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

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

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

Contact us

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

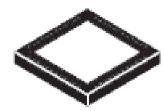
Freescale Semiconductor
Data Sheet: Product Preview

Document Number: CRTOUCHDS
Rev. 3, 04/2013

CRTouch

CRTouch Data Sheet

Capacitive and Resistive Touch Sensing Application Specific IC.



32-QFN
Case 1582
5 mm × 5 mm

The CRTouch is a Ready Play solution device designed to serve as a four wire and five wire resistive touch driver and a capacitive touch sensing device in a single 32-pin QFN package. It provides all the signal interfacing and processing to calculate X and Y coordinates as well as to detect and process two touch gestures for human-machine interface applications. It's key features are:

Features

- X and Y coordinates calculated from a resistive touch screen with built-in filter to improve stability
- Slide gesture detection for single touch.
- Two touch gesture detection for resistive four wire screens
  - Zoom In and Zoom Out
  - Rotate with clockwise and counter clockwise indication.
- Four capacitive keys in different configurations
  - Rotary
  - Slider
  - Keypad
- UART communication and I2C communication available
- Baud-rate auto detection pin to enable automatic synchronization with any UART baud-rate between 9600 and 115200 bps using the same UART RX pin
- Configurable scanning period that can calculate up to 200 coordinate points per second
- 1.8 to 3.6 volts operation
- 32-pin QFN package
- –40 °C to 105 °C operating temperature
- Normal Run mode, Sleep mode, and Shutdown mode for lower power consumption operation

