Features

- Atmel® maXTouch™ Touchscreen
  - True 12-bit multiple touch with independent XY tracking for up to 10 concurrent touches in real time with touch size reporting
- Number of Channels
  - 1 Touchscreen up to 224 channels (subject to other configurations)
  - Electrode grid configurations of 16–22 X and 8–14 Y lines (subject to 30 lines total)
  - Proximity sensor up to 16 channels (subject to other configurations)
  - Up to 32 channels can be allocated as fixed keys (subject to other configurations)
- Zero Additional Part Count
  - 16X by 14Y, 17X by 13Y, 18X by 12Y, 19X by 11Y and 20X by 10Y matrix sizes implementable with power bypass capacitors only
- Signal Processing
  - Advanced digital filtering using hardware and firmware for better noise removal
  - Auto drift compensation
  - Fast start-up and self-calibration
  - Adjacent Key Suppression® (AKS®) technology
  - Down-scaling and clipping support to match LCD resolution
  - Efficient filtering of the acquisition data for better noise removal
  - Grip and face suppression algorithms to remove unintentional touches
  - Palm touch recovery algorithm for quick restoration to normal state
  - Supports axis flipping and axis switch-over for portrait and landscape modes
- Environmental Conditions
  - Moisture tolerance good
  - Operating temperature -40°C to +85°C for use in a wide range of climates
- Unambiguous Multitouch Performance
  - Advanced algorithms facilitate exceptional user experience
- Scan Speed
  - Maximum single touch >250Hz, subject to configuration
  - Configurable to allow power/speed optimization
  - Programmable timeout for automatic transition from active to idle states
- Response Times
  - Initial latency <10 ms for first touch from idle, subject to configuration
- Sensors
  - Works with PET or glass sensors, including curved profiles
  - Works with all proprietary sensor patterns recommended by Atmel®
  - Works with a passive stylus
- Panel Thickness
  - Glass up to 3 mm, screen size dependent
  - Plastic up to 1.5 mm, screen size dependent
- Interface
  - I²C-compatible slave mode 400 kHz
- Dual-rail Power
  - Interface 1.8V to 3.3V nominal, analog 2.7V to 3.3V nominal
- Packages
  - 49-ball UFBGA 5 x 5 x 0.6 mm, 0.65 ball pitch
  - 48-pin QFN 6 x 6 x 0.6 mm, 0.4 mm pin pitch

**maXTouch
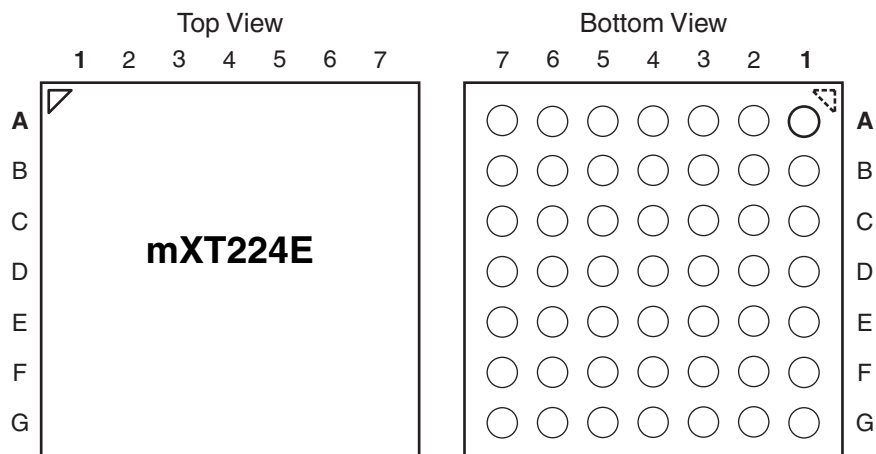224-channel
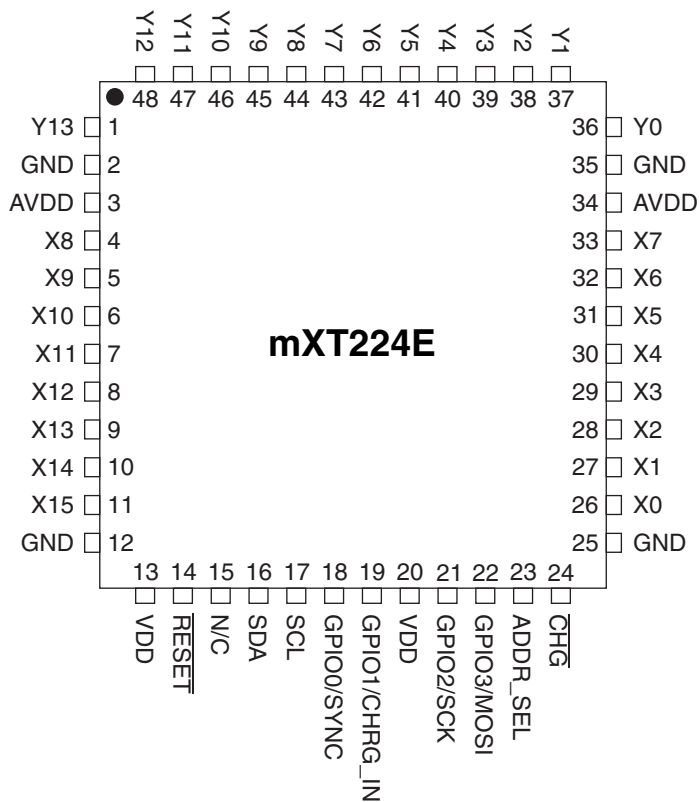Touchscreen
Sensor IC**

mXT224E

maXTouch™

1. Pinout and Schematics

1.1 Pinout Configuration

1.1.1 49-ball UFBGA

1.1.2 48-pin QFN

1.2   Pinout Descriptions

1.2.1   49-ball UFBGA

Table 1-1.   Pin Listing

Ball	Name	Type	Comments	If Unused, Connect To...
A1	AVDD	P	Analog power	–
A2	Y12	I/O	Y line connection or X line in extended mode	Leave open
A3	Y10	I/O	Y line connection or X line in extended mode	Leave open
A4	Y8	I/O	Y line connection or X line in extended mode	Leave open
A5	Y6	I	Y line connection	Leave open
A6	Y4	I	Y line connection	Leave open
A7	Y2	I	Y line connection	Leave open
B1	X8	O	X matrix drive line	Leave open
B2	GND	P	Ground	–
B3	Y11	I/O	Y line connection or X line in extended mode	Leave open
B4	Y9	I/O	Y line connection or X line in extended mode	Leave open
B5	Y5	I	Y line connection	Leave open
B6	Y1	I	Y line connection	Leave open
B7	Y0	I	Y line connection	Leave open
C1	X10	O	X matrix drive line	Leave open
C2	X9	O	X matrix drive line	Leave open
C3	Y13	I/O	Y line connection or X line in extended mode	Leave open
C4	Y7	I	Y line connection	Leave open
C5	Y3	I	Y line connection	Leave open
C6	GND	P	Ground	–
C7	AVDD	P	Analog power	–
D1	X12	O	X matrix drive line	Leave open
D2	X13	O	X matrix drive line	Leave open
D3	X11	O	X matrix drive line	Leave open
D4	GND	P	Ground	–
D5	X7	O	X matrix drive line	Leave open
D6	X5	O	X matrix drive line	Leave open
D7	X6	O	X matrix drive line	Leave open
E1	X14	O	X matrix drive line	Leave open
E2	X15	O	X matrix drive line	Leave open
E3	$\overline{\text{RESET}}$	I	Reset low; has internal 20 kΩ to 60 kΩ pull-up resistor	Vdd
E4	GPIO1/ CHRG_IN	I/O	General purpose I/O Charger present input	Input: GND Output: leave open

Table 1-1. Pin Listing (Continued)

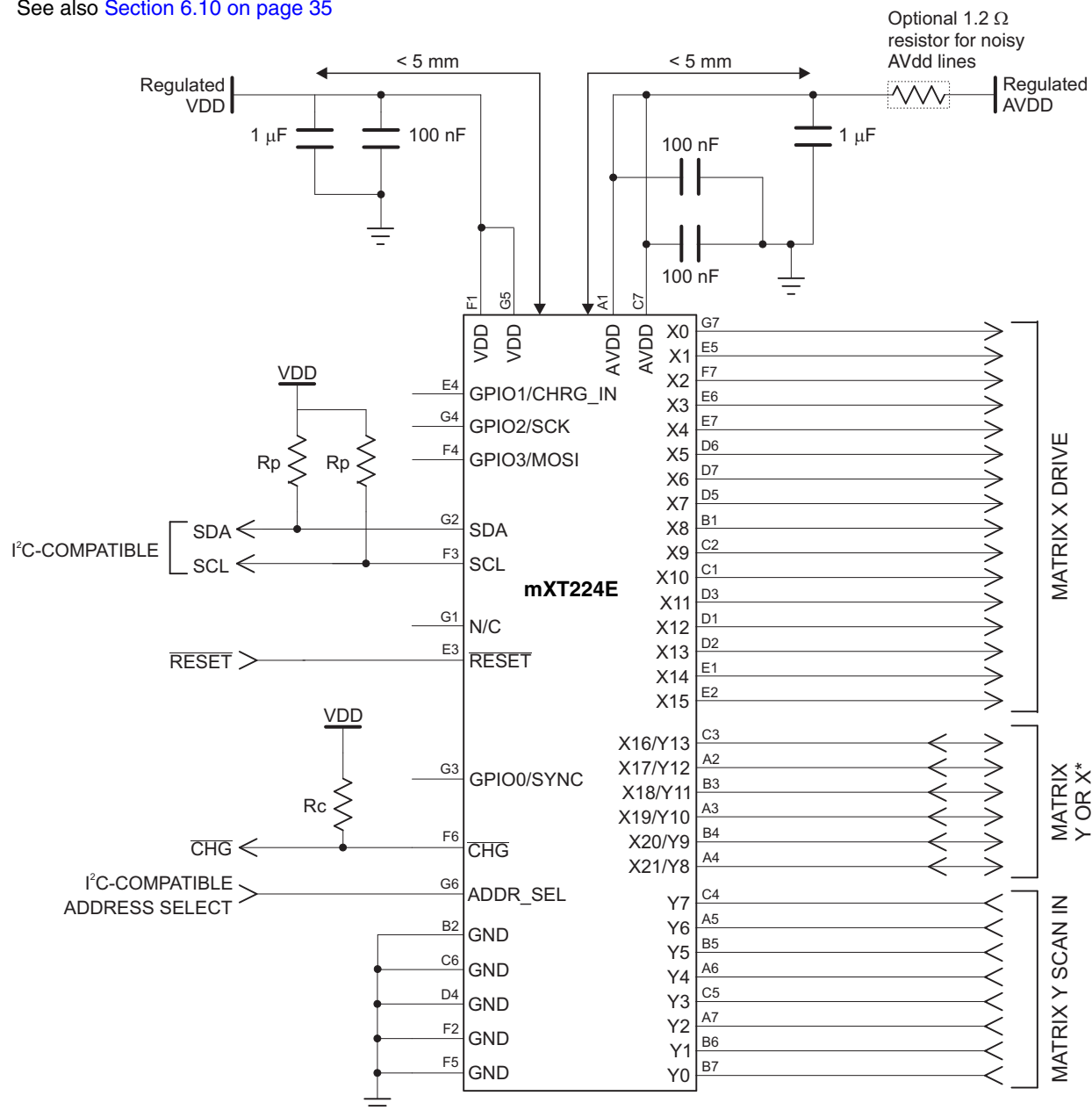| Ball	Name	Type	Comments	If Unused, Connect To...
E5	X1	O	X matrix drive line	Leave open
E6	X3	O	X matrix drive line	Leave open
E7	X4	O	X matrix drive line	Leave open
F1	VDD	P	Digital power	–
F2	GND	P	Ground	–
F3	SCL	OD	Serial Interface Clock	–
F4	GPIO3/ MOSI	I/O	General purpose I/O / Debug data	Input: GND Output: leave open
F5	GND	P	Ground	–
F6	$\overline{\text{CHG}}$ [1]	OD	State change interrupt	–
F7	X2	O	X matrix drive line	Leave open
G1	N/C	–	No connection	Leave open
G2	SDA	OD	Serial Interface Data	–
G3	GPIO0/ SYNC	I/O	General purpose I/O External synchronization	Input: GND Output: leave open
G4	GPIO2/ SCK	I/O	General purpose I/O / Debug clock	Input: GND Output: leave open
G5	VDD	P	Digital power	–
G6	ADDR_SEL	I	I^{2}C-compatible address select	–
G7	X0	O	X matrix drive line	Leave open

1. $\overline{\text{CHG}}$ is momentarily set (approximately 100 ms) as an input after power-up or reset for diagnostic purposes.

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

1.2.2   48-pin QFN

Table 1-2.   Pin Listing

Pin	Name	Type	Comments	If Unused, Connect To...
1	Y13	I/O	Y line connection or X line in extended mode	Leave open
2	GND	P	Ground	–
3	AVDD	P	Analog power	–
4	X8	O	X matrix drive line	Leave open
5	X9	O	X matrix drive line	Leave open
6	X10	O	X matrix drive line	Leave open
7	X11	O	X matrix drive line	Leave open
8	X12	O	X matrix drive line	Leave open
9	X13	O	X matrix drive line	Leave open
10	X14	O	X matrix drive line	Leave open
11	X15	O	X matrix drive line	Leave open
12	GND	P	Ground	–
13	VDD	P	Digital power	–
14	$\overline{\text{RESET}}$	I	Reset low; has internal 20 kΩ to 60 kΩ pull-up resistor	Vdd
15	N/C	–	No connection	Leave open
16	SDA	OD	Serial Interface Data	–
17	SCL	OD	Serial Interface Clock	–
18	GPIO0/ SYNC	I/O	General purpose I/O External synchronization	Input: GND Output: leave open
19	GPIO1/ CHRG_IN	I/O	General purpose I/O Charger present input	Input: GND Output: leave open
20	VDD	P	Digital power	–
21	GPIO2/ SCK	I/O	General purpose I/O / Debug clock	Input: GND Output: leave open
22	GPIO3/ MOSI	I/O	General purpose I/O / Debug data	Input: GND Output: leave open
23	ADDR_SEL	I	I^{2}C-compatible address select	–
24	$\overline{\text{CHG}}$ [1]	OD	State change interrupt	–
25	GND	P	Ground	–
26	X0	O	X matrix drive line	Leave open
27	X1	O	X matrix drive line	Leave open
28	X2	O	X matrix drive line	Leave open
29	X3	O	X matrix drive line	Leave open
30	X4	O	X matrix drive line	Leave open
31	X5	O	X matrix drive line	Leave open
32	X6	O	X matrix drive line	Leave open

Table 1-2. Pin Listing (Continued)

Pin	Name	Type	Comments	If Unused, Connect To...
33	X7	O	X matrix drive line	Leave open
34	AVDD	P	Analog power	–
35	GND	P	Ground	–
36	Y0	I	Y line connection	Leave open
37	Y1	I	Y line connection	Leave open
38	Y2	I	Y line connection	Leave open
39	Y3	I	Y line connection	Leave open
40	Y4	I	Y line connection	Leave open
41	Y5	I	Y line connection	Leave open
42	Y6	I	Y line connection	Leave open
43	Y7	I	Y line connection	Leave open
44	Y8	I/O	Y line connection or X line in extended mode	Leave open
45	Y9	I/O	Y line connection or X18 in 19X by 11Y mode	Leave open
46	Y10	I/O	Y line connection or X line in extended mode	Leave open
47	Y11	I/O	Y line connection or X line in extended mode	Leave open
48	Y12	I/O	Y line connection or X line in extended mode	Leave open

1. $\overline{CHG}$ is momentarily set (approximately 100 ms) as an input after power-up or reset for diagnostic purposes.

