

---

**Atmel AVR3005: Low Power QTouch Design**

---

**Atmel QTouch****Features**

---

- Factors affecting power consumption
- Guidelines to reduce power consumption in Atmel® QTouch® designs

**Introduction**

---

With the increasing use of capacitive touch based designs in variety of applications, power consumption has become a key parameter in most of the electronic appliances. Power expenditure in case of battery powered touch products is very critical, where battery life time needs to be higher.

Regardless of the touch measurement method used in designs, the system should be constructed such that it utilizes the proposed power management schemes wisely. Apart from utilizing the general sleep modes for cutting the power, there are many other touch design tips for reducing the effective power consumption of the system.

This application note discusses on factors that account for power consumption and provides guidelines to reach lowest possible power consumption in Atmel QTouch designs without compromising touch functionality and performance.

## Table of Contents

1. General Considerations .....	3
2. Factors Affecting Power Consumption.....	4
2.1 Unused peripherals .....	4
2.2 Unused GPIOs .....	4
2.3 Brown-Out Detector (BOD) .....	4
2.4 Debug interface .....	4
2.5 Sleep mode .....	4
2.6 Analog I/Os .....	5
3. QTouch Parameters.....	6
3.1 Charge transfer duration .....	6
3.2 Burst length .....	6
3.3 Detect Integration (DI).....	6
3.4 Power optimization feature.....	7
3.5 Sensor burst group.....	7
3.6 Disabling of sensors .....	7
4. Sensor Design .....	8
4.1 Electrode size.....	8
4.2 Front panel .....	8
4.3 Layout design.....	8
4.4 Sensor design components.....	8
4.4.1 Sampling capacitor (Cs).....	8
4.4.2 Sampling resistor (RSMP) .....	9
4.4.3 Series resistor .....	9
5. Software Techniques .....	10
5.1 Watchdog timer interrupt.....	10
5.2 Active and idle cycle mode.....	10
5.3 Proximity activation .....	11
5.4 Code optimization .....	12
5.5 External interrupts .....	12
6. Power Consumption Results.....	13
7. References.....	15
Appendix A. Acronyms .....	16
Appendix B. Revision History .....	17

# 1. General Considerations

Power consumption of the MCU is dependent on many factors like operating voltage, operating frequency, sleep mode, active period, sleep period, ambient temperature, etc.

The dominating factors that especially influence power consumption are operating voltage and operating frequency.

- **Operating voltage**  
Power consumption is directly proportional to operating voltage. Increase in operating voltage increases power consumption and vice-versa. Thus to conserve power, designer should consider using lower system voltage possible.
- **Operating frequency**  
Also, power consumption is dependent on operating frequency. Operating frequency affects the active time of the device. Actual power consumed will be a trade-off between active time and active current across different frequencies.

In touch designs, power consumption is greatly affected by active period of the device. So generally sleep modes of the device is utilized to reduce power consumption thereby decreasing active duration of the device. Since, in most of the real world touch applications, device need not be in active state all the time.

Theoretically average power consumption is calculated by the following formula:

$$\text{Average Power Consumption} = \frac{(\text{Active Current} * \text{Active Period}) + (\text{Sleep Current} * \text{Sleep Period})}{\text{Active Period} + \text{Sleep Period}}$$

Where,

- Active current is the current consumed by the device during touch measurements
- Active period is the duration for which device is in active mode
- Sleep current is the current consumed by the device in sleep mode
- Sleep period is the duration until the device is in sleep

This application note will provide details on adjusting active and sleep period, thereby reducing power consumption figures in final application.

Actually, in case of Atmel QTouch design there is no single hard rule for power reduction which meets all the power consumption requirements of the system. It is the effort of the designer to select the device, effectively design touch sensors, and utilize software techniques to achieve considerable power reduction.

## 2. Factors Affecting Power Consumption

In Atmel QTouch designs, apart from operating voltage and frequency, the factors that also contribute greatly to MCU power consumption are listed below.

### 2.1 Unused peripherals

Keeping the unused peripherals active increases power consumption in active period. Among the available peripherals the peripherals that are not used in the application can be shut down by disabling its clock. This is possible by using appropriate power reduction registers available in the device, which stops clock supply to individual peripherals.

**TIP:** Disable unused peripherals in active period of the device to cut down power consumed.

### 2.2 Unused GPIOs

It is recommended to ensure that the unused general purpose I/O-pins have a defined level. Though most of the digital inputs are disabled in deep sleep modes, floating inputs should be avoided in other modes like Active mode, ADC Noise Reduction mode, Power-save mode and Idle mode to reduce power consumption.