Table of Contents

1 Pins and Connections

1.1 Device block diagram

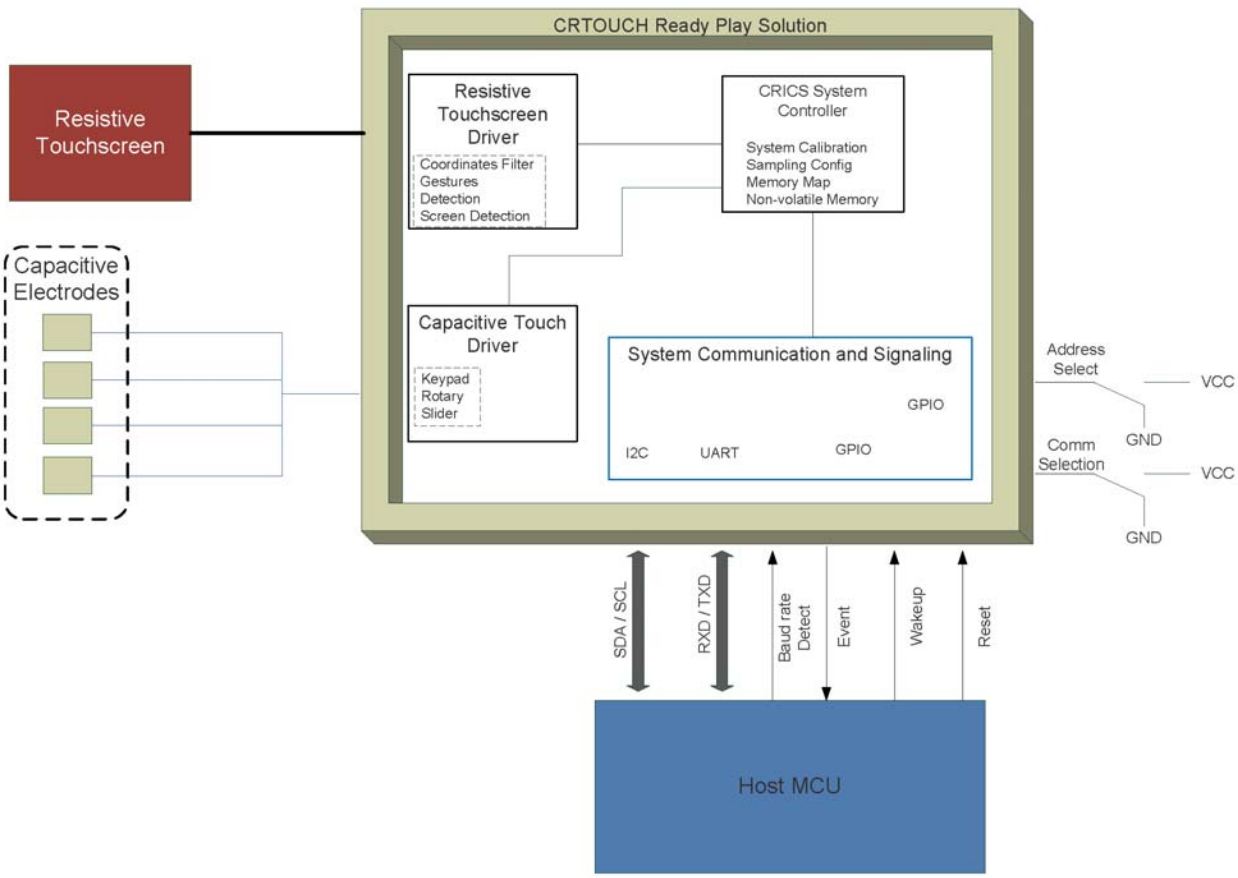


Figure 1. Block diagram

1.2 Device pin out

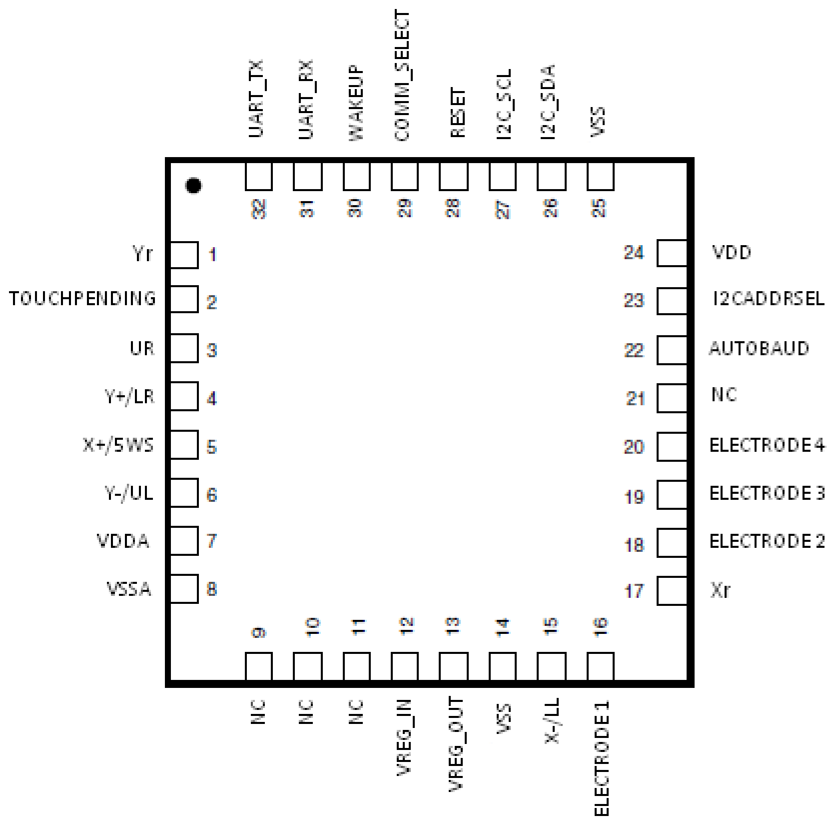


Figure 2. Device pinout

1.3 Recommended system connections

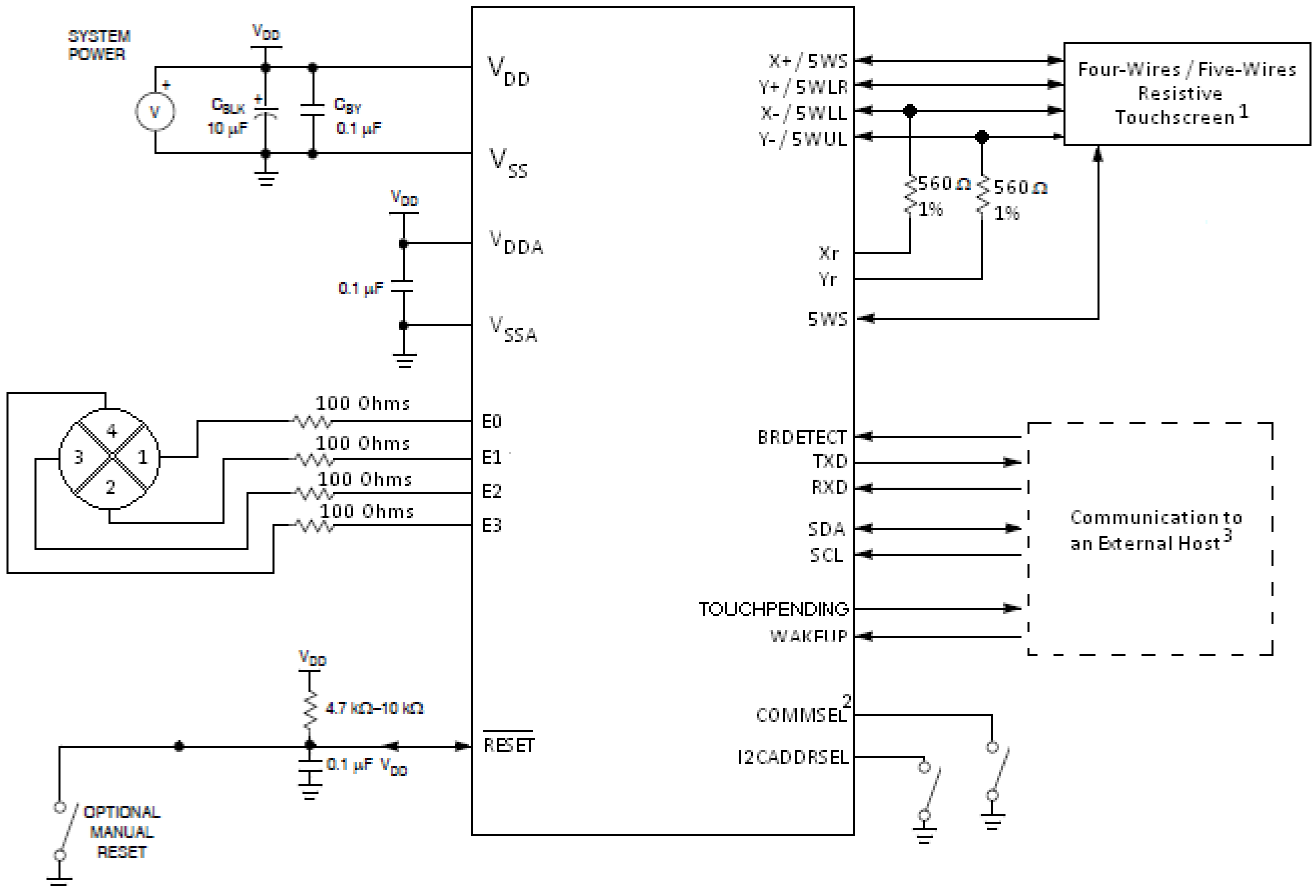


Figure 3. Recommended system connections

NOTE

1. Resistive touch screen signals are bi-directional only for four-wire screens. For five-wire 5WS is input and the other four signals are outputs.
2. COMMSEL is connected to ground only if UART communication is desired. Otherwise it can be left unconnected.
3. I2C and UART communications are mutually exclusive. Only one can work during the device run time. The other pins from the other communication protocol can be un-connected.
4. All unused pins can be left as Not Connected. Connecting them to ground may increase power consumption.

1.4 Signals description

Table 1. Signal descriptions

Pin	Function used for CRTouch	Description
1	Yr	Y voltage reference
2	TouchPending	Indicates that there is a previous touch event (resistive or capacitive) fwhere data registers have not been read
3	5WUR	5-Wire Upper Right Connection
4	Y+ / 5WLR	4-Wire Y+ or 5-Wire Lower Right connection
5	X+ / 5WS	4-Wire X+ or 5-Wire Sense connection
6	Y– / 5WUL	4-Wire Y– or 5-Wire Upper Left connection
7	VDDA	V_{DD} Signal for the ADC
8	VSSA	V_{SS} Signal for the ADC
9-11	NC	No Connect
12	VREG_IN	Voltage regulator Input
13	V33_OUT	Voltage regulator output –3.3 Volts
14	VSS	V_{SS} – Connect to Digital Ground
15	X– / 5WLL	4-Wire X– or 5-Wire Lower Left connection
16	Electrode1	Capacitive Touch Sensing Key #1
17	Xr	X voltage reference
18	Electrode2	Capacitive Touch Sensing Key #2
19	Electrode3	Capacitive Touch Sensing Key #3
20	Electrode4	Capacitive Touch Sensing Key #4
21	NC	No Connect
22	Baudrate Auto Detect	Signal used to enable Baud Rate auto detection
23	I2CAddrSel	I2C Address Selection pin is used as bit 1 of the I2C address
24	VDD	V_{DD} – Connect to system regulated power supply
25	VSS	V_{SS} – Connect to digital Ground
26	I2C1_SDA	I2C – Bidirectional SDA communication signal
27	I2C1_SCL	I2C – Clock signal. Must be driven by the bus master
28	Reset	Reset signal
29	Com Select	Communication Select Pin. 0 = UART. 1 = I2C
30	Wakeup	Signal used to wake the CRTouch when configured to go into Sleep or Shutdown modes. While in Sleep mode, this pin should be held down with a logical 0 to communicate with the device.
31	UART_RX	UART Reception pin
32	UART_TX	UART Transmission pin

2 Functional Description

The capacitive and resistive touch (CRTouch) is a device capable of interfacing with 4-wire and 5-wire resistive touch screens. It is capable of detecting single touch slide and two-touch rotate and pinch gestures. CRTouch includes a calibration procedure to increase the touch detection accuracy and configurable low power and shutdown modes to reduce power consumption.

2.1 CRTouch resistive touchscreen scanning

2.1.1 Touchscreen electrical signals

The types of resistive touchscreen supported by this device are passive elements composed of two layers of conductive material uniformly distributed across the screen . Due to the resistivity of the conductive material, it can be seen as a resistor between the terminals of each layer. The following figure shows the construction of the 4 and 5 wire types of screens, and their simplified electrical equivalent circuit.

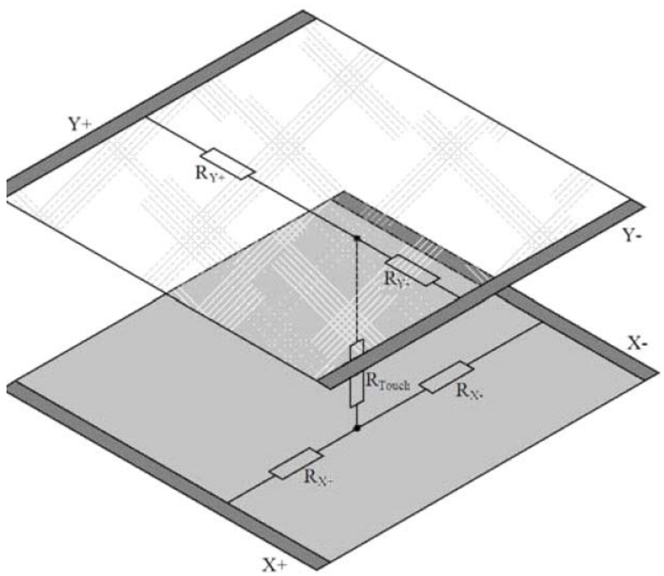


Figure 4. Construction and electrical equivalent of a four wire touchscreen

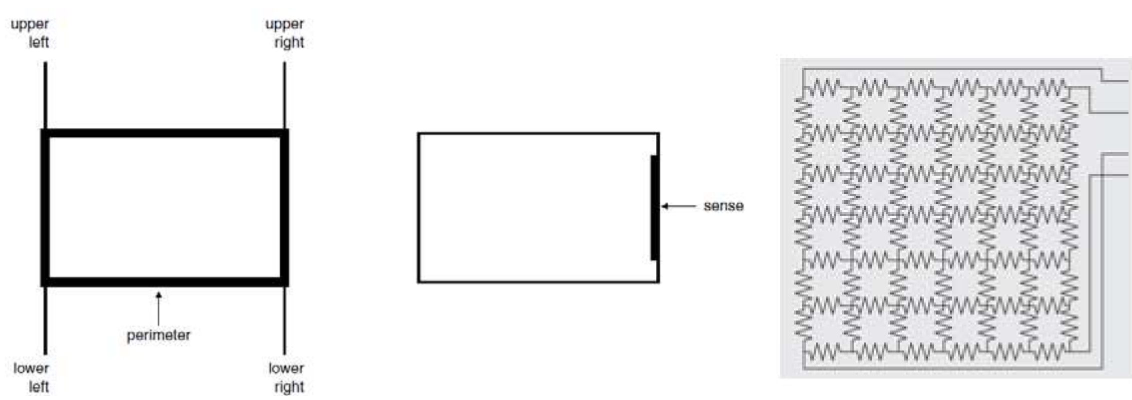


Figure 5. Construction and electrical equivalent of a five wire touchscreen

When a voltage is applied at the terminals of one layer, the voltage is linearly distributed across the conductive material. For a four wire screen, the coordinates of the point of contact can be determined by applying voltage to one layer and reading the voltage on the other. For a five wire screen, one layer is used only for reading the voltage, while the other has voltages applied in different combinations for determining the value of each axis. The following table shows how the signals are activated for X and Y measurements in each type of screen.