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

1.3 Schematics

1.3.1 49-ball UFBGA

NOTE: Bypass capacitors must be X7R or X5R and placed <5 mm away from chip.
See also Section 6.10 on page 35

Optional 1.2 Ω resistor for noisy AVdd lines

< 5 mm

< 5 mm

Regulated VDD

Regulated AVDD

1 µF   100 nF

100 nF   1 µF

100 nF

100 nF

VDD

Rp   Rp

I²C-COMPATIBLE

SDA

SCL

RESET

VDD

Rc

CHG

I²C-COMPATIBLE
ADDRESS SELECT

mXT224E

Ball	Pin
F1	VDD
G5	VDD
A1	AVDD
C7	AVDD
E4	GPIO1/CHRG_IN
G4	GPIO2/SCK
F4	GPIO3/MOSI
G2	SDA
F3	SCL
G1	N/C
E3	RESET
G3	GPIO0/SYNC
F6	CHG
G6	ADDR_SEL
B2	GND
C6	GND
D4	GND
F2	GND
F5	GND

MATRIX X DRIVE

Ball	Pin
G7	X0
E5	X1
F7	X2
E6	X3
E7	X4
D6	X5
D7	X6
D5	X7
B1	X8
C2	X9
C1	X10
D3	X11
D1	X12
D2	X13
E1	X14
E2	X15

MATRIX Y OR X*

Ball	Pin
C3	X16/Y13
A2	X17/Y12
B3	X18/Y11
A3	X19/Y10
B4	X20/Y9
A4	X21/Y8

MATRIX Y SCAN IN

Ball	Pin
C4	Y7
A5	Y6
B5	Y5
A6	Y4
C5	Y3
A7	Y2
B6	Y1
B7	Y0

* NOTE: Y8 to Y13 scan lines may be used as additional X drive lines in extended mode. A 100Ω resistor must be added to X20 (Y9) and X21 (Y8) lines if used as X lines.

1.3.2   48-pin QFN

NOTE: Bypass capacitors must be X7R or X5R and placed <5 mm away from chip.
See also Section 6.10 on page 35
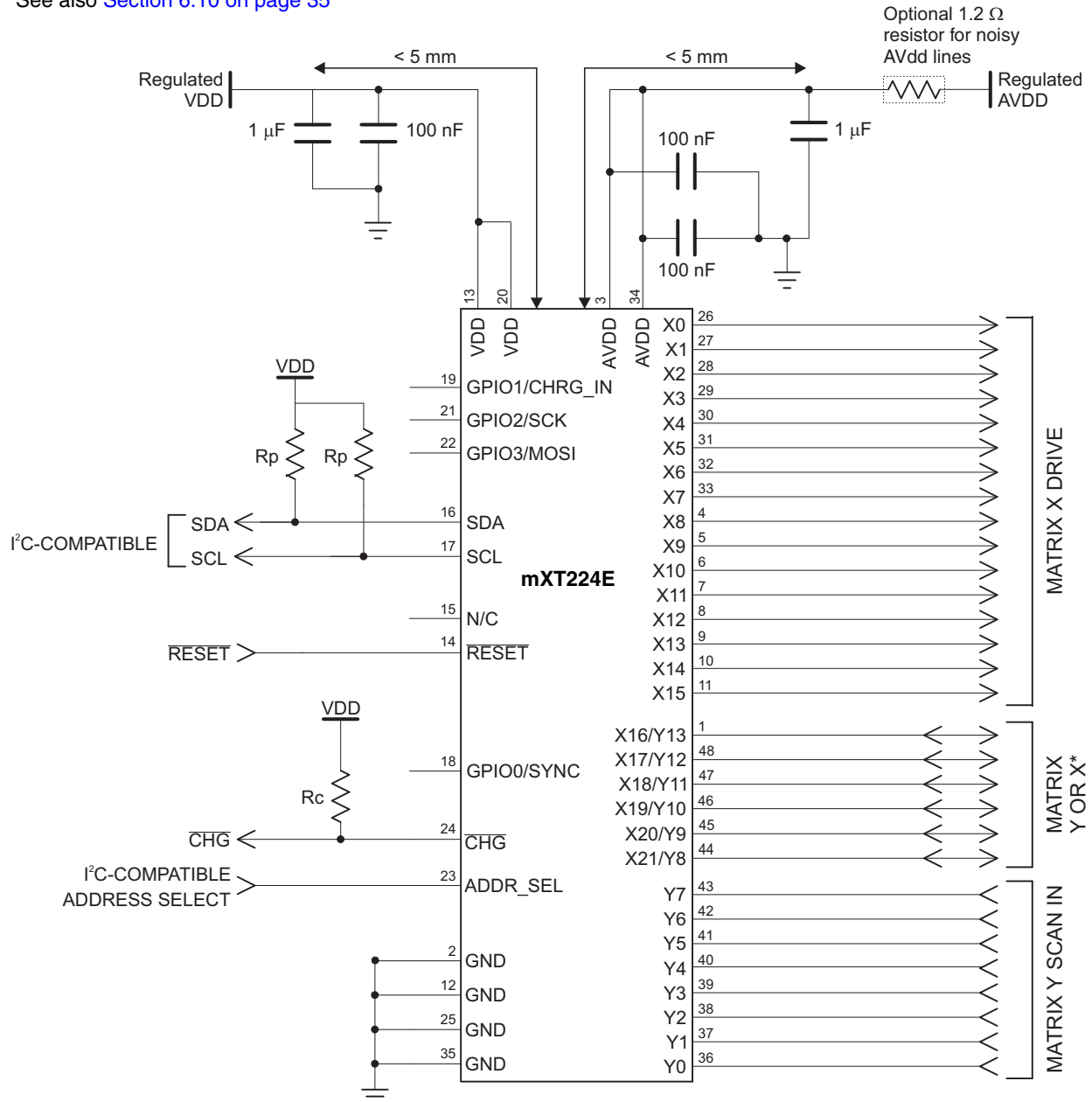
* NOTE: Y8 to Y13 scan lines may be used
as additional X drive lines in extended mode
A 100Ω resistor must be added to X20 (Y9)
and X21 (Y8) lines if used as X lines.

2. Overview of the mXT224E

2.1 Introduction

The Atmel maXTouch family of touch controllers has set a new industry benchmark for capacitive touchscreens with their low current consumption, fast response time and high levels of accuracy. The mXT224E builds on this success to provide a greatly improved user experience:

- **Patented capacitive sensing method** – The mXT224E uses a unique charge-transfer acquisition engine to implement the QMatrix® capacitive sensing method patented by Atmel. This allows the measurement of up to 224 mutual capacitance nodes in under 1 ms. Coupled with a state-of-the-art CPU, the entire touchscreen sensing solution can measure, classify and track a single finger touch every 4 ms if required.

- **Capacitive Touch Engine (CTE)** – The acquisition engine uses an optimal measurement approach to ensure almost complete immunity from parasitic capacitance on the receiver inputs (Y lines). The engine includes sufficient dynamic range to cope with touchscreen mutual capacitances spanning 0.32 pF to 5 pF. This allows great flexibility for use with Atmel's proprietary ITO pattern designs. One and two layer ITO sensors are possible using glass or PET substrates.

- **Processing power** – The main CPU has two powerful, yet low power, microsequencer coprocessors under its control. These combine to allow the signal acquisition, preprocessing, postprocessing and housekeeping to be partitioned in an efficient and flexible way. This gives ample scope for sensing algorithms, touch tracking or advanced shape-based filtering. An in-circuit reflash can be performed over the chip's hardware-driven two-wire interface (I^2C-compatible).

- **Noise filtering improvements** – The maXTouch solution already copes well with LCD noise and RF noise, but operational enhancements allow the mXT224E to cope even better with severe noise.

- **Enhanced user experience** – Atmel's mutual capacitance method provides unambiguous multitouch performance and algorithmic enhancements in the mXT224E give an even better user experience. Hysteresis algorithms ensure that where a light touch is applied this is reported as a continuous touch, even when close to the touch threshold level, to prevent jitter on the screen. Algorithms also ensure that an on-screen cursor is stationary after the touch is removed, or remains on the edge of the visible area after a drag gesture.

- **Interpreting user intention** – Additions to the Object Protocol give the mXT224E enhanced signal processing capabilities. Stylus support allows stylus touches to be detected and distinguished from other touches, such as finger touches. The suppression of unintentional touches from the user's gripping fingers, resting palm or touching cheek or ear also help ensure that the user's intentions are correctly interpreted.

- **Environmental conditions** – The mXT224E brings improved performance in harsh environmental conditions. The robust mutual capacitance measurement method means that the mXT224E works well in the presence of moisture. An operating temperature as low as -40°C allows the user's designs to be used in the coldest of environments. In addition, a high signal-to-noise ratio allows the design to be used with gloves for easy operation in cold environments, during sport or in a workshop.

Overall, the mXT224E represents a step improvement over competing technologies. Its near optimal mix of low power, small size and low part count, coupled with unrivalled true multitouch performance ensures that the mXT224E is set to take capacitive sensing to the next level.

2.2 Understanding Unfamiliar Concepts

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

2.3 Resources

The following datasheet provides essential information on configuring the mXT224E:

- *mXT224E Protocol Guide*

The mXT224E uses the same core technology as the mXT224, so the following documents may also be useful (available by contacting Atmel's Touch Technology division):

- **Configuring the mXT224E:**
  - Application Note: QTAN0058 – *Rejecting Unintentional Touches with the maXTouch*^{TM} *Touchscreen Controllers*
  - Application Note QTAN0050 – *Using the maXTouch*^{TM} *Debug Port*
  - Application Note: QTAN0059 – *Using the maXTouch*^{TM} *Self Test Feature*

- **Touchscreen design and PCB/FPCB layout guidelines:**
  - Application Note QTAN0054 – *Getting Started with maXTouch Touchscreen Designs*
  - Application Note QTAN0048 – *maXTouch PCB/FPCB Layout Guidelines*
  - Touch Sensors Design Guide (document number: 10620-AT42)

- **Bootloading:**
  - Application Note QTAN0051 – *Bootloading Procedure for Atmel*$^®$ *Touch Sensors Based on the Object Protocol*

- **Miscellaneous:**
  - Application Note QTAN0061 – *maXTouch*^{TM} *Sensitivity Effects for Mobile Devices*

3. Touchscreen Basics

3.1 Sensor Construction

A touchscreen is usually constructed from a number of transparent electrodes. These are typically on a glass or plastic substrate. They can also be made using non-transparent electrodes, such as 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 sheet resistance. Thicker ITO yields lower levels of resistance (perhaps tens to hundreds of Ω/square) at the expense of reduced optical clarity. Lower levels of resistance are generally more compatible with capacitive sensing. Thinner ITO leads to higher levels of resistance (perhaps hundreds to thousands of Ω/square) with some of the best optical characteristics.

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

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. 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. They are also an advantage 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.

The mXT224E supports various configurations of electrodes as summarized below:

Touchscreens:	1 Touchscreen allowed 3X x 3Y minimum (depends on screen resolution) Native mode: 16X x 14Y maximum (subject to other configurations) Extended modes: 17X x 13Y, 18X x 12Y, 19X x 11Y, 20X x 10Y, 21X x 9Y [1], 22X x 8Y [1] maximum
Keys:	Up to 32 keys allowed (subject to other configurations)
Proximity Sensors:	1 Proximity Sensor allowed occupying up to 16 channels in any configuration (subject to other configurations), 1X x 1Y minimum

---

1. Requires external 100Ω resistor

3.3 Scanning Sequence

All channels are scanned in sequence by the mXT224E. There is full parallelism in the scanning sequence to improve overall response time. The channels are scanned by measuring capacitive changes at the intersections formed between the first X line and all the Y lines. Then the intersections between the next X line and all the Y lines are scanned, and so on, until all X and Y combinations have been measured.

The device can be configured 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, and so on. An 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.

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

The mXT224E supports mixed configurations of different touch objects, each having independent threshold controls to allow fine tuning with mixed configurations.

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.

Front panel dielectric material has a direct bearing on sensitivity. Plastic front panels are usually suitable up to about 1.5 mm, and glass up to about 3 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.

The mXT224E is suitable for various stackup arrangments (see Table 3-1). In this table, three types of stackup are considered:

- Single layer co-planar construction, either with a diamond pattern or Atmel's patented snowflake design.
- Co-planar with a ground shield; that is, a co-planar construction with a separate ground plane on another layer.
- Two layer designs in which the Y sensor is one layer and the X sensor flooded on another layer behind it (as with Atmel's flooded-X style).

Table 3-1 lists the various mechanical stackups and their suitabliity for use with the mXT224E.

Table 3-1. mXT224E Suitability for Various Mechanical Stackups

Display Construction	Mechanical Stackup		
	Co-planar	**Co-planar with Shield**	**Two Layers**
OLED, No Air Gap	Yes, but depends on project	Suitable	Suitable
OLED, with Air Gap $\leq$ **0.25 mm**	Yes, but depends on project	Suitable	Suitable
LCD, No Air Gap	Not suitable	Suitable	Yes, but depends on project
LCD, with Air Gap $\leq$ **0.25 mm**	Not suitable	Suitable	Yes

4.   Detailed Operation

4.1   Power-up/Reset

There is an internal Power-on Reset (POR) in the device.