**TIP:** Configure the unused general purpose I/O-pins as Input with internal pull-up enabled to ensure defined level and for reduced power consumption.

### 2.3 Brown-Out Detector (BOD)

The purpose of Brown-Out Detector is to actively monitor power supply voltage. BOD is configured by BODLEVEL fuses. BOD tracks for indeterminate voltage levels even during sleep mode. In Atmel picoPower® devices, to reduce power it is recommended to disable BOD in Power-down sleep mode.

When BOD is disabled in software, it is turned off immediately after entering the sleep mode and automatically turned on upon wake-up for monitoring supply voltage.

**TIP:** When using Atmel picoPower devices, disable BOD in sleep modes in application software to conserve power.

### 2.4 Debug interface

When on-chip debugging is enabled in fuse settings (OCDEN/DWEN), device will consume considerable amount of power even in sleep mode, since the main clock source is enabled. In deeper sleep modes this will affect the total power consumption considerably.

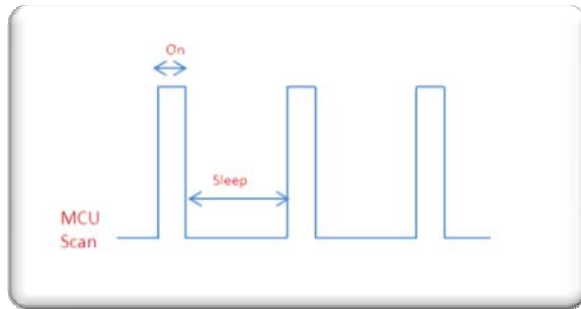
**TIP:** Disable JTAG interface and debugWIRE in the final application to reduce power consumption.

### 2.5 Sleep mode

When the device is active continuously, power consumption will increase. Sleep modes can be used to reduce the active period and reduce the power consumed. Most commonly available sleep modes are Idle, Power-down, and Power-save.

Different sleep modes, utilizes different set of peripherals. Among the sleep modes, Power-down mode is the best sleep mode, since this mode shuts down almost all the peripherals resulting in good power reduction. Based on application power requirements, active and sleep duration of the device can be configured.

Figure 2-1. Sleep mode.



**TIP:** Keep the device in active mode for a defined shorter duration and in sleep mode for longer duration.

## 2.6 Analog I/Os

In Atmel QMatrix designs, ADC pins are used as touch sense lines. The digital input from these pins is not required. So, to disable the digital input buffer, the corresponding bits in DIDR register should be set to reduce power consumption.

**TIP:** Disable the digital input buffer in the ADC pins used for analog operations.

## 3. QTouch Parameters

### 3.1 Charge transfer duration

During touch acquisition, a sequence of charge transfer occurs to perform capacitive touch measurement; Atmel QTouch Library provides a software parameter which controls the charge transfer duration.

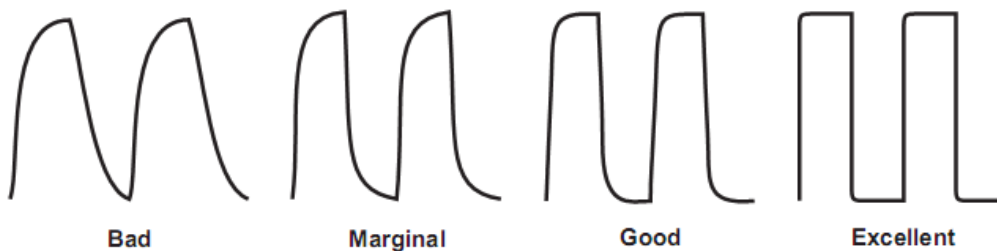
QT\_DELAY\_CYCLES → For Atmel QTouch and QMatrix methods

DEF\_QT\_CHARGE\_SHARE\_DELAY → QTouchADC method

This parameter need to be tuned to obtain proper charge in the sensor electrode. Using coin and scope method to measure charge-transfer pulses and verify that the pulse shape is proper as indicated in [Figure 3-1](#). Refer to "Section 2.1.2 – Charge Transfer" in "[QTAN0079: Buttons, Sliders and Wheels Sensor Design Guide](#)" for more details on coin and scope method. Increase in charging duration increases touch acquisition time (active time), thereby increasing total power consumption.