Table 2. Signal states for a four wire screen

Measurement	X+	X–	Y+	Y-
X	V_{dd}	V_{ss}	Z	Z
Y	z	Z	V_{dd}	V_{ss}

Table 3. Signal states for a five wire screen

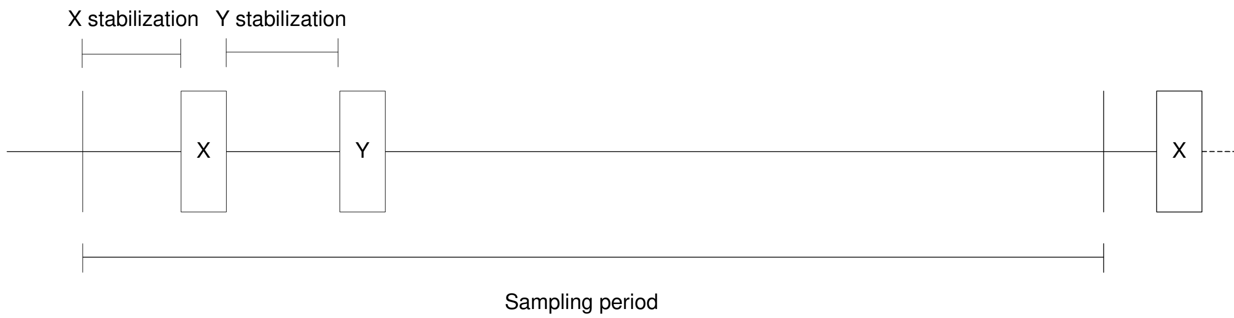
Measurement	UR	UL	LR	LL
X	V_{dd}	V_{ss}	V_{dd}	V_{ss}
Y	V_{dd}	V_{dd}	V_{ss}	V_{ss}

Considering these scanning sequences, the screen's origin (coordinate 0,0) is located in the intersection of the X– and Y– terminals for a four wires screen, and at the LL terminal for a five wire screen. It is also important to consider that the connection between the conductive material of the screen and the terminals of the CRTouch controller represent a series resistance with the screen. Because of this, the coordinates obtained at the edges of the active region of the screen may be a number of counts above 0 and a number of counts below the maximum value. This variation is given by the construction of the touchscreen, the impedance of its connection lines, the size of the active area, and so on.

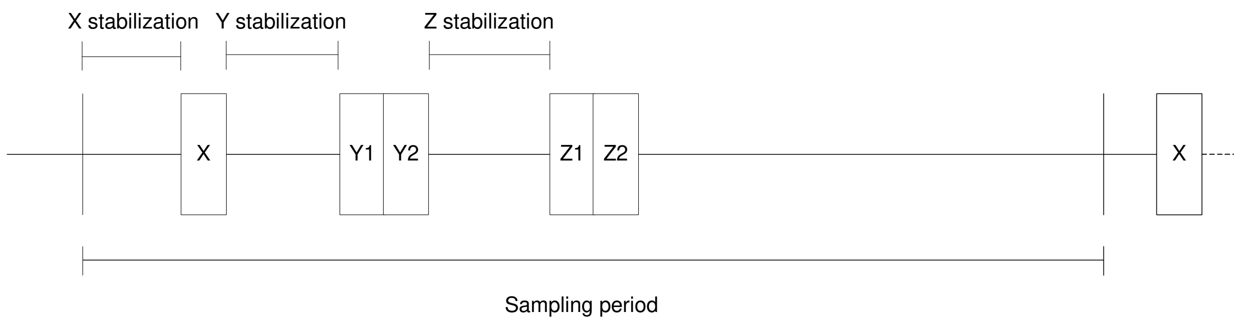
Another electrical phenomenon that needs to be considered is the capacitances created in the screen. Because the active area is composed of two plates of conductive material, this creates a set of capacitances between the two layers. When combined with the resistors created in the screen, these form an RC circuit with an associated stabilization time constant. If the voltage of one layer is read before it stabilizes, then a deviation from the real coordinate is produced. In some devices ESD diodes may be connected to the resistive panel for improved electrostatic immunity. In this case, low capacitance ESD diodes must be used to avoid increasing the resulting capacitance.

2.1.2 CRTouch scanning process

The CRTouch has three scanning modes for the resistive screen. One is active when only X and Y coordinate values are desired, the second activates when the pressure measurement is enabled, and the last one when any of the two touch gestures are enabled. Before each sample is taken, the CRTouch waits for a configurable amount of time for the signals to stabilize. The following figure shows the differences in the scanning process for each scanning mode.



Scanning Process for X and Y coordinates only



Scanning Process for X ,Y, and Pressure

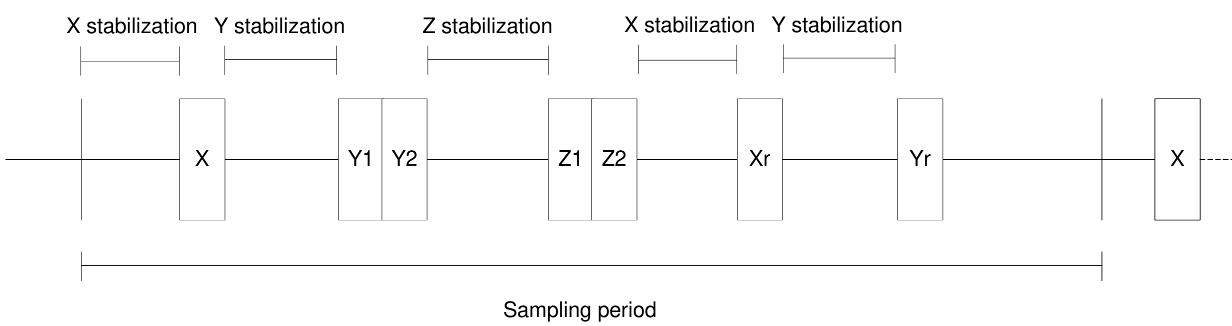


Figure 6. Scanning process for X, Y, pressure, and two touch gestures

As seen in the previous figure, the scanning sequence repeats at a fixed period of time. This period is configurable in the Sampling Time register and may be in the range of 5 ms to 100 ms. The CRTouch scans the screen signals only when a touch is detected. When the screen is not touched, the device stays in a standby state.

At the end of the scanning process, a set of coordinates is produced. These coordinates have a default resolution of 12 bits, or 4096 points per axis. However, this resolution may be modified to reflect the proportions of the screen more accurately (typical screens have a 4:3 or 16:9 proportion). This can be done by writing to the horizontal resolution and vertical resolution, which correspond to the X and Y resolution respectively. The resulting X and Y values are scaled to fit into this resolution.

The CRTouch is capable of autodetecting the type of the screen connected. The type of screen detected is reflected in the Status Register 2. Only two touch gestures and pressure detection is available for 4 wire screens, and cannot be enabled when a 5 wire screen is connected.

2.1.3 Resistive events

The CRTouch has a TouchPending signal that can assert on configurable events in the system to alert the host of a new event that requires attention. This signal can be used by the host to avoid time based polling of the status of the device and therefore unnecessary use of the communication bus. Whenever an event is detected in the system, it will be reflected in its associated bit in Status Register 1. All event bits of this register remain latched and are cleared until the register is read. This avoids losing any detected event if the host cannot read the status of the device for several sampling periods. Each event can also be enabled to assert the TouchPending signal through the Triggers Enable register. By setting the bits in this register, when an event is recorded in the status register the TouchPending signal is asserted, and remains asserted until the status is read.

The most basic events that are related to the touchscreen are the resistive new sample and the touchscreen release events. The resistive new sample is triggererd when a new sample of the screen coordinates is available, which means that it is produced at the touch detection and subsequently at every scan period. The release event is produced only when the screen transitions from touched to not touched. In this case the last valid coordinates value remains in the Coordinate registers.

Additional to the events status, the Status Register 1 reflects the state of the screen through its most significant bits, screen touched and two touch bits. These bits are updated with each sampling period and unlike the events bits, they are not latched and are not cleared with a register read, so their value always reflects the state of the screen at the last scanning sequence.

When the screen transitions to the touched state, the Resistive Touch Screen Touched and the Resistive New Sample Event bits in the Resistive Touch Status Register 1 are set. With every new coordinate sample, the Resistive Touch Screen Touched and the Resistive New Sample Event bits will be set after the coordinate values are available in the X and Y coordinates registers. When the screen is released, the Resistive New Sample Event bit will be set, indicating the end of the scanning sequence, but the Resistive Touch Screen Touched bit will now be cleared, reflecting the released state of the screen. After this condition is met, no more touchscreen events will be reported in the status register or the TouchPending pin until a new touch is detected in the screen.

The TouchPending signal is active low and is capable of driving an LED if user feedback is desired.

2.1.4 Calibration process

The resistive touch screen should be calibrated to get an accurate performance. This inlcudes X and Y coordinate and gesture detection. The calibration process consists of a series of points touched by the user to give the device references on the screen orientation, offsets, and linearity. Two different processes may be executed: one, for only single touch calibration, the other for single and two touch calibration. In both cases, three different points need to be touched for single touch calibration. It is recommended that these points be touched with a stylus to increase the precision achieved. In the case of two touch calibration, two pairs of points need to be touched. As expected the two touch gestures are performed with fingers, these pairs of points must also be pressed with fingers.

The three single touch calibration points are the following: The first coordinate should be at 10% of X and 10% of Y axis. The second coordinate should be at 50% of X and 90% of Y axis. The third coordinate should be at 90% of X and 50% of Y axis. The two pairs of two touch coordinates are the following and may be pressed in any order: The first pair is at 50% on the X axis and 10% and 90% on Y axis; the second pair is 50% on Y axis and 10% and 90% on X axis.