The device must be held in $\overline{\text{RESET}}$ (active low) while both the digital and analog power supplies (Vdd and AVdd) are powering up. If a slope or slew is applied to the digital or analog supplies, Vdd and AVdd must reach their nominal values before the $\overline{\text{RESET}}$ signal is deasserted (that is, goes high). This is shown in Figure 4-1. See Section 6.3 on page 31 for nominal values for Vdd and AVdd.

Figure 4-1.   Power Sequencing on the mXT224E
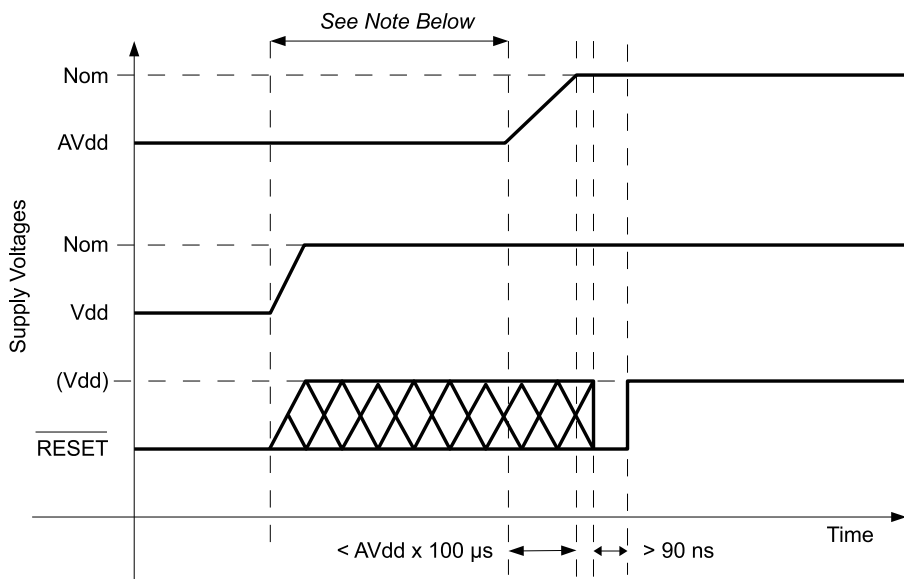
NOTE: Vdd and Avdd can be powered up in either order.
There is no prerequisite for the length of time between Vdd and AVdd powering up.

Note that there are no specific power-up, or power-down sequences required for the mXT224E. This means that the digital or analog supplies can be applied independently and in any order during power-up.

After power-up, the device takes ~22 ms before it is ready to start communications. Vdd must drop to below 1V in order to effect a proper POR. See Section 6 for further specifications.

If the $\overline{\text{RESET}}$ line is released before the AVDD supplies have reached their nominal voltage (see Figure 4-2), then the host should send either a calibration command or a reset command.

Figure 4-2. Power Sequencing on the mXT224E – Late rise on AVDD
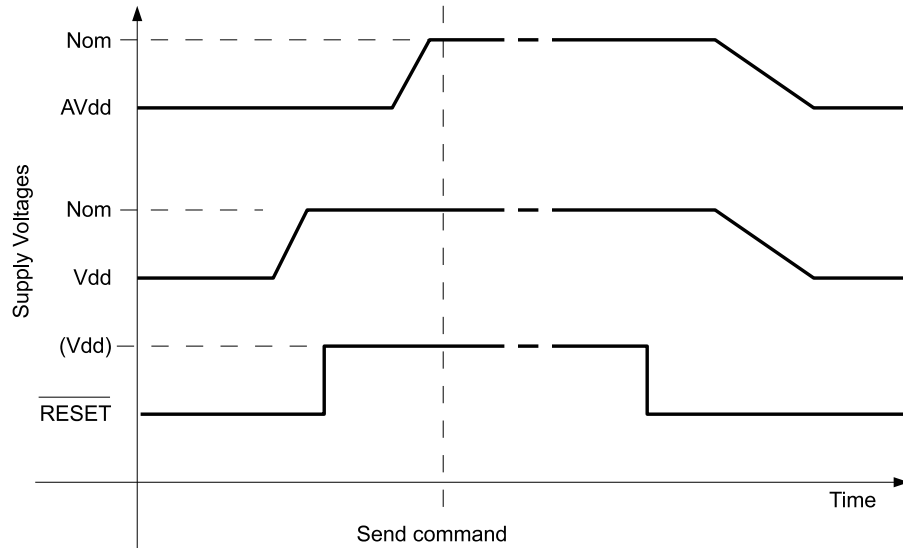
The $\overline{RESET}$ pin can be used to reset the device whenever necessary. The $\overline{RESET}$ pin must be asserted low for at least 90 ns to cause a reset. After releasing the $\overline{RESET}$ pin the device takes ~22 ms before it is ready to start communications. It is recommended to connect the $\overline{RESET}$ pin to a host controller to allow it to initiate a full hardware reset without requiring a power-down.

Note that the voltage level on the $\overline{RESET}$ pin of the device must never exceed Vdd (digital supply voltage).

A software reset command can be used to reset the chip (refer to the Command Processor object in the *mXT224E Protocol Guide*). A software reset takes ~22 ms. After the chip has finished it asserts the $\overline{CHG}$ line to signal to the host that a message is available. The reset flag is set in the Message Processor object to indicate to the host that it has just completed a reset cycle. This bit can be used by the host to detect any unexpected brownout events. This allows the host take any necessary corrective actions, such as reconfiguration.

A checksum check is performed on the configuration settings held in the nonvolatile memory. If the checksum does not match a stored copy of the last checksum, then this indicates that the settings have become corrupted. This is signaled to the host by setting the configuration error bit in the message data for the Command Processor object (refer to the *mXT224E Protocol Guide* for more information).

Note that the $\overline{CHG}$ line is momentarily set (approximately 100 ms) as an input after power-up or reset for diagnostic purposes. It is therefore particularly important that the line should be allowed to float high via the $\overline{CHG}$ line pull-up resistors during this period.

4.2 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 (that is, activated).

OR

• The channel is already enabled and one of the following applies:

– The channel is held in detect for longer than the Touch Automatic Calibration setting (refer to the *mXT224E Protocol Guide* for more information on TCHAUTOCAL setting in the Acquisition Configuration object).

– The signal delta on a channel is at least the touch threshold (TCHTHR) in the anti-touch direction, while no other touches are present on the channel matrix (refer to the *mXT224E Protocol Guide* for more information on the TCHTHR field in the Multiple Touch Touchscreen and Key Array objects).

– The user issues a recalibrate command.

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

Note that the device performs a global calibration; that is, all the channels are calibrated together.

4.3 Sensor Acquisition

The maximum X line acquisition time on the mXT224E is 5 ms. Care should be taken to ensure that the total time for one X line configured by the Acquisition Configuration T8 and CTE Configuration T46 objects does not exceed this.

4.4 Communications

4.4.1 Communications Protocol

The mXT224E uses an I^{2}C-compatible interface for communication. See Appendix C on page 45 for details of the I^{2}C-compatible protocol.

The device presents data packets when internal changes have occurred. The $\overline{\text{CHG}}$ line going active signifies that a new data packet is available, thus providing an interrupt-style interface.

4.4.2 I^{2}C-compatible Addresses

The mXT224E supports two I^{2}C-compatible device addresses that are selected using the ADDR_SEL line at start-up. The two internal I^{2}C-compatible device addresses are 0x4A (ADDR_SEL low) and 0x4B (ADDR_SEL high). These are shifted left to form the SLA+W or SLA+R address when transmitted over the I^{2}C-compatible interface, as shown in Table 4-1.

Table 4-1. Format of SLA+W and SLA+R

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Address: 0x4A or 0x4B							Read/write

4.4.3 Writing To the Device

A WRITE cycle to the device consists of a START condition followed by the I²C-compatible address of the device (SLA+W). 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, and the second byte is the Most Significant Byte (MSByte). This address is then stored as the address pointer.

Subsequent bytes in a multibyte transfer form the actual data. These are written to the location of the address pointer, location of the address pointer +1, location of the address pointer + 2, and so on. The address pointer returns to its starting value when the WRITE cycle's STOP condition is detected.

Figure 4-3 shows an example of writing four bytes of data to contiguous addresses starting at 0x1234.

Figure 4-3. Example of a Four-byte Write Starting at Address 0x1234
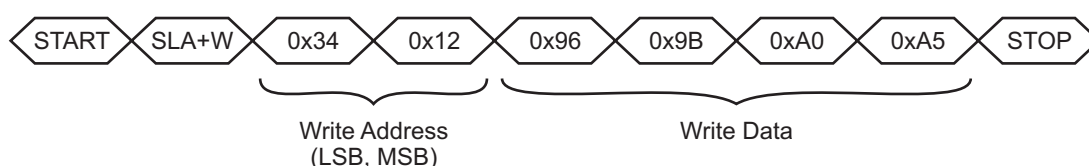
4.4.4 I²C-compatible Writes in Checksum Mode

In I²C-compatible checksum mode an 8-bit CRC is added to all I²C-compatible writes. The CRC is sent at the end of the data write as the last byte before the STOP condition. All the bytes sent are included in the CRC, including the two address bytes. Any command or data sent to the device is processed even if the CRC fails.

To indicate that a checksum is to be sent in the write, the most significant bit of the MSByte of the address is set to 1. For example, the I²C-compatible command shown in Figure 4-4 writes a value of 150 (0x96) to address 0x1234 with a checksum. The address is changed to 0x9234 to indicate checksum mode.

Figure 4-4. Example of a Write To Address 0x1234 With a Checksum
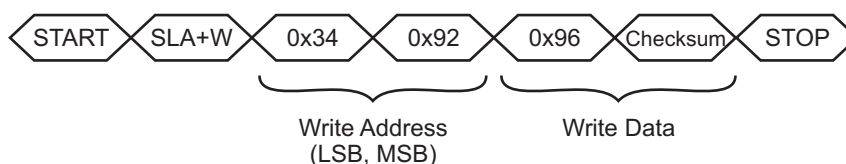
4.4.5 Reading From the Device

Two I²C-compatible bus activities must take place to read from the device. The first activity is an I²C-compatible write to set the address pointer (LSByte then MSByte). The second activity is the actual I²C-compatible read to receive the data. The address pointer returns to its starting value when the READ cycle's NACK is detected.

It is not necessary to set the address pointer before every read. The address pointer is updated automatically after every read operation. The address pointer will be correct if the reads occur in order. In particular, when reading multiple messages from the Message Processor object, the address pointer is automatically reset to allow continuous reads (see Section 4.4.6).

The WRITE and READ cycles consist of a START condition followed by the I^{2}C-compatible address of the device (SLA+W or SLA+R respectively).

Figure 4-5 shows the I^{2}C-compatible commands to read four bytes starting at address 0x1234.

Figure 4-5. Example of a Four-byte Read Starting at Address 0x1234

Set Address Pointer

START ⟩⟨ SLA+W ⟩⟨ 0x34 ⟩⟨ 0x12 ⟩⟨ STOP

Read Address
(LSB, MSB)

Read Data

START ⟩⟨ SLA+R ⟩⟨ 0x96 ⟩⟨ 0x9B ⟩⟨ 0xA0 ⟩⟨ 0xA5 ⟩⟨ STOP

Read Data
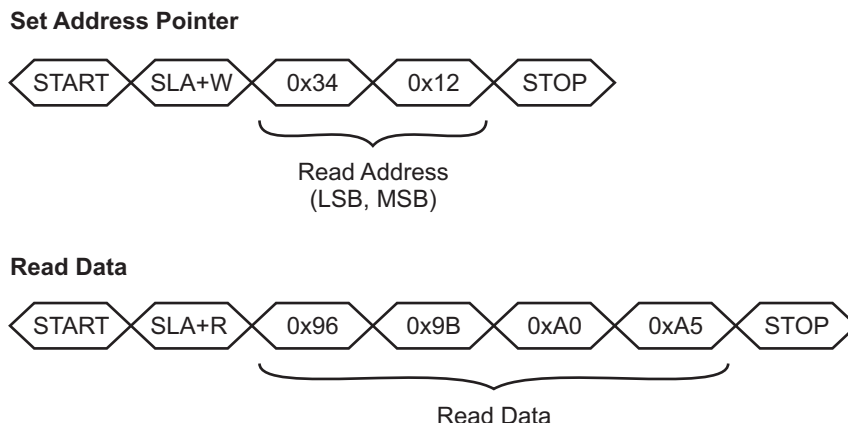
4.4.6 Reading Status Messages with DMA

The device facilitates the easy reading of multiple messages using a single continuous read operation. This allows the host hardware to use a direct memory access (DMA) controller for the fast reading of messages, as follows:

1. The host uses a write operation to set the address pointer to the start of the Message Count object, if necessary. [1] If a checksum is required on each message, the most significant bit of the MSByte of the read address must be set to 1.

2. The host starts the read operation of the message by sending a START condition.

3. The host reads the Message Count object (one byte) to retrieve a count of the pending messages (refer to the *mXT224E Protocol Guide* for details).

4. The host calculates the number of bytes to read by multipling the message count by the size of the Message Processor object. [2]

   Note that the size of the Message Processor object as recorded in the Object Table includes a checksum byte; if a checksum is not required, this should be deducted from the size of the object. That is: number of bytes = count x (size-1).

5. The host reads the calculated number of message bytes. It is important that the host does *not* send a STOP condition during the message reads, as this will terminate the continuous read operation and reset the address pointer. No START and STOP conditions must be sent between the messages.

6. The host sends a NACK at the end of the read operation after the last message has been read. The NACK resets the address pointer to the start of Message Count object.

Figure 4-6 shows an example of using a continuous read operation to read three messages from the device without a checksum. Figure 4-7 on page 20 shows the same example with a checksum.

---

1. The NACK at the end of the read resets the address pointer to its intial location, so it may already be pointing at the Message Count object following a previous message read.

2. The host should have already read the size of the Message Processor object in its initialization code (see Section 5.3 on page 29).

