...Z=52

**Pin 1 ID**

15 = Code Revision 1.5, released

**Abbreviation of part number:**

(AT42QT1070-MMH)

**X = Assembly location code (variable text)**

**Date Code, released**

**YZZ = traceability code (variable text)**

Y = the last digit of the year
(for example 0 for year 2010, 1 for year 2011),

ZZ is the trace code for each assembly lot.
6.10 Part Number

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT42QT1070-SSU</td>
<td>14-pin SOIC RoHS compliant IC</td>
</tr>
<tr>
<td>AT42QT1070-MMH</td>
<td>20-pin 3 x 3 mm VQFN RoHS compliant IC</td>
</tr>
</tbody>
</table>

6.11 Moisture Sensitivity Level (MSL)

<table>
<thead>
<tr>
<th>MSL Rating</th>
<th>Peak Body Temperature</th>
<th>Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSL3</td>
<td>260°C</td>
<td>IPC/JEDEC J-STD-020</td>
</tr>
</tbody>
</table>
Appendix A. \(^2\text{C}\)-compatible Operation

A.1 Interface Bus

The device communicates with the host over an \(^2\text{C}\)-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 \(^2\text{C}\)-compatible bus as shown in Figure A-1. Both bus lines are connected to Vdd via pull-up resistors. The bus drivers of all \(^2\text{C}\)-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 A-1.** \(^2\text{C}\)-compatible Interface Bus

---

A.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 A-2.** Data Transfer

---

A.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 A-3, START and STOP conditions are signaled by changing the level of the SDA line when the SCL line is high.


A.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 will acknowledge 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.

A.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.
A.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. When reading, a repeated START is allowed. So S, SLA+W, A, S, SLA+R, A, DATA1, A,.....,DATAx, /A, P can be sent.

Figure 6-1 shows a typical data transmission. Note that several data bytes can be transmitted between the SLA+R/W and the STOP.
# Contents

**Features** ......................................................................................................... 1

1 **Pinouts and Schematics** ............................................................................. 2

   1.1 Pinout Configuration – Comms Mode (14-pin SOIC) .................................. 2
   1.2 Pinout Configuration – Standalone Mode (14-pin SOIC) ........................... 2
   1.3 Pinout Configuration – Comms Mode (20-pin VQFN) ............................... 3
   1.4 Pinout Configuration – Standalone Mode (20-pin VQFN) ........................... 3
   1.5 Pin Descriptions ....................................................................................... 4
   1.6 Schematics ............................................................................................... 6

2 **Overview** ..................................................................................................... 9

   2.1 Introduction .............................................................................................. 9
   2.2 Modes ....................................................................................................... 9
      2.2.1 Comms Mode ................................................................................ 9
      2.2.2 Standalone Mode .......................................................................... 9
   2.3 Keys ......................................................................................................... 9
   2.4 Input/Output (IO) Lines .......................................................................... 9
   2.5 Acquisition/Low Power Mode (LP) ........................................................... 10
   2.6 Adjacent Key Suppression (AKS) Technology .......................................... 10
   2.7 CHANGE Line (Comms Mode Only) .......................................................... 10
   2.8 Proximity Sensing .................................................................................... 11
   2.9 Types of Reset .......................................................................................... 11
      2.9.1 External Reset ............................................................................... 11
      2.9.2 Soft Reset ....................................................................................... 11
   2.10 Calibration .............................................................................................. 11
   2.11 Guard Channel ......................................................................................... 11
   2.12 Signal Processing .................................................................................... 12
      2.12.1 Detect Threshold ........................................................................ 12
      2.12.2 Detect Integrator ........................................................................... 12
      2.12.3 Cx Limitations .............................................................................. 12
      2.12.4 Max On Duration .......................................................................... 13
      2.12.5 Positive Recalibration ................................................................... 13
      2.12.6 Drift Hold Time .............................................................................. 13
      2.12.7 Hysteresis ...................................................................................... 13
3  *Wiring and Parts* ................................................................. 13
  3.1 Rs Resistors ........................................................................... 13
  3.2 LED Traces and Other Switching Signals .............................. 13
  3.3 PCB Cleanliness ................................................................. 14
  3.4 Power Supply ....................................................................... 14