These percentages correspond to the graphic display screen resolution. The Horizontal Display Resolution and Vertical Display Resolution must be configured with the corresponding display resolution before executing the calibration process. Note that both registers must be written for their new values to take effect. When the CRTouch is calibrated with the correct resolution, the X and Y coordinates calculated by the device directly correspond to the display pixels and need no additional processing by the graphics controller. Usually the user application displays an image showing the calibration points one at a time that must be touched to calibrate the screen.

For increased precision, the CRTouch performs a validation of the touched points during calibration to determine if the values are within the expected range. When the validation is not positive, a calibration error is signaled in the Status Error register. This validation mechanism does not prevent the device from being calibrated. It serves as information to the host processor to determine if the calibration was executed correctly. Other methods of validation can be to display an additional point in the display and verify that the coordinate reported by the CRTouch corresponds to the displayed point. The built-in validation mechanism is accurate only within a 5° diphase between the display and the touch panels.

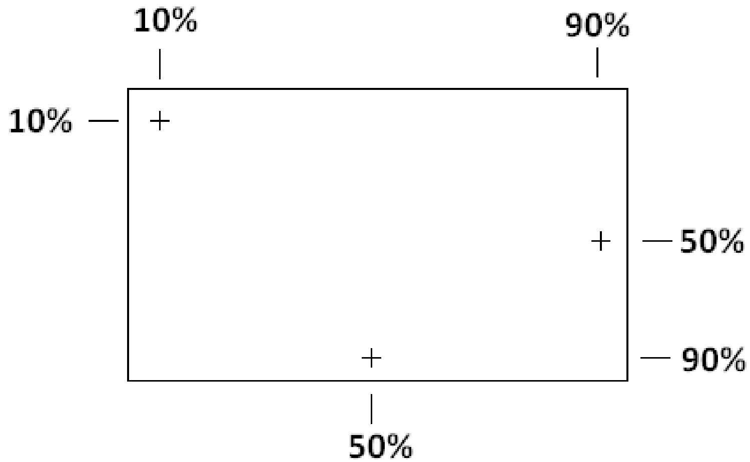


Figure 7. Touch points for single touch calibration

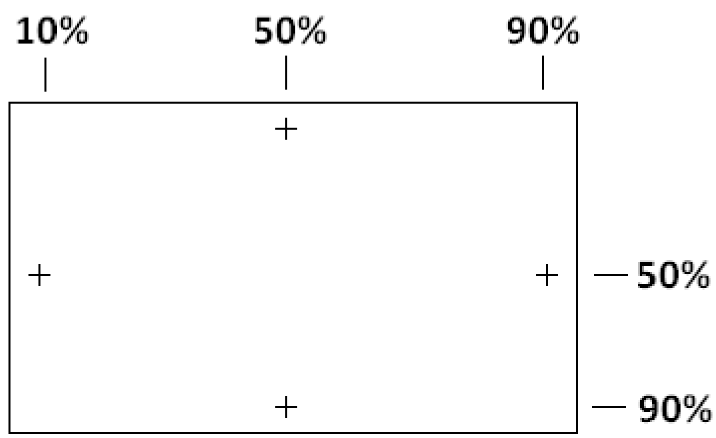


Figure 8. Touch pairs for two touch calibration

The calibration process is initiated by setting bit6 into the Trigger Events register. For two touch calibration at least one of the two touch gestures must be enabled prior to starting the calibration process. It is also important to properly configure the resolution values to match the screen ratio, to increase the precision of the gestures detection especially for the rotate gesture. The following diagrams show the sequence of actions followed after setting the calibration bit to start the process.