Figure 4-6. Continuous Message Read Example – No Checksum

Set Address Pointer

START SLA+W LSB MSB STOP

Start Address of
Message Count Object

Continuous
Read

Read Message Count

START SLA+R Count=3

Message Count Object

Read Message Data

(size-1) Bytes

Report ID Data • • • Data

Message Processor Object – Message #1

Report ID Data • • • Data

Message Processor Object – Message #2

Report ID Data • • • Data STOP

Message Processor Object – Message #3
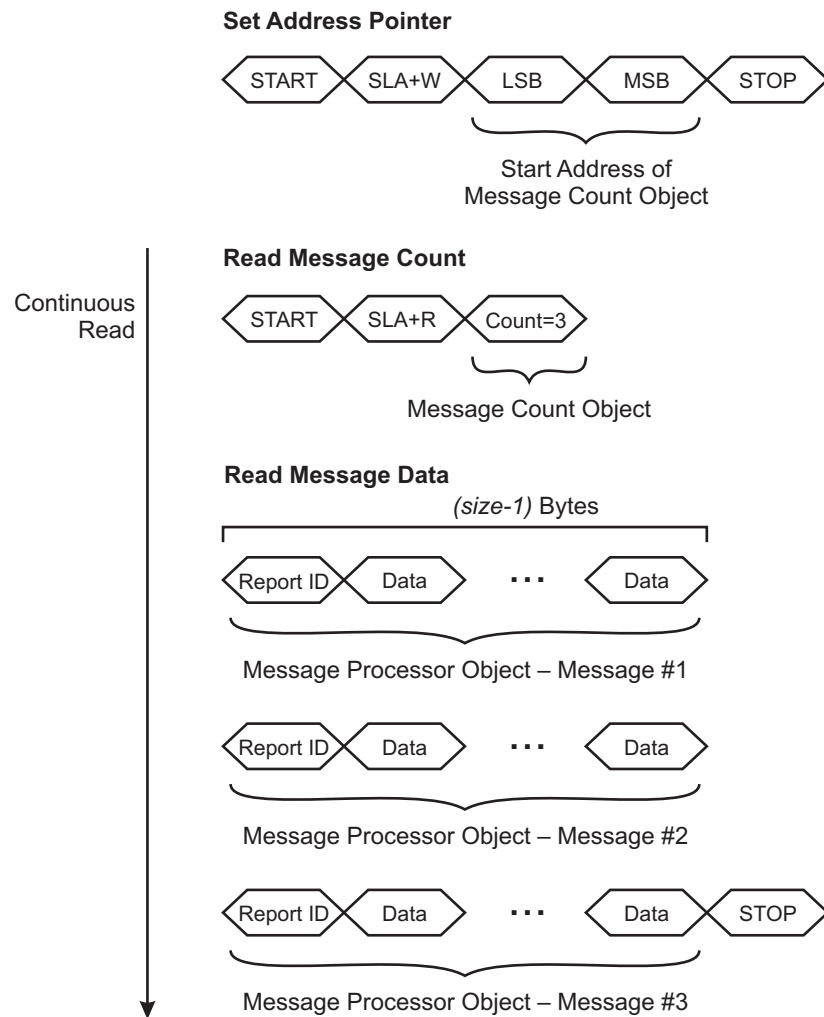
Figure 4-7. Continuous Message Read Example – I²C-compatible Checksum Mode
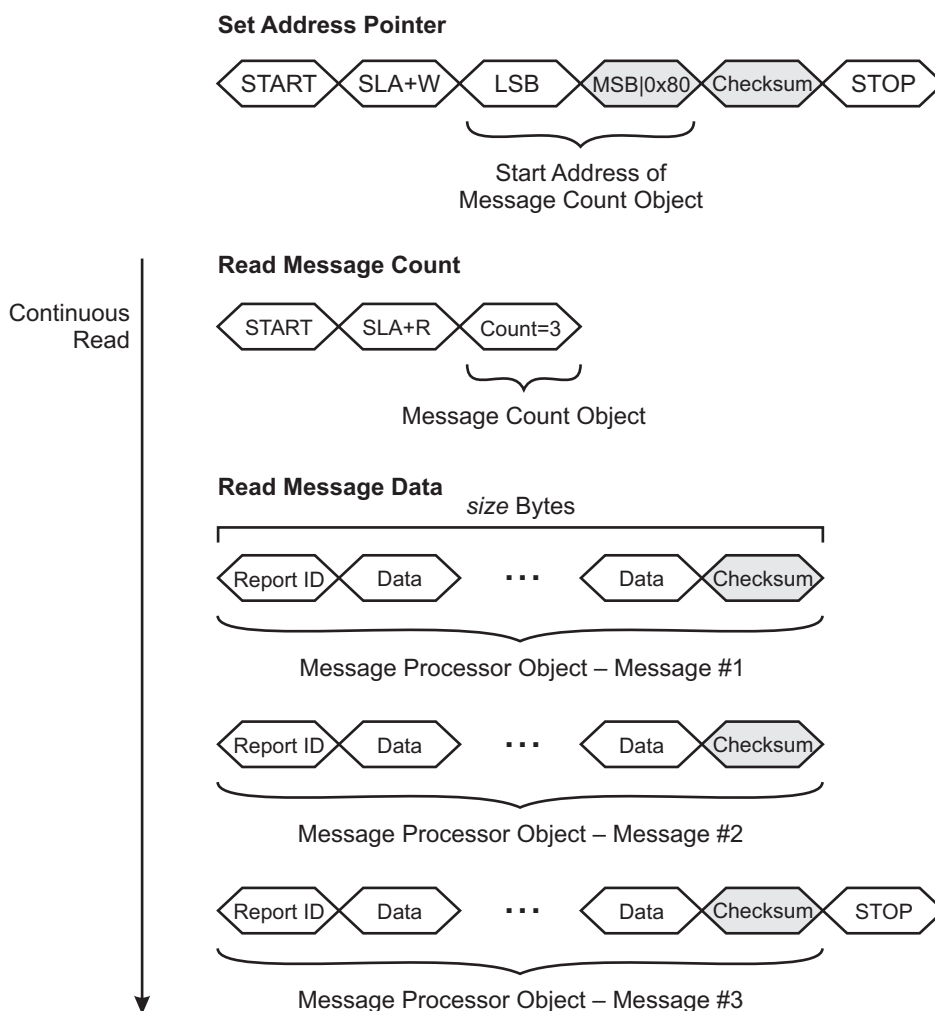
There are no checksums added on any other I²C-compatible reads. An 8-bit CRC can be added, however, to all I²C-compatible writes, as described in Section 4.4.4.

An alternative method of reading messages using the $\overline{\text{CHG}}$ line is given in Section 4.4.7.

4.4.7 $\overline{\text{CHG}}$ Line

The $\overline{\text{CHG}}$ line is an active-low, open-drain output that is used to alert the host that a new message is available in the Message Processor object. This provides the host with an interrupt-style interface with the potential for fast response times. It reduces the need for wasteful I²C-compatible communications.

The $\overline{\text{CHG}}$ line remains low as long as there are messages to be read. The host should be configured so that the $\overline{\text{CHG}}$ line is connected to an interrupt line that is level-triggered. The host should not use an edge-triggered interrupt as this means adding extra software precautions.

The $\overline{\text{CHG}}$ line should be allowed to float during normal usage. This is particularly important after power-up or reset (see Section 4.1 on page 14).

A pull-up resistor is required, typically 10 kΩ to Vdd.

The $\overline{\text{CHG}}$ line operates in two modes (as defined by the Communications Configuration object; refer to the *mXT224E Protocol Guide*).

In Mode 0:

1. The $\overline{\text{CHG}}$ line goes low to indicate that a message is present.
2. The $\overline{\text{CHG}}$ line goes high when the first byte of the first message (that is, its report ID) has been sent and acknowledged (ACK sent) and the next byte has been prepared in the buffer.
3. The NACK at the end of an I^{2}C-compatible read causes the $\overline{\text{CHG}}$ line to stay high if there are no more messages. Otherwise the $\overline{\text{CHG}}$ line goes low to indicate a further message.

Mode 0 allows the host to read the messages until a report ID of 255 ("invalid message") is received. Alternatively, the host can send a NACK after receiving the last byte of any message and send a STOP condition to end the transfer. If and when there is another present, the $\overline{\text{CHG}}$ line goes low, as in step 1. In this mode the state of the $\overline{\text{CHG}}$ line does not need to be checked during the I^{2}C-compatible read.

In Mode 1:

1. The $\overline{\text{CHG}}$ line goes low to indicate that a message is present.
2. The $\overline{\text{CHG}}$ line remains low while there are further messages to be sent after the current message.
3. The $\overline{\text{CHG}}$ line goes high again only once the first byte of the last message (that is, its report ID) has been sent and acknowledged (ACK sent) and the next byte has been prepared in the output buffer.

Mode 1 allows the host to continually read the messages until the $\overline{\text{CHG}}$ line goes high, and the state of the $\overline{\text{CHG}}$ line determines whether or not the host should continue receiving messages from the device.

Note: The state of the $\overline{\text{CHG}}$ line should be checked only between messages and not between the bytes of a message. The precise point at which the $\overline{\text{CHG}}$ line changes state cannot be predicted and so the state of the $\overline{\text{CHG}}$ line cannot be guaranteed between bytes.

Figure 4-8. $\overline{\text{CHG}}$ Line Modes for I²C-compatible Transfers

Mode 0

Mode 1
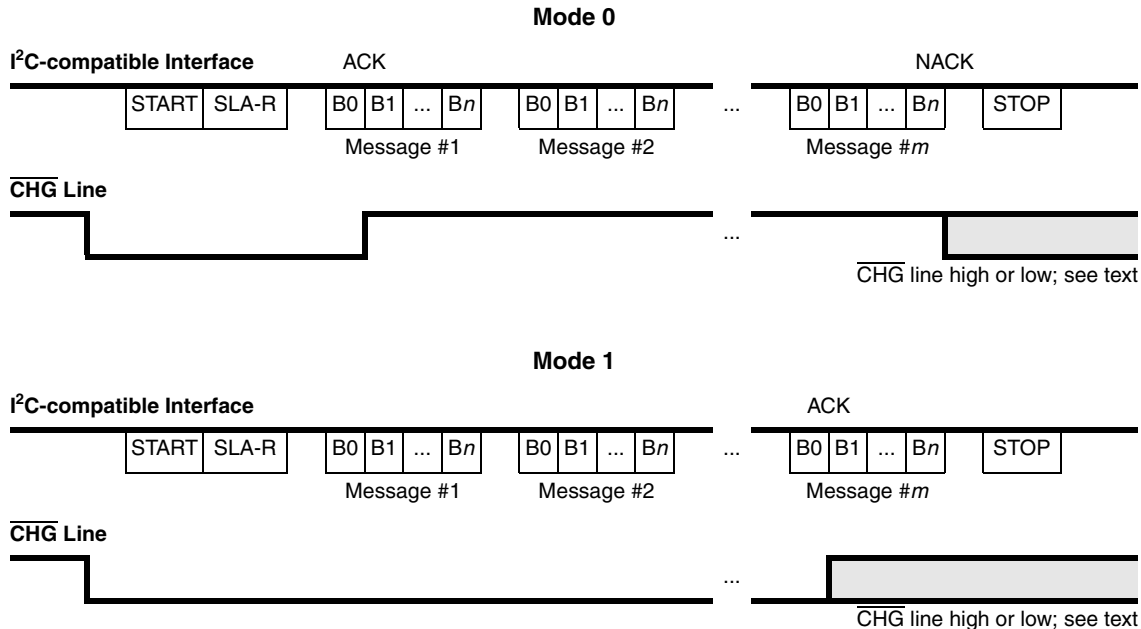
4.4.8 SDA, SCL

The I²C-compatible bus transmits data and clock with SDA and SCL, respectively. These are open-drain. The I²C-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 I²C-compatible device is pulling it down.

The termination resistors commonly range from 1 kΩ to 10 kΩ. They should be chosen so that the rise times on SDA and SCL meet the I²C-compatible specifications (see Section 6.6 on page 32).

4.4.9 Clock Stretching

The mXT224E supports clock stretching in accordance with the I²C specification. It may also instigate a clock stretch if a communications event happens during a period when the mXT224E is busy internally. The maximum clock stretch is approximately 10 – 15 ms.

The mXT224E has an internal bus monitor that can reset the internal I²C-compatible hardware if SDA or SCL is stuck low for more than 100 – 200 ms. This means that if a clock stretch of more than 200 ms is seen by the mXT224E, then any ongoing transfers with the mXT224E may be corrupted. The bus monitor is enabled or disabled using the Communications Configuration object. Refer to the *mXT224E Protocol Guide* for more information.

4.5 Operational Modes

The device operates in two modes: active (touch detected) and idle (no touches detected). Both modes operate as a series of burst cycles. Each cycle consists of a short burst (during which measurements are taken) followed by an inactive sleep period. The difference between these modes is the length of the cycles. Those in idle mode typically have longer sleep periods. The cycle length is configured using the IDLEACQINT and ACTVACQINT settings in the Power Configuration object. In addition, an Active to Idle timeout (ACTV2IDLETO) setting is provided.

Refer to the *mXT224E Protocol Guide* for full information on how these modes operate, and how to use the settings provided.

4.6 Sensor Layout Modes

The mXT224E has a total of 30 lines, which can be distributed between the X and Y lines, with limitations. In native mode these are arranged as a matrix of 16X x 14Y lines, giving 224 available channels.

In extended modes, pins Y8 – Y13 can be used as X drivers if required (see Table 4-2).

Table 4-2. Lines in Extended Modes

Y Line	Reassigned to X Line...
Y13	X16
Y12	X17
Y11	X18
Y10	X19
Y9	X20
Y8	X21

This allows for the permutations shown in Table 4-3.

Table 4-3. Allocation of X and Y Lines

XY Matrix	Number of Channels	Highest Y Line	Mode
16X x 14Y	224	Y13	Native
17X x 13Y	221	Y12	Extended
18X x 12Y	216	Y11	Extended
19X x 11Y	209	Y10	Extended
20X x 10Y	200	Y9	Extended
21X x 9Y	189	Y8	Extended
22X x 8Y	176	Y7	Extended

In 21X x 9Y and 22X x 8Y extended modes, a 100Ω resistor is required in series with each extended X driver (X20 or X21 as applicable). The resistors should be no further than 10 mm from the chip body. They can be 0201 type or larger, 5 percent 200 ppm/°C or better.