4  *I2C-compatible Communications (Comms Mode Only)* .......... 15
  4.1 I2C-compatible Protocol ...................................................... 15
     4.1.1 Protocol ........................................................................ 15
     4.1.2 Signals .......................................................................... 15
  4.2 I2C-compatible Address ...................................................... 15
  4.3 Data Read/Write ................................................................. 15
     4.3.1 Writing Data to the Device ........................................... 15
     4.3.2 Reading Data From the Device ...................................... 16
  4.4 SDA, SCL ............................................................................ 17

5  *Setups* .................................................................................. 18
  5.1 Introduction .......................................................................... 18
  5.2 Address 0: Chip ID ............................................................. 19
  5.3 Address 1: Firmware Version ............................................. 19
  5.4 Address 2: Detection Status .............................................. 20
  5.5 Address 3: Key Status ....................................................... 20
  5.6 Address 4 – 17: Key Signal ............................................... 20
  5.7 Address 18 – 31: Reference Data ..................................... 20
  5.8 Address 32 – 38: Negative Threshold (NTHR) .................... 21
  5.9 Address 39 – 45: Averaging Factor/Adjacent Key Suppression (AVE/AKS) .... 21
  5.10 Address 46 – 52: Detection Integrator (DI) ....................... 21
  5.11 Address 53: FastOutDI/Max Cal/Guard Channel ............... 22
  5.12 Address 54: Low Power (LP) Mode .................................. 22
  5.13 Address 55: Max On Duration ......................................... 23
  5.14 Address 56: Calibrate ..................................................... 23
  5.15 Address 57: RESET ......................................................... 23

6  *Specifications* ...................................................................... 24
  6.1 Absolute Maximum Specifications .................................. 24
  6.2 Recommended Operating Conditions .............................. 24
  6.3 DC Specifications .......................................................... 24
  6.4 Power Consumption Measurements ............................... 25
6.5 Timing Specifications ........................................................................................................25
6.6 Mechanical Dimensions .................................................................................................26
6.7 AT42QT1070X-SSU – 14-pin SOIC ..............................................................................26
6.8 AT42QT1070X-MMH – 20-pin 3 x 3 mm VQFN ............................................................27
6.9 Marking ..........................................................................................................................28
  6.9.1 AT42QT1070-SSU – 14-pin SOIC ........................................................................ 28
  6.9.2 AT42QT1070-MMH – 20-pin 3 x 3 mm VQFN ................................................. 29
6.10 Part Number ...............................................................................................................30
6.11 Moisture Sensitivity Level (MSL) ..............................................................................30

Appendix A I2C-compatible Operation .............................................................................31
  A.1 Interface Bus ............................................................................................................. 31
  A.2 Transferring Data Bits .......................................................................................... 31
  A.3 START and STOP Conditions .............................................................................. 31
  A.4 Address Byte Format ............................................................................................ 32
  A.5 Data Byte Format .................................................................................................. 32
  A.6 Combining Address and Data Bytes into a Transmission .................................. 33

Associated Documents .................................................................................................. 37

Revision History ............................................................................................................. 37
Associated Documents

- QTAN0062 – QTtouch and QMatrix Sensitivity Tuning for Keys, Slider and Wheels
- Touch Sensors Design Guide

Revision History

<table>
<thead>
<tr>
<th>Revision Number</th>
<th>History</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revision A – October 2010</td>
<td>• Initial release of document for code revision 1.5</td>
</tr>
</tbody>
</table>
## 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, 19F  
BEA Tower, Millennium City 5  
418 Kwun Tong Road  
Kwun Tong  
Kowloon  
HONG KONG  
Tel: (+852) 2245-6100  
Fax: (+852) 2722-1369

**Atmel Munich GmbH**  
Business Campus  
Parkring 4  
D- 85748 Garching b. MUNICH  
Tel: (+49) 89-31970-111  
Fax: (+49) 89-3194621

**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

1560 Parkway  
Solent Business Park  
Whiteley  
Fareham  
Hampshire  
PO15 7AG  
UNITED KINGDOM  
Tel: (+44) 844 894 1920  
Fax: (+44) 1489 557 066

---

## Product Contact

**Web Site**  
www.atmel.com

**Technical Support**  
touch@atmel.com

**Sales Contact**  
www.atmel.com/contacts

**Literature Requests**  
www.atmel.com/literature

---

### Disclaimer:

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

© 2010 Atmel Corporation. All rights reserved. Atmel®, Atmel logo and combinations thereof, Adjacent Key Suppression®, AKS®, QTtouch®, QMatrix® and others are registered trademarks, QT™ and others are trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be registered trademarks or trademarks of others.