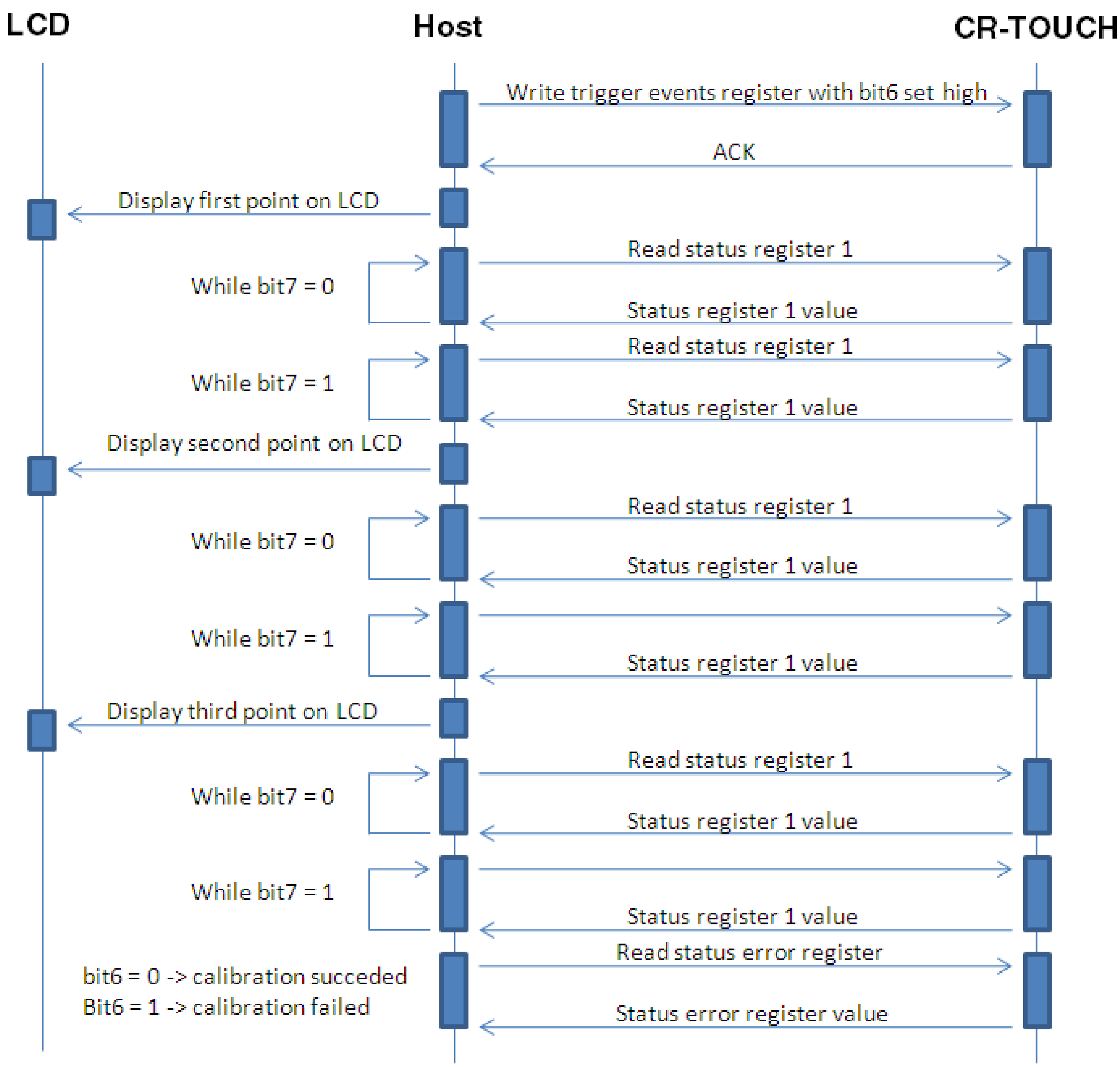


Figure 9. Single touch calibration sequence

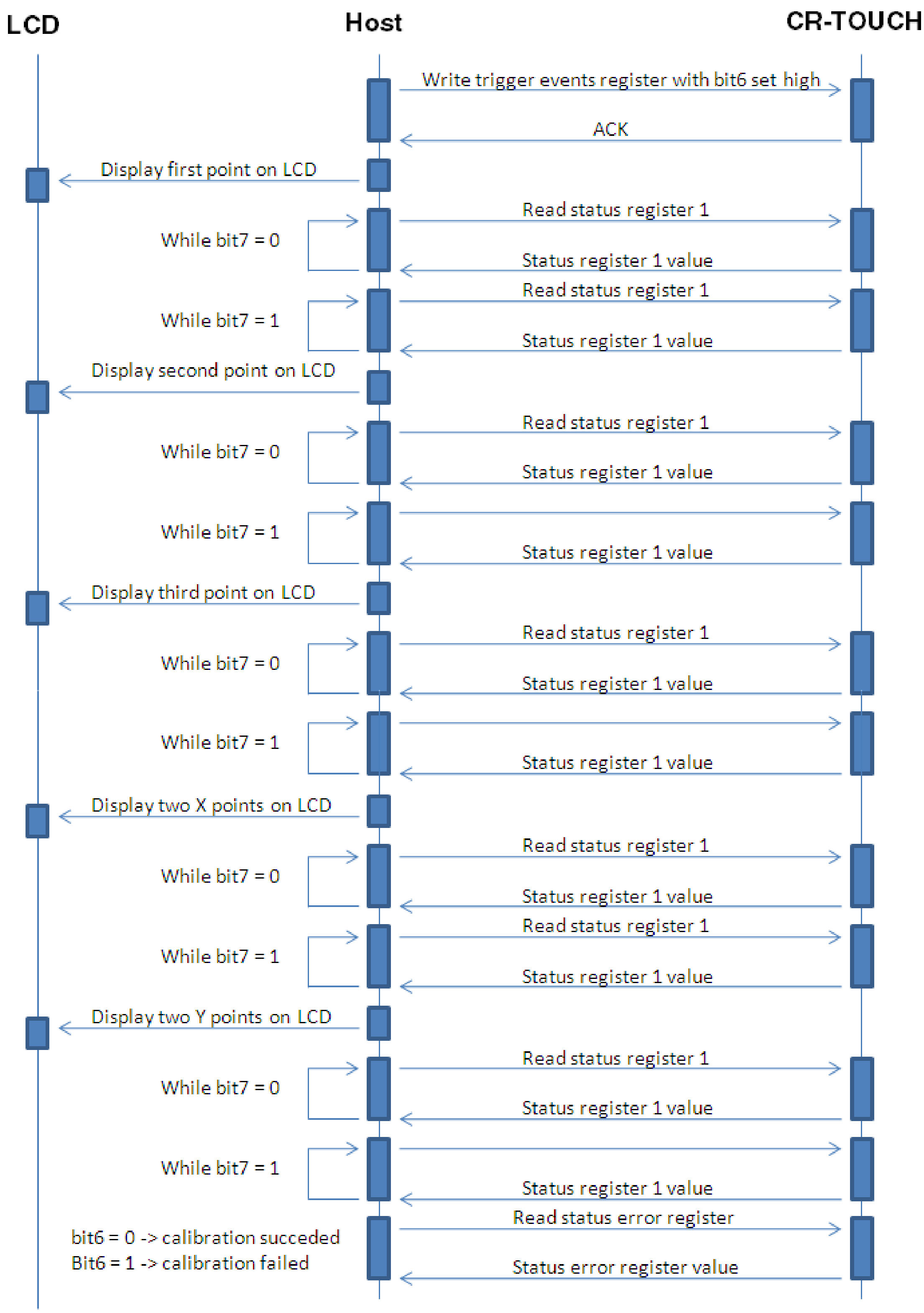


Figure 10. Single and two touch calibration sequence

During calibration, the touch screen scan period is fixed at 10 ms, regardless of the value configured in the Sampling Rate register. For each point, 128 samples are taken, so each point should be held at least for 1.3 s for proper calibration.

The calibration processed may be stopped at anytime by clearing bit6 on the Trigger Events register. Asampling period must pass before enabling it again for a new calibration process. To discard an established calibration both of the display resolution registers must be written. It is also possible to calibrate the CRTouch device through the Calibration Values set of registers without any touches required by the user. This is particularly useful for factory calibration to reduce the time required for this process. In this case, a system must be manually calibrated to generate the calibration values that can be reproduced to other devices. It is important to consider that the precision achieved by this method depends on the repeatability of the touch sensor impedances by the manufacturer. After writing the values to these registers the CRTouch must be reset for the calibration to take effect.

2.1.5 Data FIFO

The device has three FIFO buffers, one for X coordinates, one for Y coordinates, and one for Pressure values. Each FIFO buffer stores up to16 samples and new samples can not be stored once the FIFO buffer is full. Each point touched on the screen yields one X coordinate and one Y coordinate which are store onto its respective FIFO buffer. If pressure is enabled on the device, then each point yields also a pressure value that is stored onto its respective FIFO buffer.

The three FIFO buffers are synchronized and should be read in sequence. The device has a mechanism avoiding that just one FIFO buffer is read. If there are samples on the FIFO buffers and just one of them is read several times the device will report the same value until the other FIFO buffers is read. This means if the pressure is enabled the X FIFO buffer, Y FIFO buffer, and Pressure FIFO buffer should be read once before trying to read any of them again. If Pressure is disabled then only X FIFO buffer and Y FIFO buffer should be read.

Setting the RTFIFOEN bit into the FIFO Setup register enables the FIFO buffers. Clearing this bit disables the FIFO buffers. X, Y, and Pressure FIFO buffers are flushed each time the FIFO Setup register is written. The device returns the value 0xFFFF if the FIFO buffers are disabled or empty.

If the FIFO buffers are enabled a watermark can be configured to generate an event once the number of samples stored onto the FIFO buffers are bigger or equal than the watermark. The watermark is disabled by setting a zero value.

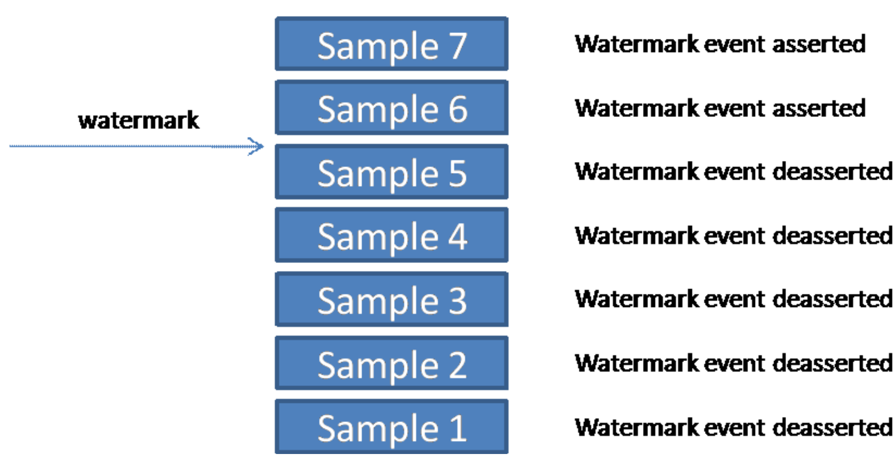


Figure 11. FIFO Watermark example

2.1.6 Resistive gestures

The CRTouch controller has built-in gestures detection capabilities with unique support for zoom and rotate multi-touch gestures, as well as single touch slides. This feature offloads the host processor of continuous coordinate reading and processing for gesture detection.

2.1.6.1 Slide

A slide is detected for linear motion in any region of the screen in a direction parallel to any axis of the screen. This means that any motion that varies only one axis can be detected as a slide. The detection algorithm supports a deviation of 7.5° from the axis direction to still consider the motion as a slide. The following figure shows the valid ranges for a slide motion.

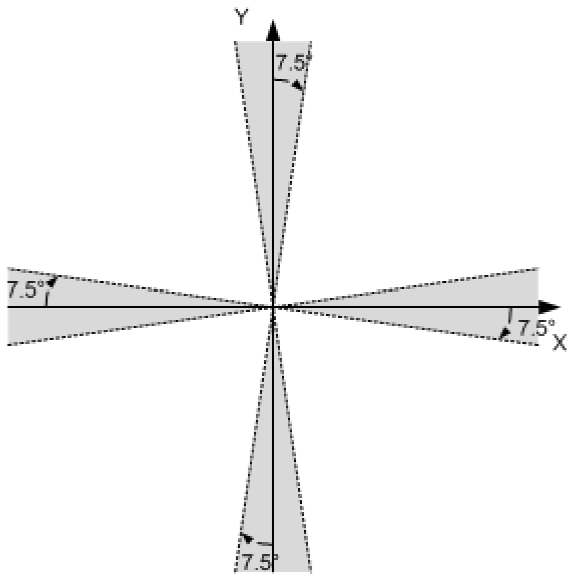


Figure 12. Slide valid regions

The slide gesture is enabled through the slide enable bit in the Configuration Register, and it is reported through the Slide Event bit in the Status Register 1. A slide can be performed in four different directions: horizontal or vertical, positive or negative in each case. The direction of the gesture is reported in the Status Register 2 through the Slide Direction [1:0] bits. The following table shows the reported value for each case.

Table 4. Slide gesture directions

Motion	Direction	Slide Direction [1:0]
X0 < X1, Y0 = Y1	Horizontal positive	00
X0 > X1, Y0 = Y1	Horizontal negative	01
Y0 < Y1, X0 = X1	Vertical positive	10
Y0 > Y1, X0 = X1	Vertical negative	11

The slide gesture is reported when the motion distance is higher than a configurable threshold. This threshold is a portion of the screen resolution across each axis. For a vertical slide, a portion of the Y resolution is the threshold and for a horizontal slide, a portion of the X resolution is the threshold. This portion is defined by the Slide Steps register value. For a value of ten, the threshold is set to one tenth of the resolution; a value of one would mean one unit of the resolution, which cannot produce a slide because no motion may have a distance greater than the resolution.

The Slide Displacement register reports the accumulated steps of the motion after it begins, therefore each time the threshold is crossed, this value is incremented. This value is reset if a change of direction is detected or when the screen is touched after a release.