4.7 Touchscreen Layout

4.7.1 Introduction

The physical matrix can be configured to have one or more touch objects. These are configured using the appropriate touch objects (Multiple Touch Touchscreen, Key Array, Proximity Sensor). It is not mandatory to have all the allowable touch objects present. Only those that you wish to use need to be enabled, as the objects are disabled by default. Refer to the *mXT224E Protocol Guide* for more information on configuring the touch objects.

When designing the physical layout of the touch panel, obey the following rules:

- Each touch object should be a regular rectangular shape in terms of the lines it uses.
- The design of the touch objects does not physically need to be on a strict XY grid pattern.

4.7.2 Example Layout – A Multiple Touch Touchscreen and a Key Array

Figure 4-9 shows an example layout of a Multiple Touch Touchscreen with a Key Array. Note how the touch objects do not share the same physical XY grid pattern.

Figure 4-9. Example Screen Layout (Native Mode)
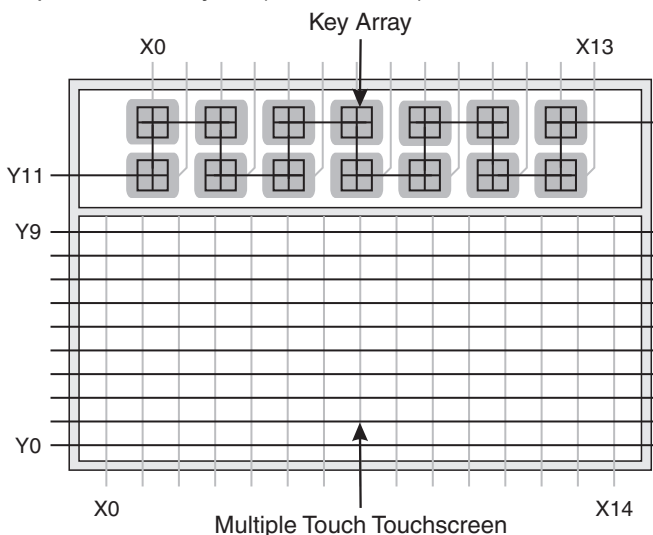
The logical XY grid allocation for this layout (as configured by a Multiple Touch Touchscreen and a Key Array object) is shown in Figure 4-10.

Figure 4-10. Example Touchscreen Configuration (Native Mode)
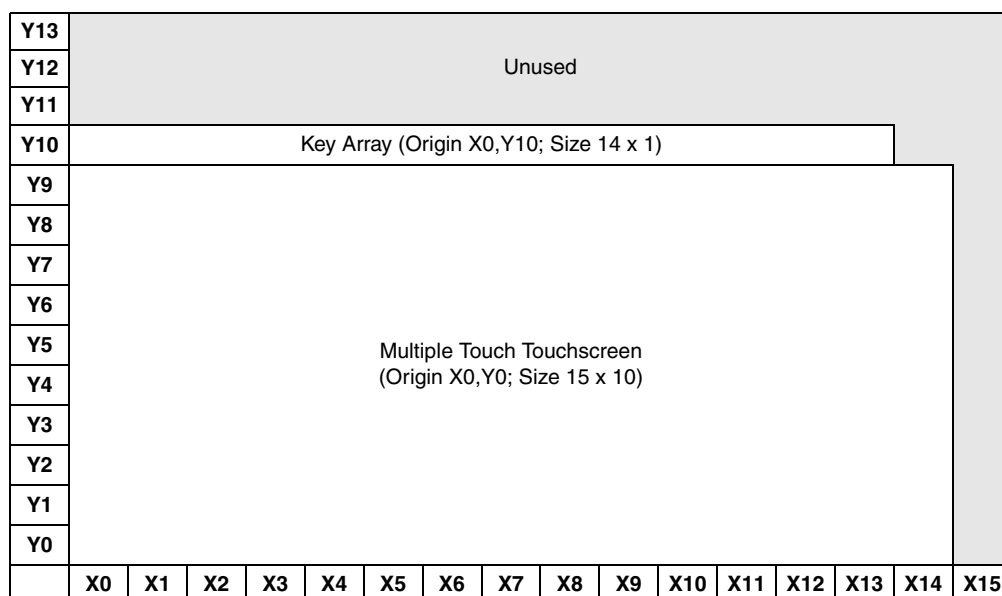
4.7.3   Example Layout – A Multiple Touch Touchscreen and a Proximity Sensor

Figure 4-11 and Figure 4-12 show an example layout of a Multiple Touch Touchscreen with a Proximity Sensor. This example uses the 18X x 12Y extended mode with Y13 and Y12 assigned to X16 and X17 respectively. Note how the proximity sensor's X and Y lines physically cover a larger area than the touchscreen, even though it logically shares X lines with the touchscreen.

Figure 4-11. Example Screen Layout (18X x 12Y Extended Mode)
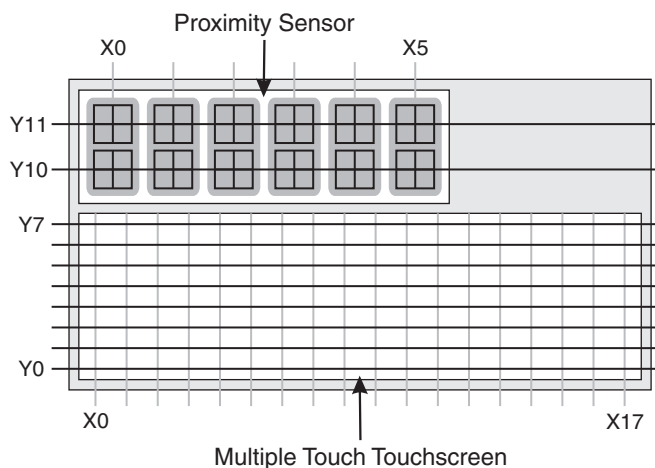
Figure 4-12. Example Touchscreen Configuration (18X x 12Y Extended Mode)

Y11	Proximity Sensor (Origin X0,Y10; Size 6 x 2)						Unused											
Y10																		
Y9																		
Y8																		
Y7	Multiple Touch Touchscreen (Origin X0,Y0; Size 18 x 8)																	
Y6																		
Y5																		
Y4																		
Y3																		
Y2																		
Y1																		
Y0																		
	X0	**X1**	**X2**	**X3**	**X4**	**X5**	**X6**	**X7**	**X8**	**X9**	**X10**	**X11**	**X12**	**X13**	**X14**	**X15**	**X16**	**X17**

4.8 Signal Processing

4.8.1 Adjacent Key Suppression Technology

Adjacent Key Suppression (AKS) technology is a patented method used to detect which touch object is touched when objects are located close together. A touch in a group of AKS objects is only indicated on the object in that group that is touched first. This is assumed to be the intended object. Once an object in an AKS group is in detect, there can be no further detections within that group until the object is released. Objects can be in more than one AKS group.

Note that AKS technology works best when it operates in conjunction with a detect integration setting of several acquisition cycles.

The mXT224E has two levels of AKS. The first level works between the Touchscreen and Key Array objects. The touch objects are assigned to AKS groups. If a touch occurs within one of the touch objects in a group, then touches within other objects inside that group are suppressed. For example, if a Touchscreen and Key Array are placed in the same AKS group, then a touch in the Touchscreen will suppress touches in the Key Array, and vice versa.

The second level of AKS is internal AKS within a Key Array object (note that internal AKS is not present on other touch objects). If internal AKS is enabled, then when one key is touched, touches on all the other keys within the Key Array are suppressed.

AKS is configured using the Multiple Touch Touchscreen or Key Array objects (refer to the *mXT224E Protocol Guide* for more information).

Note: If a touch is in detect and then AKS is enabled, that touch will not be forced out of detect. It will not go out of detect until the touch is released. AKS will then operate normally. This applies to both levels of AKS.

4.8.2 Detection Integrator

The device features a touch detection integration mechanism. This acts to confirm a detection in a robust fashion. A counter is incremented each time a touch has exceeded its threshold and has remained above the threshold for the current acquisition. When this counter reaches a preset limit the sensor is finally 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.

The detection integrator is configured using the appropriate touch objects (Multiple Touch Touchscreen, Key Array, Proximity Sensor). Refer to the *mXT224E Protocol Guide* for more information. You may also wish to refer to application note QTAN0049, *mXT224 Power and Speed Considerations*, for more discussion on the touch detection integration (TCHDI) setting.

4.9 Circuit Components

4.9.1 Bypass Capacitors

One 100 nF bypass capacitor and one 1 µF bypass capacitor must be used on the Vdd digital supply. Two 100 nF bypass capacitors and one 1 µF bypass capacitor must be used on the AVdd analog supply. That is, a total of 5 ceramic X7R or X5R bypass capacitors are needed, as detailed on the schematics in Section 1.3 on page 7. The PCB traces connecting the capacitors to the pins of the device must not exceed 5 mm in length. This limits any stray inductance that would reduce filtering effectiveness. See also Section 6.10 on page 35.

4.9.2 PCB Cleanliness

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

CAUTION: If a PCB is reworked to correct soldering faults relating to the mXT224E, or to any associated traces or components, be sure that you fully understand the nature of the flux used during the rework process. Leakage currents from hygroscopic ionic residues can stop capacitive sensors from functioning. If you have any doubts, a thorough cleaning after rework may be the only safe option.

4.9.3 QFN Package Restrictions

The central pad on the underside of the QFN chip should be connected to ground. Do not run any tracks underneath the body of the chip, only ground. Figure 4-13 shows an example of good/bad tracking.

Figure 4-13. Examples of Good and Bad Tracking

Example of GOOD Tracking   Example of BAD Tracking
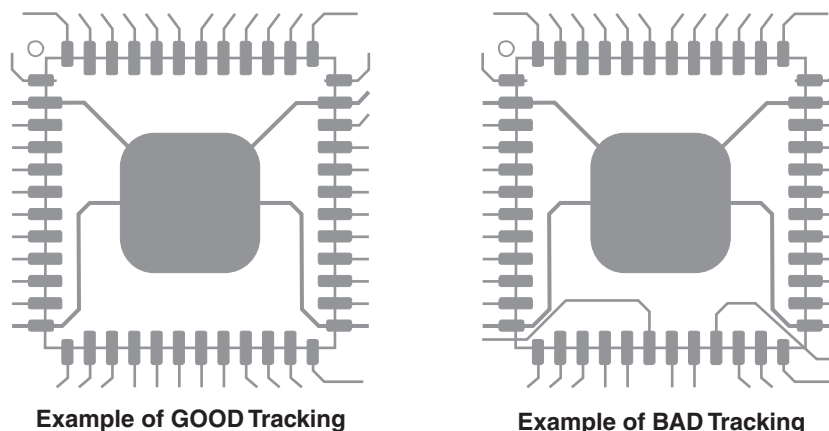
4.9.4   Supply Quality

The mXT224E has good Power Supply Rejection Ratio properties. Poorly regulated and/or noisy power can significantly reduce performance. See .

Always operate the mXT224E with a well-regulated and clean AVdd supply. It supplies the sensitive analog stages in the chip. Refer to QTAN0063, *maXTouch EMC Considerations*, for peak-to-peak noise requirements on this supply.

There is no separate GND return pin for the analog stages. You are advised to consider return current paths from other current consumers in the system. Try to provide a separate heavy GND return trace or flood for the mXT224E that connects at a PSU star-point or connector pin. This helps to avoid inductive transient voltages coupling into the capacitive measurements made by the chip.

Vdd is used internally to power the digital stages in the chip and is far less critical to overall performance. It is still recommended, however, that a low noise supply is used to prevent cross-talk into the analog sections.

4.9.5   Supply Sequencing

Vdd and AVdd can be powered independently of each other without damage to the device.

Make sure that any lines connected to the device are below or equal to Vdd during power-up. For example, if $\overline{\text{RESET}}$ is supplied from a different power domain to the mXT224E's Vdd pin, make sure that it is held low when Vdd is off. If this is not done, the $\overline{\text{RESET}}$ signal could parasitically couple power via the mXT224E's $\overline{\text{RESET}}$ pin into the Vdd supply.

4.9.6   Oscillator

The device has an internal oscillator. No external oscillator or clock input is required.

4.10   Debugging

The mXT224E provides a mechanism for obtaining raw data for development and testing purposes by reading data from the Diagnostic Debug object. Refer to the *mXT224E Protocol Guide* for more information on this object.

A second mechanism is provided that allows the host to read the real-time raw data using the low-level debug port. Refer to QTAN0050, *Using the mXT224 Debug Port*, for more information.

There is also a Self Test object that runs self-test routines in the device to find hardware faults on the sense lines and the electrodes. Refer to the *mXT224E Protocol Guide* and QTAN0059, *Using the mXT224 Self Test Feature*, for more information.

4.11   Configuring the mXT224E

The mXT224E has an object-based protocol that organizes the features of the device into objects that can be controlled individually. This is configured using the Object Protocol common to many of Atmel's touch sensor devices. For more information on the Object Protocol and its implementation on the mXT224E, refer to the *mXT224E Protocol Guide*.

5. Getting Started With the mXT224E

5.1 Communication with the Host

The mXT224E uses an I^{2}C-compatible bus to communicate with the host. See Section 4.4 on page 16 and Appendix C on page 45 for more information.

5.2 Establishing Contact

On power-up, the $\overline{\text{CHG}}$ line goes low to indicate that there is new data to be read from the Message Processor object. If the $\overline{\text{CHG}}$ line does not go low, there is a problem with the device.

The host should attempt to read any available messages to establish that the device is present and running following power-up or a reset. Examples of messages include reset or calibration messages. The host should also check that there are no configuration errors reported.

5.3 Using the Object Protocol

The mXT224E has an object-based protocol that is used to communicate with the device. Typical communication includes configuring the device, sending commands to the device, and receiving messages from the device. See Section 4.11 on page 28 and refer also to the *mXT224E Protocol Guide*.

The host must perform the following initialization so that it can communicate with the mXT224E:

1. Read the start positions and sizes of all the objects in the mXT224E from the Object Table and build up a list of these addresses.
2. Use the Object Table to calculate the report IDs so that messages from the device can be correctly interpreted.

5.4 Writing to the Device

See Section 4.4.3 on page 17 for information on the format of the I^{2}C-compatible write operation.

To communicate with the mXT224E, you write to the appropriate object:

- To send a command to the device, you write the appropriate command to the Command Processor object (for example, to send a reset, backup or calibrate command). Refer to the *mXT224E Protocol Guide* for the full list of available commands.
- To configure the device, you write to an object. For example, to configure the device's power consumption you write to the global Power Configuration object, and to set up a touchscreen you write to a Multiple Touch Touchscreen object. Some objects are optional and need to be enabled before use.