**TIP:** Use coin and scope method to obtain optimal value for Delay Cycles parameter.

**Figure 3-1. Good and bad charge pulses.**



### 3.2 Burst length

In QMatrix based designs, increasing Burst Length improves the sensitivity of sensors. Burst Length is a configurable software parameter for individual sensors, which actually defines the number of charge transfer sequences that need to be performed for capacitive touch measurement. Set the burst length for each sensor such that the reverse voltage build up is between 70mV to 100mV, but should never increase above 250mV. Increase or decrease burst length as required to achieve adequate sensitivity. Using oscilloscope, probe the Yb line to verify for Cs saturation during charging as explained in 'Section 2.6' in the application note "[QTAN0062: QTouch and QMatrix Sensitivity Tuning for Keys, Sliders and Wheels](#)".

Increasing Burst Length increases signal resolution, which also increases touch measurement time that contributes to increase in total power consumption.

**TIP:** Use appropriate value of Burst Length; such that the reverse voltage build up is between 70mV to 100mV and the sensors have adequate sensitivity.

### 3.3 Detect Integration (DI)

The QTouch Library features detect integration mechanism, which is used to confirm touch detection. DI basically suppresses false detections caused by spurious events like electrical noise. When there is a touch in sensor, the per-sensor DI counter increments each time the acquired sensor delta exceeds detect threshold in succession. If the touch delta is maintained for DI count times above the threshold level, this declares a valid touch. Also, DI is applicable for release of touch sensor.

Increasing DI increases touch measurement duration (active time), thereby increasing power consumption. Though, higher value improves system noise immunity.

**TIP:** Use typical value for DI (say for example 4), such that there is a trade-off between response time, noise immunity and power consumption.

### 3.4 Power optimization feature

Atmel QTouch Library provides an option in software to reduce the power consumed by the library. This option is configurable using the macro define, `'_POWER_OPTIMIZATION_'`. When this is enabled it curbs some functionality, but provides almost 40% power reduction in QTouch.

This feature is applicable only for Atmel QTouch method. When power optimization is enabled, unused pins within a port used for QTouch, may not be usable for other applications. This feature also disables the spread spectrum noise reduction. This feature is applicable only for Atmel tinyAVR® and Atmel megaAVR® touch libraries; and the Atmel AVR XMEGA® and Atmel AVR® UC3 libraries are by default optimized for reducing power consumption.

**TIP:** Enable the Power Optimization feature in noiseless QTouch designs where power requirements are critical, pin complexity does not exist and when radiated emissions is not a requirement.

### 3.5 Sensor burst group

Sensor configuration should be chosen as per the application requirement. Touch library scans the sensors connected in even channel in parallel and then scans the sensors connected in odd channels.

For example, when 8-channels are configured in an application, then library will scan the sensors on channels 0, 2, 4, and 6 in parallel, followed by sensors on channels 1, 3, 5, and 7.

When the application needs only four sensors, configure all the sensors either in even channel group or in odd channel group which allows the touch library to measure sensors faster. In this case, it is also desirable to use intra-port configuration instead of inter-port configuration, since all sensors are grouped and scanned in parallel in intraport configuration. As this configuration reduces the scan interval, the device power consumption will certainly come down.

**TIP:** Use intra-port sensor configuration, when the design uses less number of sensors (less than or equal to 4).

### 3.6 Disabling of sensors

Among the sensors used in the design, all sensors need not be active all the time. This is however application dependent. These sensors can be disabled in runtime to reduce the total active time. The disabled sensors can be re-enabled whenever required.

**TIP:**

- Disable sensors in QTouch method using the `qt_reset_sensing()` api, which disables all sensors and resets all configuration settings to their default values
- Disable sensors in Atmel QMatrix method by setting `Burst_Length` parameter to 0 in generic touch libraries for 8-bit AVR devices and by setting `Burst_Length` parameter to 1 in UC3 device specific libraries

Refer to 'Section 5.6.5' in "[Atmel QTouch Library User Guide](#)" for more details on usage of public routines available in QTouch library for disabling and re-enabling sensors as necessary.

## 4. Sensor Design

Touch sensor electrodes should be designed optimally to reduce power consumed in the system. For optimal sensor design, designer need to follow the recommendations provided in '[QTAN0079: Buttons, Sliders and Wheels Sensor Design Guide](#)'.