2.1.6.2 Two touch gestures

The CRTouch device is capable of detecting when two independent touches are present in the screen and when Zoom and Rotate gestures are performed by these two touches. Two touch detection is enabled when either Zoom or Rotate are enabled in the Configuration register. When two touches are detected, the Two Touch bit in the Status register 1 is set. In this state, coordinates reported in the X and Y registers should be disregarded by the display controller. When detecting a two touch event, the CRTouch continuously measures displacement of the fingers. This displacement may reflect in terms of the distance between the fingers (Zoom gesture), the slope of the line between the fingers (Rotate gesture), or both.

For two touch gestures to operate, it is mandatory to run the calibration process described in section Section 2.1.4, “Calibration process.” Two touch gestures and display resolution must be configured before running the calibration to properly calibrate the gesture detection. For two touch gestures, the CRTouch uses internally a different resolution for the distances between the fingers in each axis. The smaller axis has a fixed resolution of 1000, while the other axis resolution is calculated to match the screen proportions. For a screen with 16:9 proportions, for example, one axis would have a resolution of 1000 and the other of 1777. The screen proportion is calculated according to the Vertical Resolution and Horizontal Resolution values. These values are used to calculate the distance between the fingers.

$$TwoTouchResolutionA = 1000 \qquad \textbf{Eqn. 1}$$

$$TwoTouchResolutionB = 1000 \times \frac{HorizontalResolution}{VerticalResolution} \qquad \textbf{Eqn. 2}$$

or

$$TwoTouchResolutionB = 1000 \times \frac{VerticalResolution}{HorizontalResolution} \qquad \textbf{Eqn. 3}$$

Whichever is the highest.

$$MaxTwoTouchDistance = \sqrt{ResolutionA)^2 + (ResolutionB)^2} \qquad \textbf{Eqn. 4}$$

The signals required for two touch detection are obtained through the resistors connected to the Xr and Yr signal. Unexpected behaviors may be produced if these resistors are not connected and two touch gestures are enabled.

2.1.6.3 Zoom gesture

When the distance between the fingers varies, a zoom gesture is detected. If the distance increases, a zoom-in direction is reported through the RTSZD bit in the Status Register 2. If the distance decreases, a zoom-out direction is reported through the RTSZD bit in the Status Register 2. In either case, the RTSZ bit is set in Status Register 1. Similar to the slide gesture, the zoom gesture is reported when a threshold is crossed. This threshold is approximately one tenth of the smaller axis of the screen. The Zoom Size register is updated each time the threshold is crossed and is cumulative throughout the gesture. The value of this register is equal to one tenth of the distance displacement.

$$ZoomSize = \frac{|InitDistance - ActualDistance|}{10} \qquad \textbf{Eqn. 5}$$

The maximum distance reported in the Zoom Size register can vary this depends on the proportions of the screen.

2.1.6.4 Rotate gesture

The rotate gesture is reported when a variation of the slope between the fingers is detected. This variation must be at least 10° for it to be reported in the RTSR of the Status Register 1. The direction of the rotate gesture is reported in the RTSRD in the Status Register 2. The Rotate Angle register reflects the cumulative angle that has been displaced in that direction. This angle is reported in radians with 5 fractional bits and 3 integer bits. This determines a limit of 456° of continuous rotation that can be reported by the CRTouch. The following equation can be used to convert the value of the Rotate Angle register to a value in degrees.

$$\theta = \frac{(RotateAngle)*180}{32\pi} \qquad \textbf{\textit{Eqn. 6}}$$

2.2 Capacitive subsystem

The CRTouch controller supports up to four capacitive electrodes as an extended interface. These electrodes may be detected independently or they can be arranged in a keypad, rotary or slider control configuration. The capacitive subsystem is enabled through the Configuration register with the Capacitive Control [1:0] bits according to table 4. Additionally, each of the electrodes of the CRTouch controller may be individually enabled or disabled in the Electrode Enablers register.

Table 5. Capacitive control configuration bits

Capacitive Control [1:0]	Configuration
00	Capacitive subsystem disabled
01	Rotary control enabled
10	Slider control enabled
11	Keypad control enabled

When enabled, the capacitive electrodes are scanned sequentially, from E0 to E3, at a fixed rate of 7 ms. The capacitive subsystem of the CRTouch controller has a set of configuration and status registers to control several features.

NOTE

Before the desired control is enabled through the Configuration Register, the desired events to be detected must be enabled in the Capacitive register; otherwise the control will not be enabled.

2.2.1 Capacitive touch detection

The touch detection is based on a baseline tracking algorithm and thresholds for touch detection. When the system is enabled, an initial baseline is calculated and continuous baseline recalibration can be enabled through the DCTracker feature with the DC tracker enable bit in the capacitive system configuration. This recalibration is executed at a fixed number of scanning periods, determined by the DC Tracker Rate register value. Since the scanning period is fixed at 7 ms, the recalibration period is equal to the DC Tracker Rate register value multiplied by 7 ms.

To detect a touch, the capacitance samples taken for each electrode are compared against the baseline. If a number of consecutive samples exceed a certain threshold then that electrode is considered and reported as touched. These thresholds are defined by the Ex Sensitivity registers and represent the minimum difference that must exist between a capacitive sample and the signal baseline for a touch to be detected. To avoid false touch detection due to sporadic noise, the Response Time register defines how many samples the algorithm will consider to determine a touch or a no touch condition.

After an electrode has been detected as touched, recalibration is not performed for that electrode to avoid recalibrating the baseline at a non-idle signal level. However, under certain environmental conditions, large and sudden changes in the measured capacitance may occur which may trigger a touch detection. If this condition persists (excessive humidity over an electrode, for example), the affected electrode will not be able to detect real touches. For this kind of scenario, the stuck key feature allows the host to define a touch timeout,after which the touched status is released and the electrode is recalibrated.

The actual status of each individual electrode may be read at the Electrode Status register. Each bit represents the status of one electrode, reporting a 1 when the electrode is detected as touched and a 0 when the electrode is not touched.

For application customization of these parameters, the CRTouch controller provides a set of registers that allow the analysis of the capacitance behavior of each electrode under the specific application conditions. The Electrode Baseline registers and Electrode Instant Delta registers provide this information.

The resolution of the capacitive samples and the calculated baselines can be modified through the Capacitive Resolution registers. Increasing or decreasing the resolution is useful depending on the thickness of the dielectric used. This allows to modify the signal (touch) to noise ratio for more flexibility on the touch detection. This value is only effective after a reset, a reset is then needed to make a change in this register to take effect.

2.2.2 Keypad control

The keypad control supports the reporting of touch or release events for each individual electrode. These events are reported through an events buffer and each of them is enabled through the Events register. The events buffer is mapped to the Electrodes FIFO register of the device. Each read of this register returns the oldest previously stored in the buffer, until the buffer is empty and any subsequent read returns a value of 0xFF. The format of the events in the buffer may be consulted in the registers section of this document.

Additionally to the touch detection, the keypad control provides an autorepeat feature, to log new touch events into the buffer at a certain rate after detecting the electrode as touched for a period of time. The Auto-repeat Start register controls the timeout before it starts logging new events, and the Auto-repeat Rate register controls how often a new event is stored in the buffer.

The keypad control also provides the capability of restricting the number of keys that may be pressed at the same time. This is done by writing a value different than 0 in the Max Touches register. When the number of simultaneous touches is equal to the Max Touches register, no new touch events for additional keys are stored in the events buffer until one or more keys are released.

2.2.3 Rotary and slider control

The slider and rotary control types provide support for linear or circular arrangements of electrodes. A rotary may be seen as a circular slider, where the first and last electrodes are adjacent. In the case of a rotary a simultaneous touch of these two electrodes is considered valid, where in a slider it is not. This is the only difference between these two types of controls, otherwise all of their behavior is the same.

Slider and rotary controls are oriented to the detection of motion through the control, so they provide the capability to detect this motion, report its direction, and the amount of displacement. The control is also capable of reporting when the initial touch is detected, when the motion ends and when all of the control’s electrodes have been released. All these events are enabled through the Events register.

Two status registers are provided for these controls, the Static Status register and the Dynamic Status register. The Dynamic Status register reports if movement is being detected at the moment through the Movement Flag, reports the direction of this movement in the Direction bit and how many positions were advanced since the last status. The static status reports if the control is currently touched through the touch flag, the position of the touch, and if an invalid position is detected.

To increase the number of positions detectable in a rotary or slider control, positions “between” the electrodes are also reported. This means that when two electrodes are touched, the control reports a position different than the position of each of the touched electrodes. For example, E0 corresponds to position 0, E0 and E1 correspond to position 1, and E1 corresponds to position 2. When more than two electrodes are touched, the central position is reported.

2.3 Modes of operation

The CRTouch operating modes are described in this section. Entry into each mode and exit from each mode as well as the function while in each of the modes are described.

2.3.1 Run mode

This is the normal operating mode for CRTouch which is the default mode for a new device. In this mode the device is not entering into low-power nor shutdown modes at any time. Serial communications are active all the time waiting for a command to be received. Scanning of a resistive touchscreen panel will be performed periodically as defined in the Sampling Rate register.

During run mode, all the internal circuitry remains enabled all the time. All the functions do not have any constraints or special considerations to be used while in this mode. Power consumption is higher than in any other operating mode.