Refer to the *mXT224E Protocol Guide* for more information on the objects.

5.5 Reading from the Device

See Section 4.4.5 on page 17 for information on the format of the I^{2}C-compatible read operation.

Status information is stored in the Message Processor object. This object must be read to receive any status information from the mXT224E. The $\overline{\text{CHG}}$ line is asserted whenever a new message is available in the Message Processor object. This provides an interrupt-style interface. The host should always use the $\overline{\text{CHG}}$ line to be notified of messages. The host should not poll the device for messages. See Section 4.4.7 on page 20 for more information on using the $\overline{\text{CHG}}$ line.

5.6 Configuring the Device

The objects are designed such that a default value of zero in their fields is a "safe" value that typically disables functionality. The objects must be configured before use and the settings written to the nonvolatile memory using the Command Processor object. Refer to the *mXT224E Protocol Guide* for more information.

The following objects must be configured before use:

- Power Configuration
  - Set up the Idle Acquisition Interval, Active Acquisition Interval and Active to Idle Timeout.
- Acquisition Configuration

The following objects should also be configured and enabled, as required:

- Touch objects: Multiple Touch Touchscreen, Key Array, Proximity Sensor
  - Enable the object.
  - Configure the origin and the number of channels it occupies. Configure the other fields in the object, as required. For example, set up the AKS group(s), specify the burst length and threshold.
  - Enable reporting to receive touch messages from the object.
- Signal processing objects: Noise Suppression (T48), Grip Suppression, Touch Suppression (T42), Stylus (T47)
  - Enable the object.
  - Configure the fields in the object, as required.
  - Enable reporting to receive signal processing messages from the object.
- Support objects: Communications Configuration, GPIO/PWM Configuration, CTE Configuration (T46), Self Test, User Data
  - Enable the object, if the object requires it.
  - Configure the fields in the object, as required.
  - Enable reporting, if the object supports messages, to receive messages from the object.

Refer to the *mXT224E Protocol Guide* for more information on configuring the objects.

6. Specifications

6.1 Absolute Maximum Specifications

Vdd	3.6V
AVdd	3.6V
Max continuous pin current, any control or drive pin	20 mA
Voltage forced onto any pin	-0.5V to (Vdd or AVdd)+0.5V
Configuration parameters maximum writes	10,000

⚠️ **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

Operating temp	-40°C to +85°C
Storage temp	-65°C to +150°C
Vdd	1.8V ±5%
AVdd	2.7V ±5%
Vdd vs AVdd power sequencing	No sequencing required
Supply ripple + noise	See Section 6.10 on page 35
Cx transverse load capacitance per channel	0.32 pF to 5 pF
GPO current	<5 mA
Temperature slew rate	10°C per minute

6.3 DC Specifications

6.3.1 Digital Power (Vdd)

Parameter	Description	Min	Typ	Max	Units	Notes
Vdd	Operating limits	1.71		3.47	V	

Vdd = 1.8V, Ta = recommended range, unless otherwise noted

Parameter	Description	Min	Typ	Max	Units	Notes
Idd	Average supply current, 80Hz, 1 touch, XSIZE=18, CHRGTIME=2 μs		1.13		mA	1.8V; see Section 6.10
	Sleep		1		μA	1.8V; see Section 6.10
Vil	Low input logic level	-0.5		0.3Vdd	V	1.8V <Vdd <3.3V
Vih	High input logic level	0.7Vdd		Vdd+0.5	V	1.8V <Vdd <3.3V
Vol	Low output voltage			0.2Vdd	V	I_{OL} = 10 mA; 1.8V <Vdd <3.3V
Voh	High output voltage	0.8Vdd			V	I_{OH} = -20 mA; 1.8V <Vdd <3.3V
Iil	Input leakage current		< 0.001	1	μA	
Slew rate	Minimum slew rate	1			V/100 μs	

6.3.2 Analog Power (AVdd)

Parameter	Description	Min	Typ	Max	Units	Notes
AVdd	Operating limits	2.57		3.47	V	

Note: **AVdd must be stable and have a nominal tolerance in the host system of ±5% or better.**

AVdd = 2.7V, Ta = recommended range, unless otherwise noted

Parameter	Description	Min	Typ	Max	Units	Notes
AIdd	Average supply current, 80Hz, 1 touch, XSIZE = 18, CHRGTIME = 2 µs		1.25		mA	2.7V; see Section 6.10
	Sleep		4.5		nA	2.7V; see Section 6.10
Slew rate	Minimum slew rate	1			V/100 µs	

6.4 Timing Specifications

Touches = 1, XSIZE = 18, CHRGTIME = 2 µs

Parameter	Description	Min	Typ	Max	Units	Notes
Tlatency	80Hz	10	16.25	22.5	ms	
	100Hz	10	15.45	20.9	ms	
	200Hz	10	12.95	15.9	ms	

6.5 Reset Timings

Parameter	Min	Typ	Max	Units	Notes
Power on to $\overline{\text{CHG}}$ line low		21.7		ms	
Hardware reset to $\overline{\text{CHG}}$ line low		21.6		ms	
Software reset to $\overline{\text{CHG}}$ line low		22		ms	

6.6 I^{2}C-compatible Bus Specifications

Parameter	Operation
Addresses	0x4A or 0x4B
Maximum bus speed (SCL)	400 kHz
I^{2}C specification	Version 2.1

6.7   Power Consumption

Digital Power
1.8V Supply Current, XSIZE = 18, YSIZE = 12,
CHRGTIME = 24 (2 µs), IDLESYNCSPERX/ACTVSYNCSPERX = 8

Analog Power
2.7V Supply Current, XSIZE = 18, YSIZE = 12,
CHRGTIME = 24 (2 µs),IDLESYNCSPERX/ACTVSYNCSPERX = 8

6.8 Speed

Acquisition Rate vs Number of Moving Touches
1.8V Supply Current, CHRGTIME = 24 (2 µs),
IDLEACQIN/ACTVACQINT = 255 (Free Run)
IDLESYNCSPERX/ACTVSYNCSPERX = 8
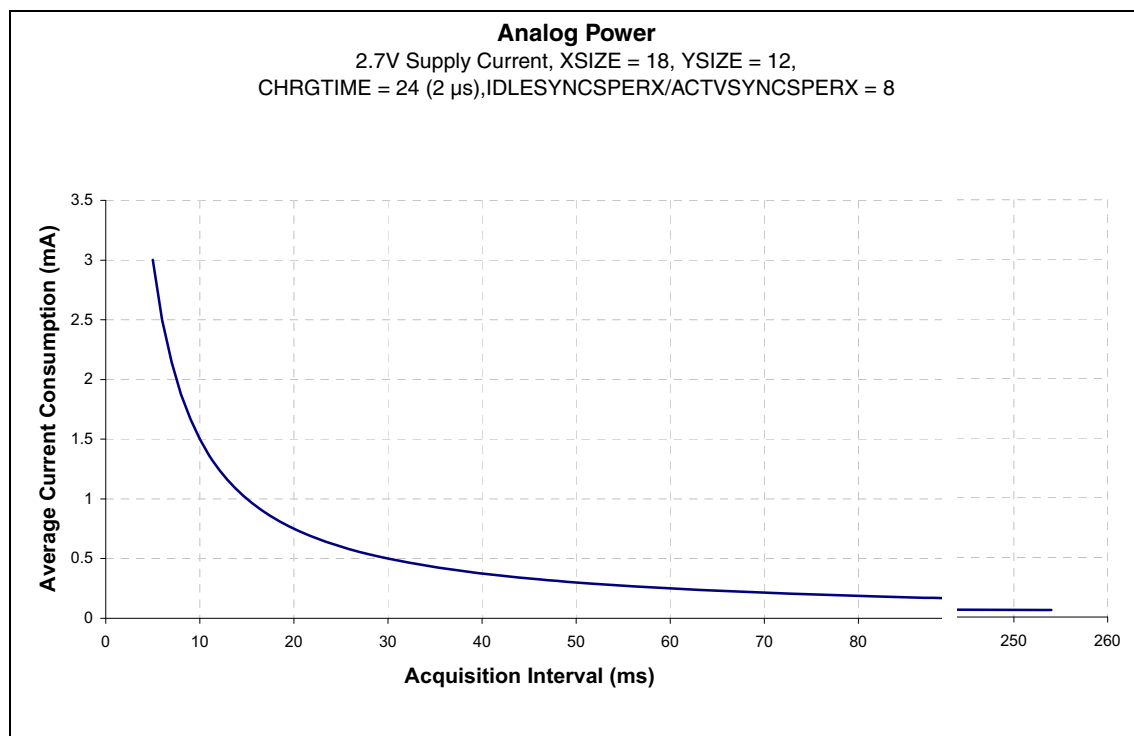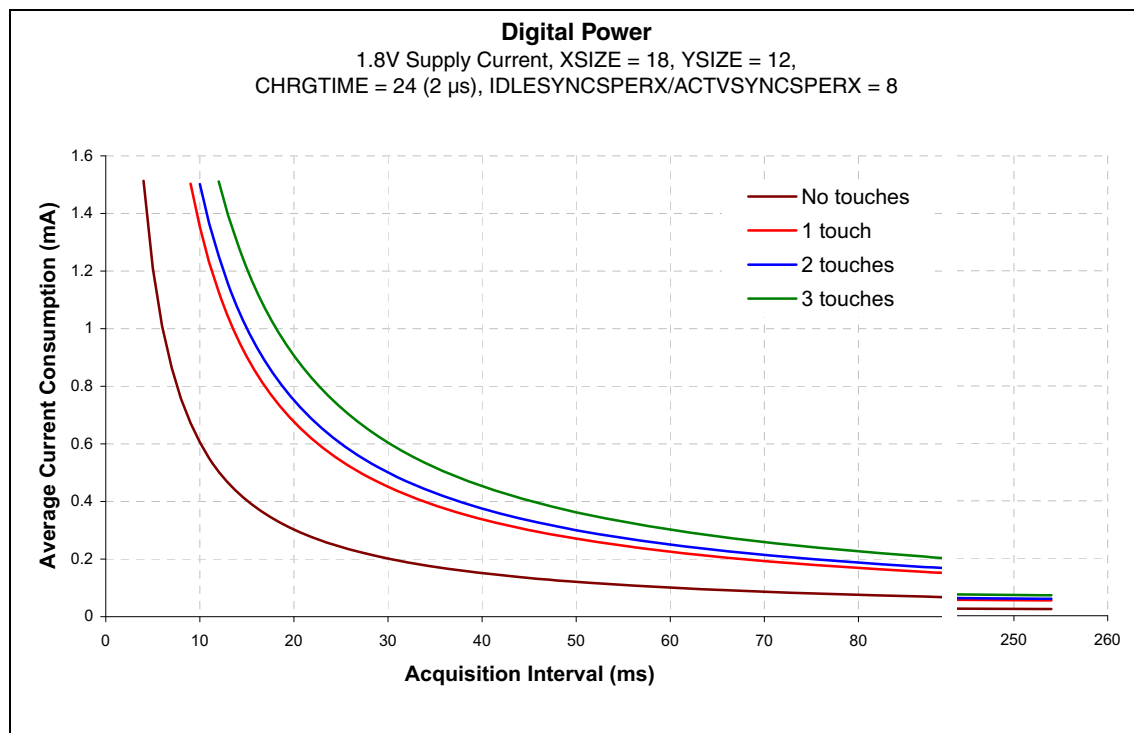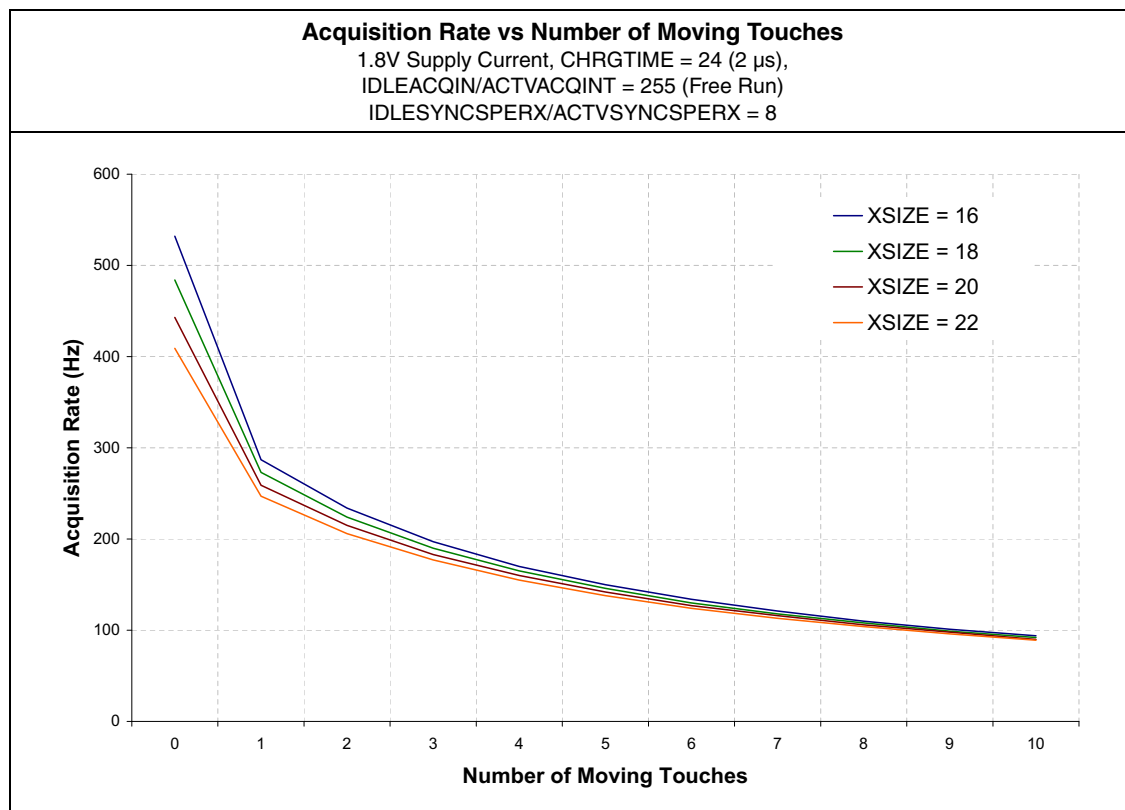
6.9 Touch Accuracy and Repeatability

Touchscreen pitch= 4.7 mm, front panel = 1 mm, touch size = 8 mm

Parameter	Min	Typ	Max	Units	Notes
Linearity		±0.5		mm	
Accuracy		±1		mm	
Accuracy edge		±2		mm	
Repeatability		±0.25		%	X axis with 12-bit resolution

6.10 Power Supply Ripple and Noise

See Section 6.10.1 on page 36 and Section 6.10.2 on page 37 for the power supply ripple and noise characterization charts.