### 4.1 Electrode size

The electrode size should be of the same size or slightly larger than the object being sensed. In most cases it is the human finger. It is good practice to design the sensor electrode which is slightly larger in size compared to the graphic symbol present in the front panel. Typically the electrode size needs to be at least 6 – 8mm i.e., the size of human finger to couple sufficient charge during touch.

**TIP:** Use typical size for sensor electrode around 6 – 8mm or the size of the object being sensed.

### 4.2 Front panel

Front panel material used and its thickness greatly influence sensitivity of the sensors. Using thicker front panels, sensitivity of the sensors need to be compensated with other software and/or hardware parameters, which results in increase in power consumption. Front panel materials with lesser dielectric constant will have reduced field penetration at the electrode surface. Again this affects the sensitivity of the sensors, which need to be compensated by other means which increases power consumption.

Refer to “Section 2.3.3: Front Panel Materials” in '[QTAN0079: Buttons, Sliders and Wheels Sensor Design Guide](#)', for more details on front panel materials and also find dielectric constants for commonly used materials.

**TIP:** Use front panel materials with appropriate dielectric constant and reduced thickness possible.

### 4.3 Layout design

Sensor routing should be proper such that sense traces connecting sensor electrode to the device are as short as possible and the sense components are located close by the touch device. Having longer sense traces increases stray capacitance, which affects power consumption. Also longer traces pick up noise easily which might lead to instability in touch detection. To balance the sensitivity of the sensors tuning of software parameters like DI, Delay cycles and burst length is needed.

**TIP:** Lay shorter sense traces and optimize PCB design, to achieve good system power reduction.

### 4.4 Sensor design components

Tune the sensors to get adequate sensitivity and target for typical touch delta value of 15. Since unnecessarily boosting the sensor signal values to higher number, would result in increased power consumption. Refer to the application note, "[QTAN0062: QTouch and QMatrix Sensitivity Tuning for Keys, Slider and Wheels](#)" for touch sensor tuning guidelines.

Adjust the design components like sampling capacitor, sampling resistor, etc., as detailed below.

#### 4.4.1 Sampling capacitor (Cs)

In Atmel QTouch method, increasing sampling capacitor (Cs) increases sensitivity of the sensors and vice-versa.

Increasing Cs increases signal resolution, thereby increases touch measurement time (active time) which results in increased power consumption.

**TIP:** Choose optimal value for Cs, which is a trade-off between sensitivity and power consumption.



#### 4.4.2 Sampling resistor (RSMP)

In Atmel QMatrix method, altering sampling resistor is one option for varying the sensitivity of sensors. This parameter controls the discharge duration of the sampling capacitor (Cs).

Increasing sampling resistor increases sensitivity of the sensors, as increase in discharge slope directly increases signal resolution. Increase in discharge duration would affect touch measurement time (active time).

**TIP:** Use typical values for sampling resistors, which provides sufficient sensitivity to conserve power. Typically used sampling resistor is 470kΩ and can be varied between 470kΩ and 1MΩ.

#### 4.4.3 Series resistor

Generally series resistors (referred to as Rs in QTouch or Rx, Ry in QMatrix) are used for improving EMC/EMI/ESD performance of the sensor. Ensure for proper charge pulse shape in sensor electrode using coin and scope method, when altering series resistors to improve noise immunity. As increasing series resistor, affects settling time of charge transfer pulses using unnecessarily large values for series resistor affects the power consumption of the system.

For information on how series resistor helps in improving system noise immunity, refer to the application note, "[Atmel AVR3000: QTouch Conducted Immunity](#)".

**TIP:** Use typical value for series resistors, which fully charges the electrode and provides sufficient noise immunity. Typically 1kΩ is used in normal designs and Rs is varied between 1kΩ and 100kΩ in the designs that require immunity to conducted noise.

## 5. Software Techniques

There are many touch products that demand huge power saving options, say to extend the life time of the battery. Listed below are few additional suggestions to conserve power.

### 5.1 Watchdog timer interrupt

Most of the applications uses timer interrupts for performing periodic touch measurement. Here device is kept active all the time which increases total power consumption, unless there is a forced sleep period introduced between successive touch measurements.

Timer interrupts cannot be used as wakeup source, to wake the device from deep sleep modes. So to reduce power consumption significantly, suggestion is to use watch dog timer interrupts for performing periodic touch measurements. When touch acquisition is done once, then the device is put in sleep mode (power-down mode) until the watch dog timer expires to initiate next round of touch acquisition.