2.3.2 Sleep mode

Sleep mode is enabled through the SLEEPEN bit in the Configuration Register. This mode sends the part into a low power mode between scanning periods of a resistive touchscreen panel. When the part is in sleep mode it turns off the internal clocks for the serial communication peripherals, both UART and I2C. The overall function of the device is the same compared to normal run mode, with the exception of communication interfaces.

2.3.2.1 CRTouch function in sleep mode

When Sleep mode is enabled, the part will remain in this state until the timeout configured in the Sampling rate register expires. After the Sampling Rate period expires, the part will start the sequence to scan if the resistive touch screen connected has a new coordinate and calculate the X and Y coordinates and gestures detection if needed.

Besides a sampling rate period expiration, there are two additional ways to transition from Sleep mode into Run mode:

- Wakeup pin—The active low wakeup pin will transition the part from Sleep mode into run mode. While being asserted, the device will remain in run mode. The part will return into Sleep mode after the signal has been de-asserted, unless there is an active communication (UART or I2C) or the part is actively scanning a resistive touchscreen panel. For the communication case, the communication timeout rules will apply before going back to Sleep mode. When the part is scanning the resistive touchscreen panel, it will go back to sleep mode as soon as the scanning process finishes.
- Serial communication—Either UART or I2C can transition the part from sleep mode into run mode. Each communication interface has specific characteristics and rules while working in sleep mode.

2.3.2.2 I2C communication in sleep mode

If I2C communication is selected for CRTouch communication, a start condition followed by a Slave Address match can be used to wakeup the part from sleep mode and transition to normal run mode. Because the I2C internal clock was turned off until the slave address matched, CRTouch answers with a Not Acknowledge to this initial request. There are two alternatives to use I2C communication with Sleep mode enabled:

- Include in the host the logic a re-send of the Slave Address with the appropriate read or write request upon reception of a not acknowledge.

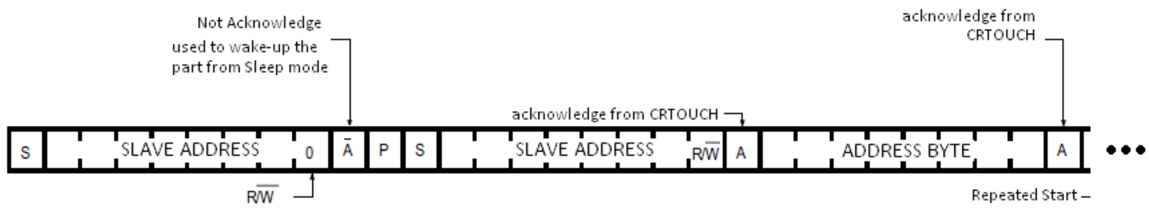


Figure 13. CRTouch wakeup using I2C communication

- Use the wakeup signal to bring the part into normal run mode before starting any I2C communication. It is important to return the wakeup signal into its idle state (high) after the communication is finished, otherwise the part will remain into normal run mode increasing the overall system power consumption.

When the wakeup pin is used to return the part to run mode, the system will remain in this state for 50 µs. If a new communication starts with a Start condition followed by the device slave address and a valid command for the CRTouch while in normal run mode, the part will remain in run state. It will return to sleep mode only after the communication remains idle for 1ms after receiving a stop condition. When the part wakes by receiving its slave address through I2C, it remains in run mode for 1ms waiting for a new communication to start. It goes back to sleep mode after 1 ms of inactivity on the I2C bus.

2.3.2.3 UART communication in sleep mode

When UART communication is used for CRTouch, a start bit will transition the part from sleep mode to run mode. When the part is in sleep mode the internal clock used for the UART communication is off. Upon reception of a start bit the internal clock is re-started, the part returns into normal run mode and the communication can be resumed normally.

In the majority of the cases the byte that wokeup the part is properly received and stored. The exception to this rule occurs when the internal circuitry is transitioning from normal run mode (either because of a previous communication, wakeup pin use, or screen X and Y coordinates calculation) into sleep mode. If the UART start bit is received at the exact moment the internal circuitry starts the transition, the initial byte (start of frame) will be lost. This will result in a complete frame loss when it occurs. There are three alternatives when enabling Sleep mode using UART:

- Send the start of frame byte twice in each frame.
- Asserting the wakeup pin for at least 10 µs before sending the start of frame. It is important to return the wakeup signal into its idle state (high) after the communication is finished, otherwise the part will remain in normal run mode increasing the overall system power consumption.
- Implementing a timeout on the host side to retry the command if there is no response received within the next 1ms of sending the command.

2.3.3 Shutdown mode

Shutdown is enabled through the SHUTDOWN bit in the Configuration register. This mode sends the part into the lowest power consumption state. In this mode, all the resistive touchscreen scanning, serial communication and any other internal activity are stopped. This mode is intended for when a device is in a standby or hibernating state and wishes to reduce power consumption to its minimum.

There are three ways to come out of Shutdown mode:

- Asserting the wakeup signal for more than 10 µs
- Using the reset pin
- Optionally enabling a capacitive electrode as wakeup source and performing a touch on it.

In all cases the part will recover the latest value for the configuration registers and will resume will resume normal operating mode. The part will use either normal run mode or Sleep mode based on the latest configuration for the SLEEPEN bit before going into Shutdown.

Any of the capacitive electrodes may be enabled as a Low Power Electrode and can be configured for a scanning period and sensitivity. To enable this electrode as a wakeup source from Shutdown, the CLPEN bit must be enabled in the Capacitive System Configuration register. Enabling this bit inhibits normal functioning of all the electrodes, therefore it should only be enabled prior to sending the device into Shutdown mode. Baseline tracking is not performed for the Low Power Electrode.

2.4 Internal voltage regulator

The voltage regulator module is a LDO linear voltage regulator to provide 3.3 V power from an input power supply varying from 2.7 V to 5.5 V. It consists of one 3.3 V power channel. The internal voltage regulator can be used to be the main power supply of the device. When the input power supply is below 3.6 V, the regulator goes to pass-through mode.

2.4.1 Internal voltage regulator features

- Low drop-out linear voltage regulator with one power channel (3.3 V)
- Low drop-out voltage— 300 mV.
- Output current—120 mA.
- Three different power modes— RUN, SLEEP, and SHUTDOWN.
- Low quiescent current in RUN mode.
- Typical value is around 120 µA (one thousand times smaller than the maximum
- load current).
- Very low quiescent current in STANDBY mode.
- Typical value is around 1 µA.
- Automatic current limiting if the load current is greater than 290 mA.
- Automatic power-up once some voltage is applied to the regulator input.
- Pass-through mode for regulator input voltages less than 3.6 V
- Small output capacitor— 2.2 µF
- Stable with aluminum, tantalum, or ceramic capacitors.

2.4.2 Internal voltage regulator modes of operation

The regulator has these power modes:

- RUN— The regulating loop of the RUN regulator and the SLEEP regulator are active, but the switch connecting the SLEEP regulator output to the external pin is open.
- SLEEP— The regulating loop of the RUN regulator is disabled and the standby regulator is active. The switch connecting the SLEEP regulator output to the external pin is closed.
- SHUTDOWN— The module is disabled.

This is the recommended connections to power CRTouch from internal voltage regulator:

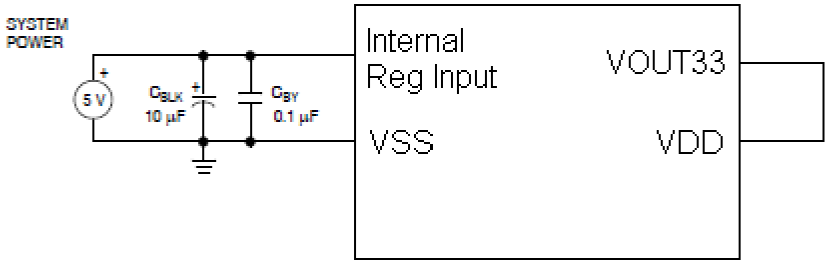


Figure 14. Connections to power CRTouch from internal voltage regulator

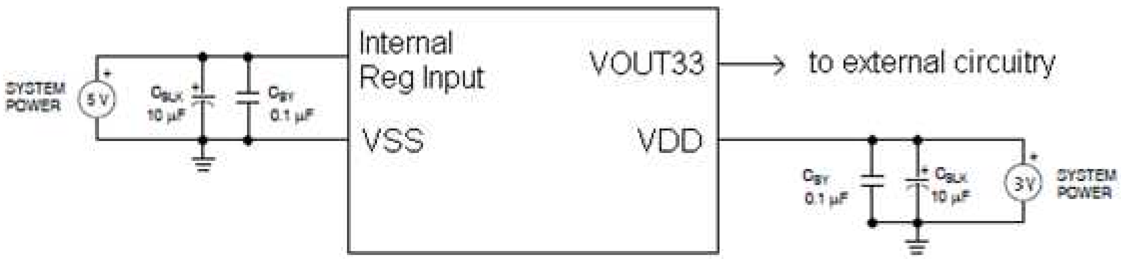


Figure 15. Recommended connections to power external circuitry from internal voltage regulator

3 Serial Communications

The CRTouch is a four wire and five wire resistive touch Screen and a four key capacitive touch sensing device. It can interact with a master device either through UART or I2C interfaces

3.1 I2C interface

The registers inside CRTouch can be accessed through the Inter-Integrated Circuit serial interface (I2C, I^2C, or IIC). The I2C interface provides a method of communication with a number of devices and the CRTouch can act only as a slave device inside an I2C network, therefore it will respond to read and write operations from a Bus master and can never initiate a communication within the bus. To enable the I2C interface, COM SEL must be tied high when coming out of reset.