The test circuit used for the charts is shown in Figure 6-1.

Figure 6-1. Circuit Used for Power Supply Ripple Characterization Charts
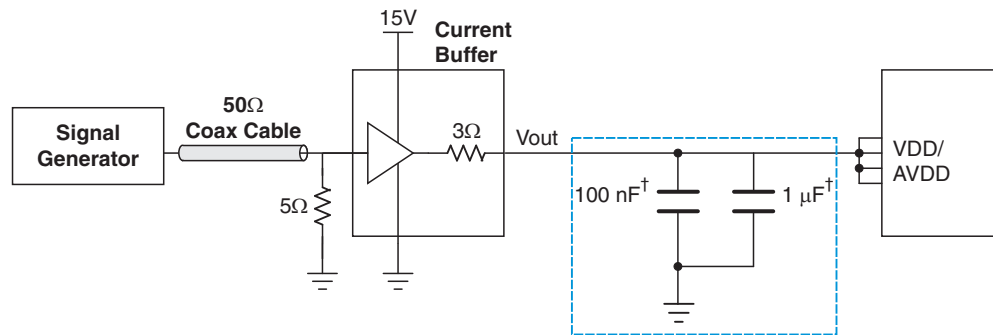
Note: Bypass capacitors are <5 mm away from the chip.

6.10.1 Digital Power (Vdd) Characterization

Vdd Injected Noise
Noise Suppression Disabled

Vdd Injected Noise
Noise Suppression Enabled
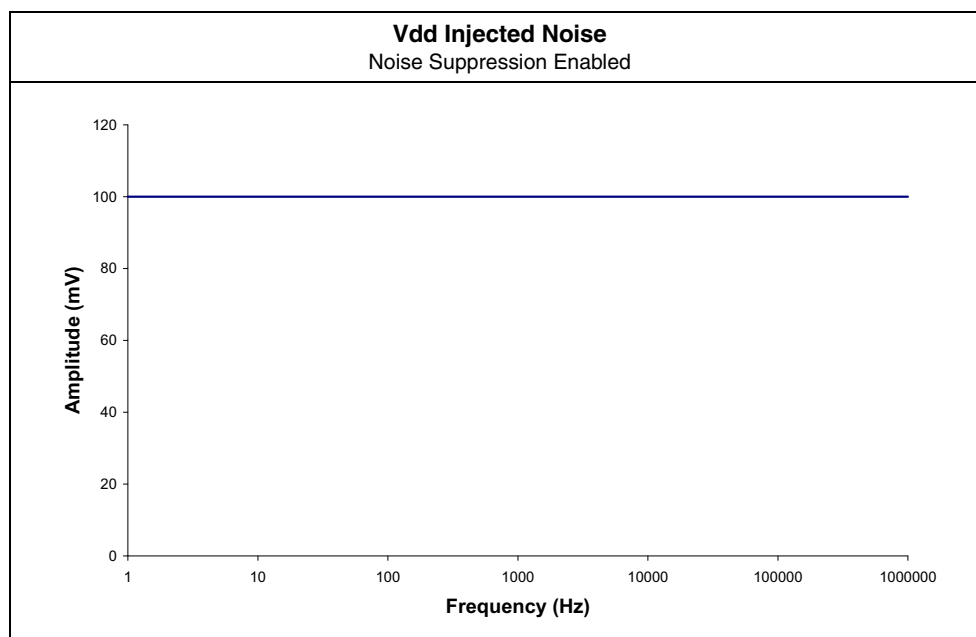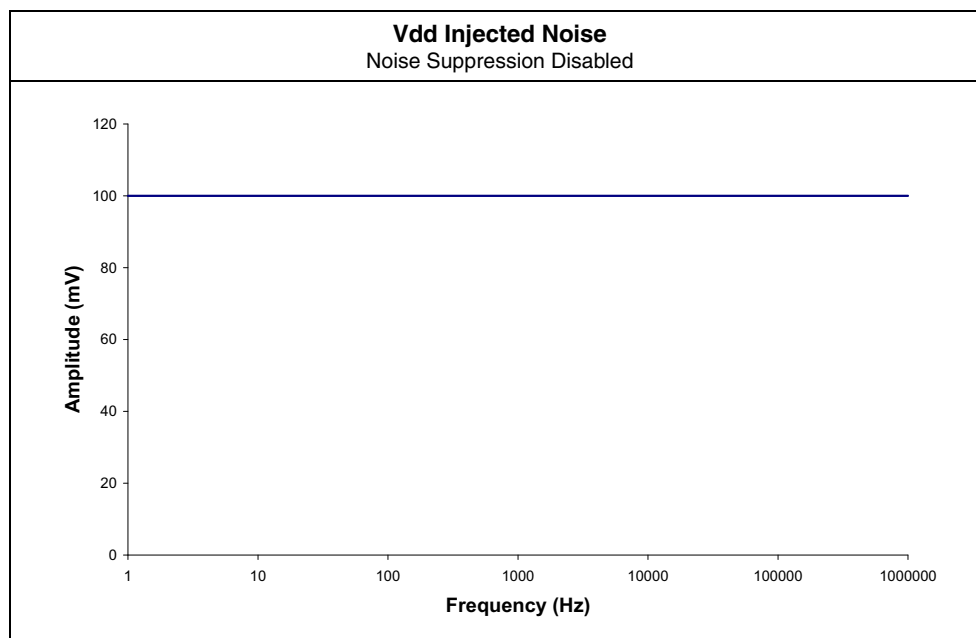
6.10.2   Analog Power (AVdd) Characterization

AVdd Injected Noise
Noise Suppression Disabled

AVdd Injected Noise
Noise Suppression Enabled
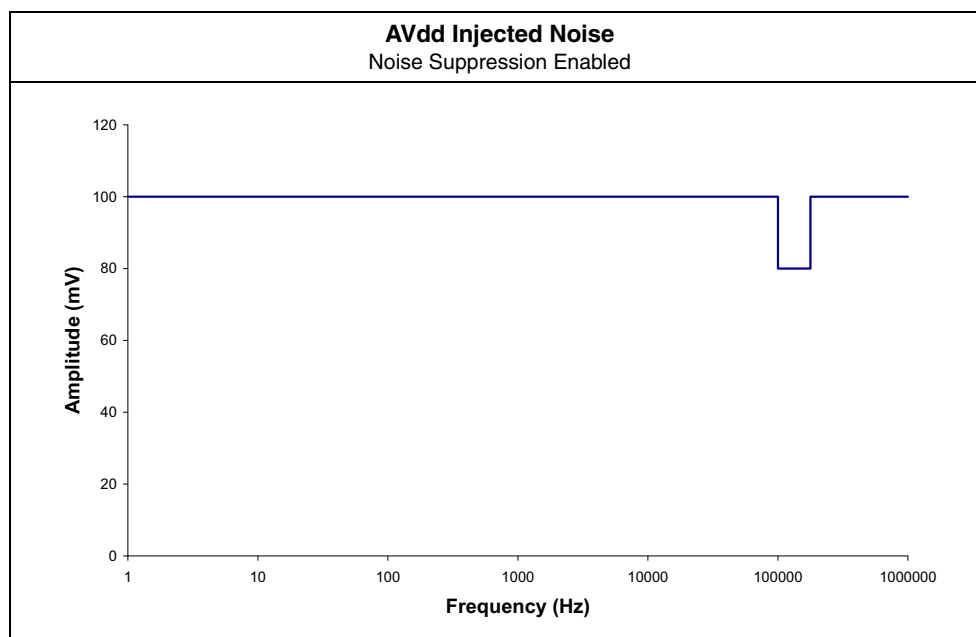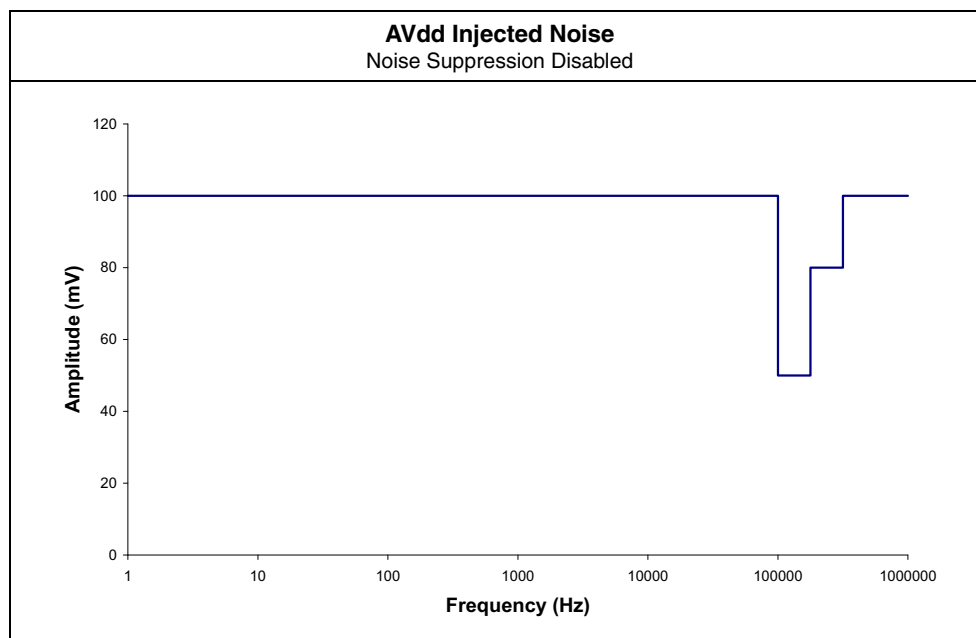
6.11 Mechanical Dimensions

6.11.1 49-ball UFBGA (ATMXT224E-CCUR)

TOP VIEW

SIDE VIEW

49-Øb

BOTTOM VIEW

A1 BALL CORNER

COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	0.60	
A1	0.05	0.10	0.15	
A2	0.43 REF			
b	0.25	0.30	0.35	1
b1	0.25	–	–	2
D	4.90	5.00	5.10	
D1	3.90 BSC			
E	4.90	5.00	5.10	
E1	3.90 BSC			
e	0.65 BSC			
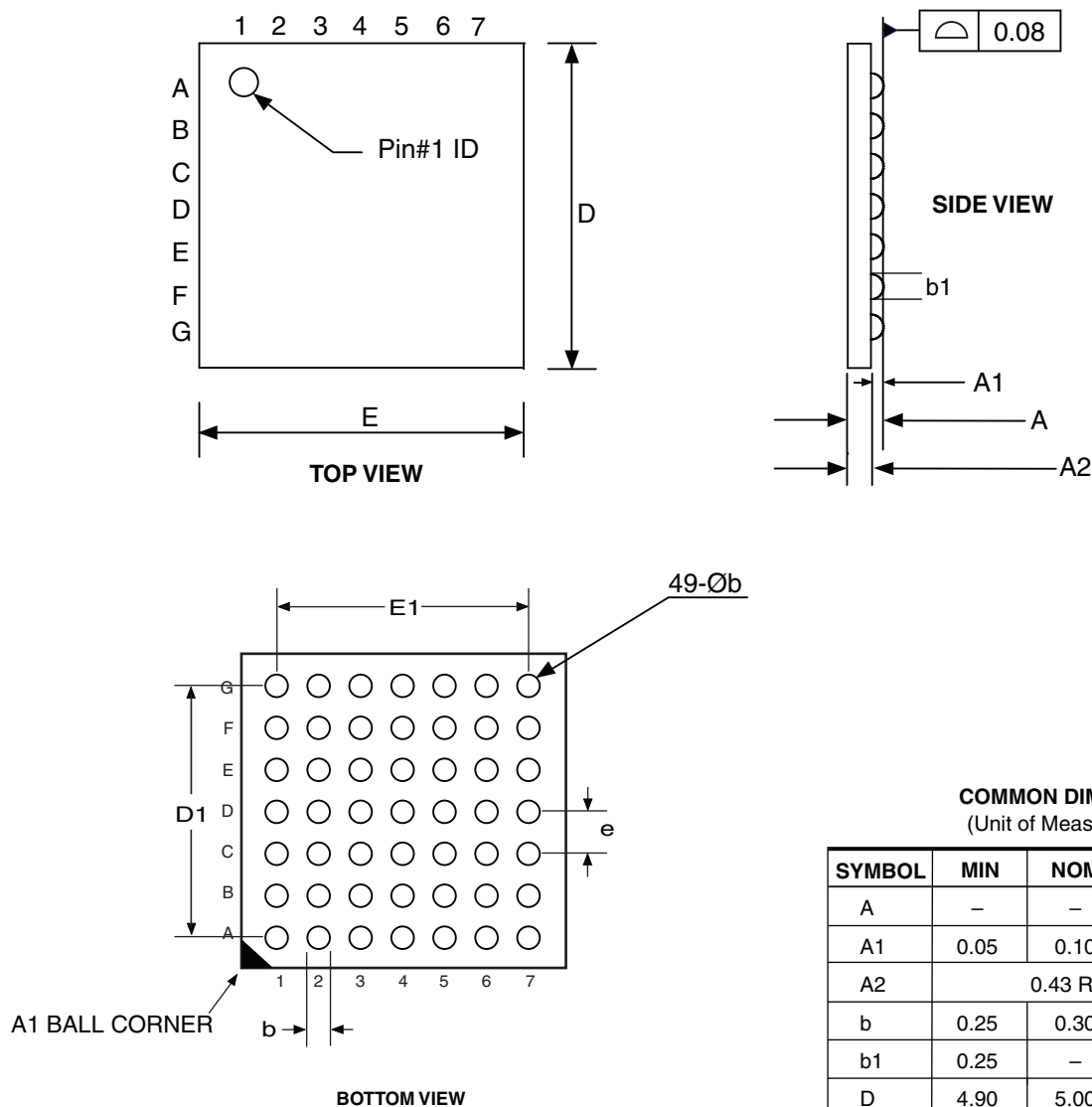
Note 1: Dimension "b" is measured at the maximum ball dia. in a plane parallel to the seating plane.