CRTouch operates as an I2C slave that sends and receives data through the SDA line on an I2C bus. A bus master initiates all data transfers to and from CRTouch, and generates the SCL clock that synchronizes the data transfer.

The CRTouch line operates as an open drain bidirectional signal. A pull-up resistor (typically between 4.7 Ω and 10 KΩ) is required on CRTouch. For CRTouch, SCL is an input only signal which also requires an external pull-up.

Each transmission is initiated with a START condition generated by the master, followed by the CRTouch slave address (7 bits) plus one bit that indicates if the transaction is a read or write operation, in the case of write sequences it includes one register address byte and 1 to N data bytes for read or write transactions, followed by a STOP condition that indicates the end of that transmission.

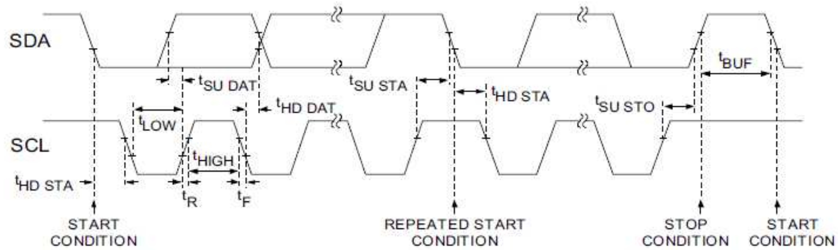


Figure 16. I2C START and STOP conditions and timing

3.1.1 I2C bit transfer

Because CRTouch can operate with different voltages, the levels of the logical low and high values are not fixed and depend on the associated level of V_{DD}.

The data on the SDA line must be stable during the HIGH period of the clock. Any change on the SDA line for data transmission can only occur when the clock signal on the SCL line is LOW.

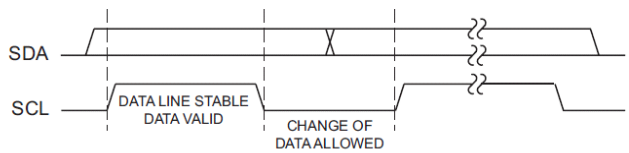


Figure 17. Bit Transfer

3.1.2 I2C START and STOP conditions

A transition to low on the SDA line while SCL is high indicates a start (S) condition. A transition to high on the SDA line while SCL is high defines a STOP (P) condition.

START and STOP conditions are always generated by the master. The bus is free when no master device is engaging the bus (both SCL and SDA are high). When the bus is free, a master may initiate communication by sending a START signal. The bus is considered to be busy after the START condition. The bus is considered to be free again at a certain time after the STOP condition.

The bus stays busy if a repeated START (Sr) is generated instead of a STOP condition. The START (S) and repeated START (Sr) are functionally identical, therefore the S symbol will be used as a generic term to represent both the START and repeated START conditions, unless Sr is particularly relevant.

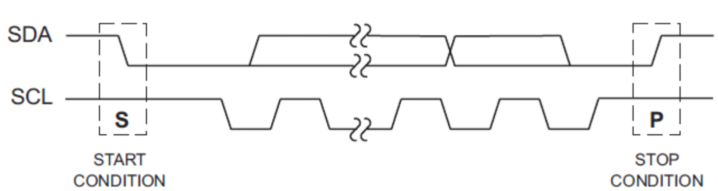


Figure 18. START and STOP conditions

3.1.3 I2C transferring data

The number of bytes that can be transmitted per transfer is unrestricted. Each byte has to be followed by an acknowledge bit. Data is transferred with the most significant bit (MSB) first. Within each byte, there must be 8-data bits and one acknowledge bit.

3.1.4 I2C acknowledge

Data transfer with acknowledge is obligatory for the I2C protocol. The acknowledge-related clock pulse is generated by the I2C master as a 9th bit which the recipient uses to handshake reception of each byte of data. Thus each byte transferred effectively requires 9 bits. To acknowledge a byte, the device sending data through the I2C bus releases the CRTouch line in the 9th bit of data. In this clock the receiver has to pull down and maintain the CRTouch line stable low during the high period of this clock pulse as an acknowledgement that the byte was received properly.

When CRTouch does not acknowledge a byte in a transaction it can indicate one of the following conditions:

- The address is trying to be accessed (either read or write) does not exist in the memory map
- The location written does not have write attributes

If the CRTouch does not acknowledge the data byte, the CRTouch line is left high during the acknowledge clock bit and the master has to generate a STOP or repeat START condition.

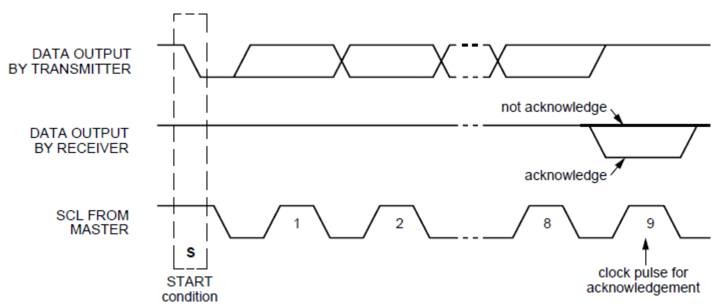


Figure 19. I2C Acknowledge

3.1.5 CRTouch I2C slave address

The CRTouch has a 7-bit long slave address. The eight bit following the slave address is the R/W bit used to indicate if the transaction is going to be a Read or Write transaction (low indicates a write command).

The I2C address has a configurable bit through the AddrSel signal. When connected to ground, bit 1 of the I2C address will be 0 and 1 when connected to V_{DD}. The resulting two 7 bit addresses possible are: 0x49 and 0x4B.

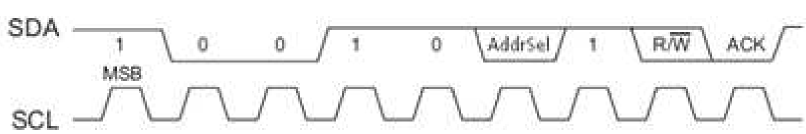


Figure 20. I2C Slave Address

3.1.6 Message format for Writing CRTouch

A write sequence comprises the transmission of the CRTouch slave address with a value of 0 for the R/W bit followed by two bytes of information. The first byte of information is the memory map address byte and the second byte is the first data byte to be written.

The memory map section of this document details the address of each register, its read and write permissions and the auto increment address in case more than 1 data byte is written. Any byte received after the address byte is taken as data bytes and is written in the location where the internal memory map logic points at that moment. A Not Acknowledge may be generated if the location written does not exist or does not have write permissions.

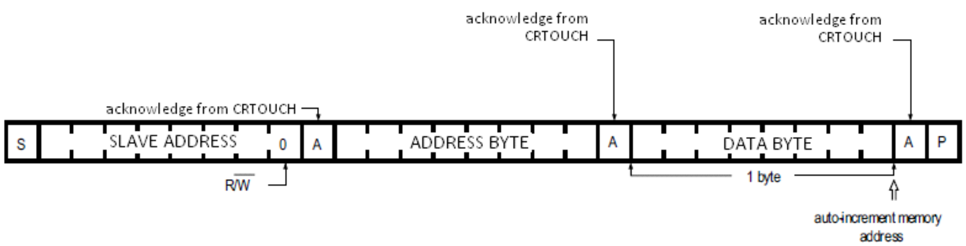


Figure 21. I2C Write sequence

3.2 UART

The CRTouch device can also send and receive data through a UART communication. Default configuration is 115200 bps, 8 data bits, odd parity and one stop bit. The only configurable parameter is the baud rate through the auto-detection feature. All the other settings are fixed.

To synchronize start and end of transactions with a host device, UART communication uses a start and end of frame reserved characters. All the data within the data payload will be sent in unpacked BCD (that is a value of 0 x 8C will be sent in two bytes, 0 x 08 and 0 x 0C) to avoid overlapping with the start and end of frame characters.

Because there are TX and RX dedicated lines for the UART communication, for any command (read or write) there must be a response from CRTouch, either replying to a read transaction or replying a status byte (success or error) from the previous transaction.

3.2.1 UART Baudrate auto detection

The CRTouch comes with an auto detect feature that allows configuring any baudrate from 9600 bps to 115200 bps. To use this feature:

1. Baudrate auto detection signal is asserted with a LOW value.
2. Host sends three consecutive bytes with 0 x 55 values with 8-data bits, odd parity, and one stop bit at the desired baudrate to configure.
3. Finally new baudrate is configured on CRTouch.

If the requested baudrate is out of range or the values sent are not the appropriate ones to calculate a valid baudrate the last valid value stored on the memory map is used and a Baudrate error will be indicated in the Status Error register.