Note 2: Dimension "b1" is the solderable surface defined by the opening of the solder resist layer.

08/03/09

	TITLE	GPC	DRAWING NO.	REV.
Package Drawing Contact: packagedrawings@atmel.com	**49CU1,** 49-ball (7 x 7 Array), 5 x 5 x 0.6 mm body, lead pitch 0.65 mm, Ultra Thin, Fine-Pitch bump Ball Grid Array Package (UFBGA)	CDG	49CU1	A

6.11.2   48-pin QFN (ATMXT224E-MAHR)

TOP VIEW

SIDE VIEW

SEATING PLANE

BOTTOM VIEW

Note1: Refer to JEDEC Drawing MO-248, variation UHHE-1 (saw singulation).

Note2: Dimension "b" refers to metalized terminal and is measured
between 0.15 and 0.30mm from the terminal tip. If the terminal has
the optional radius on the other end of the terminal, the dimension
should not be measured in that radius area.
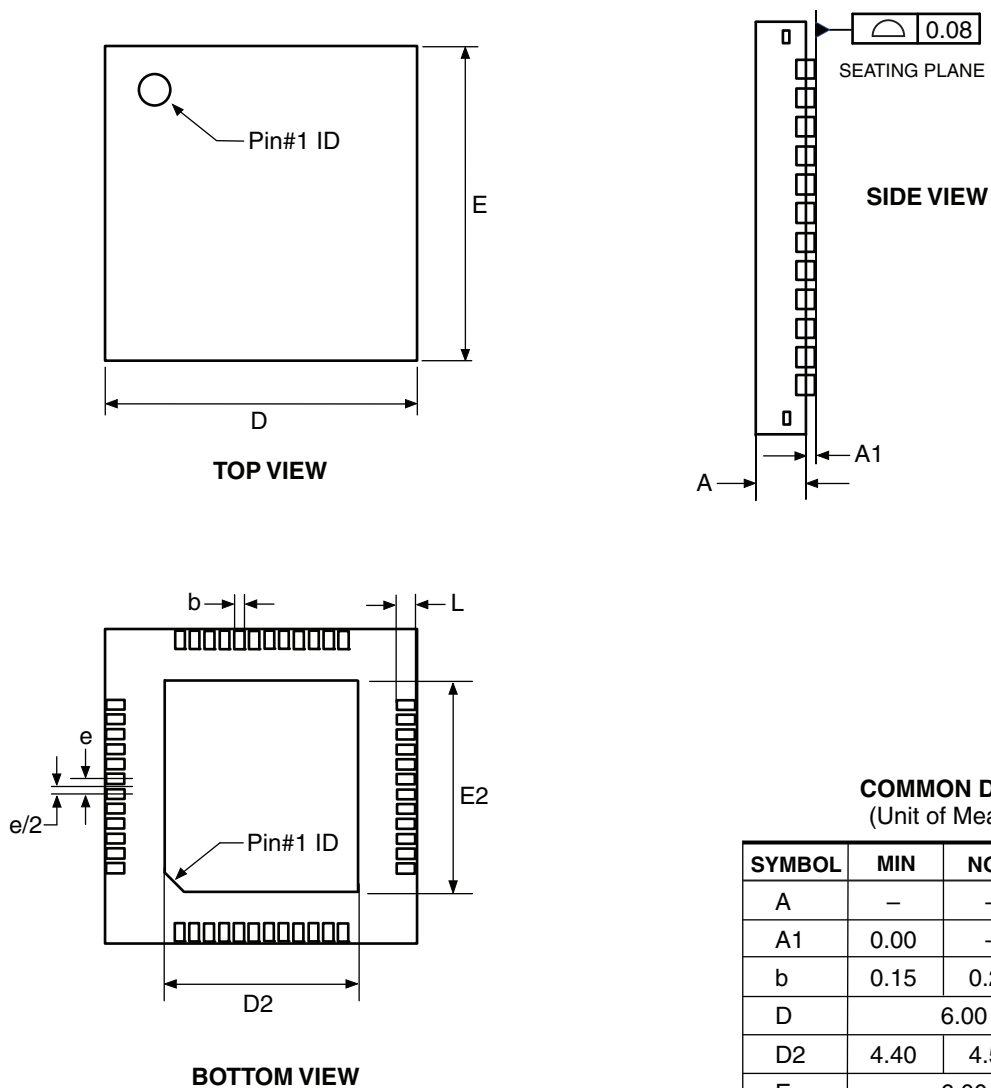
COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	0.60	
A1	0.00	–	0.05	
b	0.15	0.20	0.25	2
D	6.00 BSC			
D2	4.40	4.50	4.60	
E	6.00 BSC			
E2	4.40	4.50	4.60	
e	–	0.40	–	
L	0.35	0.40	0.45	

30/06/09

	TITLE	**GPC**	**DRAWING NO.**	**REV.**
Package Drawing Contact: packagedrawings@atmel.com	**48MA1,** 48 pad, 6 x 6 x 0.6mm body, 0.40mm pitch,4.5 x 4.5mm exposed pad, Saw singulated Thermally enhanced plastic Ultra thin quad flat no lead package (UQFN).	ZAL	48MA1	A

6.12 Marking

6.12.1 49-ball UFBGA (ATMXT224E-CCUR)

Ball A1 ID →

ATMEL
MXT224E

Abbreviation of
Part Number →

CCU-1R0 ← Code Revision 1.0
(Released)

Lot Code (variable text) →

LOTCODE

6.12.2 48-pin QFN (ATMXT224E-MAHR)

Pin 1 ID →

ATMEL
MXT224E

Abbreviation of
Part Number →

MAH-1R0 ← Code Revision 1.0
(Released)

Lot Code (variable text) →

LOTCODE
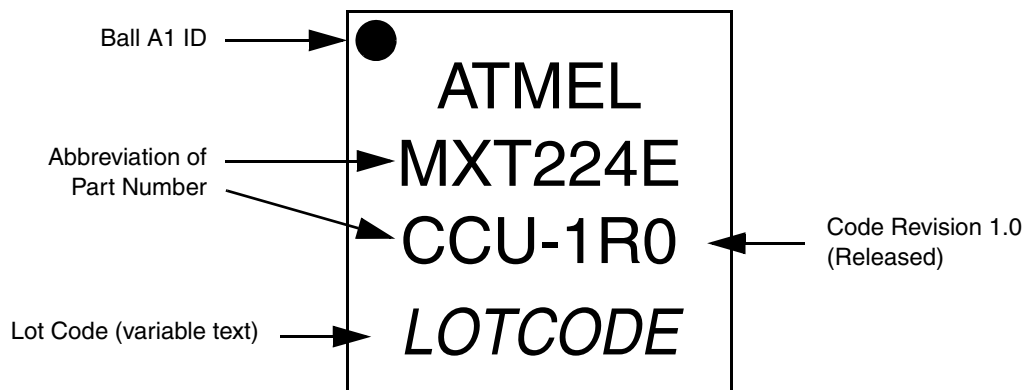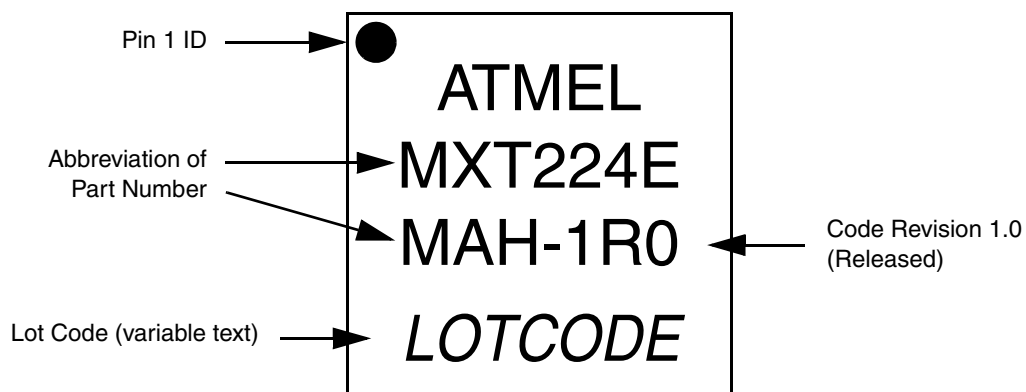
6.13 Part Numbers

Part Number	Description
ATMXT224E-CCUR (tape and reels)	49-ball 5 x 5 mm UFBGA RoHS compliant
ATMXT224E-MAHR (tape and reels)	48-pin 6 x 6 mm QFN RoHS compliant

6.14 Moisture Sensitivity Level (MSL)

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

Appendix A.   Glossary of Terms

Channel

One of the capacitive measurement points at which the sensor controller can detect capacitive change.

Jitter

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, the effects of linearity are ignored.

Linearity

The 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.

Resolution

The 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 resolution of 10 bits can resolve a movement of 0.0977 mm on a slider 100 mm long. Jitter in the reported position degrades usable resolution.

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 12-bits each, ranging from 0 to 4095, representing the extents of the touchscreen active region.

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 (that is, above the panel). Some proportion of the field does escape the surface of the dielectric, however, 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.

B.2   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.

---

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.

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.3   Interference Sources

B.3.1   Power Supply

See 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.

The device itself uses the AVdd power supply as an analog reference, so 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 AVdd to fluctuate enough to cause false detection or sensitivity shifts. The digital Vdd supply is far more tolerant to noise.

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

Noise on AVdd 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 AVdd and 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; both must be X7R dielectric capacitors.

B.3.2   LCD Drive

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

B.3.3   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.3.4   Other Noise Sources

Refer to the *Touch Sensors Design Guide* (downloadable from the Touch Technology area of Atmel's website) for information.

Appendix C. I²C Basics (I²C-compatible Operation)

C.1   Interface Bus

The device communicates with the host over an I²C-compatible bus, in accordance with version 2.1 of the I²C specification. 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
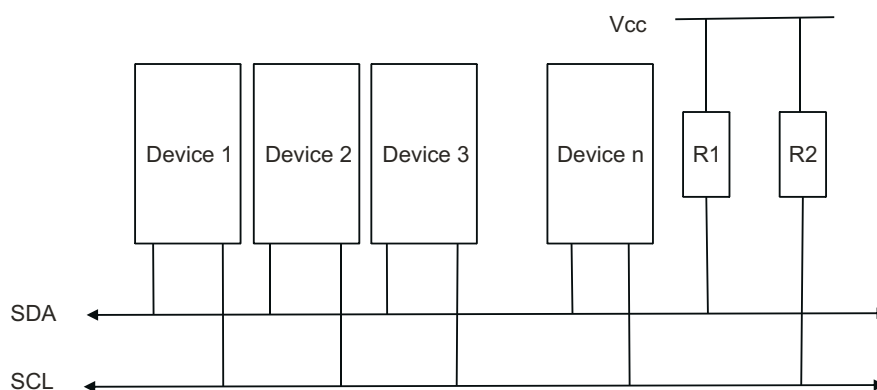
C.2   Transferring Data Bits

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
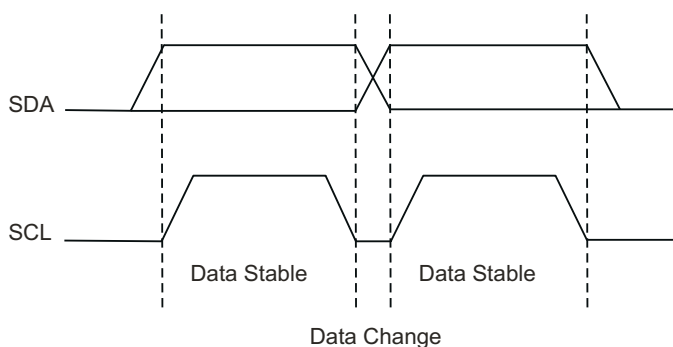
C.3 START and STOP Conditions

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 on page 46, 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
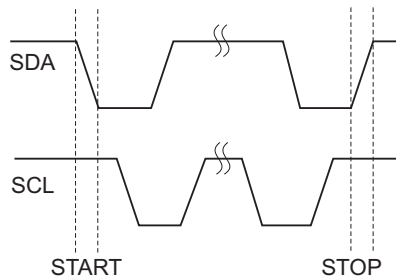
C.4 Address Byte Format

All address bytes are 9 bits long. They consist 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. An address byte consisting of a slave address and a READ or a WRITE bit is called SLA+R or SLA+W, respectively. When the device recognizes that it is being addressed, it acknowledges by pulling SDA low in the ninth SCL (ACK) cycle.

The most significant bit of the address byte is transmitted first.

Figure C-4. Address Byte Format
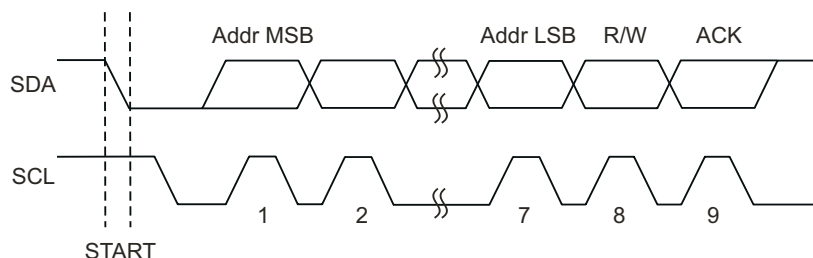
C.5 Data Byte Format

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. The slave device is responsible for acknowledging the reception. An acknowledge (ACK) is signaled by the slave device pulling the SDA line low during the ninth SCL cycle. If the slave device 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 or SLA+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.

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

Figure C-6. Byte Transmission

Revision History

Revision Number	History
Revision AX – January 2011	Initial release for chip revision 0.3
Revision BX – March 2011	Updated for chip revision 0.4 • Section 4.1 (Powerup/Reset): minor update • Section 4.3 (Sensor Acquisition) added • Section 4.4 (Communications): Descriptions of I^{2}C-compatible communications and $\overline{\text{CHG}}$ line updated
Revision CX – April 2011	Updated for chip revision 1.0 • Section 1 (Pinouts and schematics): GPIO1 now shared with CHRG_IN pin. • Section 4.1 (Powerup/Reset): updated with specifications • Section 6 (Specifications) updated

Notes

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

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

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

Product Contact

Web Site
www.atmel.com

Technical Support
touch@atmel.com

Sales Contact
www.atmel.com/contacts

Literature Requests
www.atmel.com/literature