**TIP:** Use watch dog timer interrupts for invoking touch measurements.

### 5.2 Active and idle cycle mode

To achieve better power reduction, active and idle cycles can be implemented in application software. Touch measurement need to be invoked at regular intervals based on application requirement. Typically 25ms interval is used for touch measurement, increasing the interval affects the response time of the sensors.

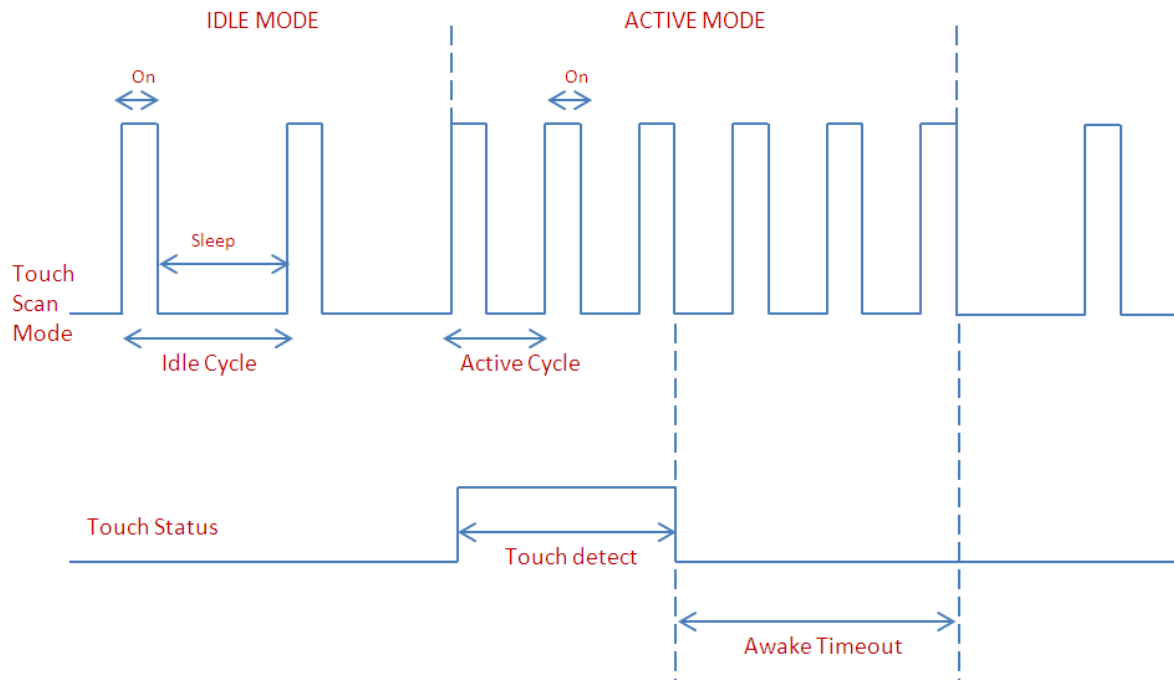
Active cycle, where touch measurement is polled periodically in a faster rate, for example say every 50ms.

Idle cycle, where touch measurement is polled periodically in a slower rate, for example say every 250ms. Here, increasing the time interval between successive touch acquisitions is by introducing sleep periods, which can typically be in the range 100ms ~ 500ms.

The goal is to keep the device in idle mode as long as there is no touch activity. When there is a touch detected, then the device will enter active cycle mode. To recognize the first touch detect when the device is in idle mode, might be slightly sluggish. Later on device in active cycle mode will have faster response. The device will stay in active mode for a defined time interval, and then revert back to idle mode if there is no touch activity during the timeout period.

As the device is active for defined time interval during touch detect which would consume more power. And then the device enters idle mode and stays there indefinitely as long as there is no touch activity, which consumes very less power. Effectively this method reduces power consumption of the device to greater extent.

Figure 5-1. Active and idle cycle modes.



### 5.3 Proximity activation

Another method to achieve significant power reduction is to implement an additional proximity sensor in the design. This method requires two devices. One device which is mainly responsible for touch applications and the other device is responsible for single channel proximity sensor.

The proximity controller can be chosen such that it can support a single channel, for example generic AVR device like Atmel ATtiny10 or a fixed function device like Atmel AT42QT1010. The Proximity sensor should be designed such that it surrounds all the actual touch sensors present in the system, as shown in [Figure 5-2](#).

The target here is to keep the main touch device in deep sleep mode indefinitely, and use the proximity sensor detect state as external interrupt for the main device to wake up from sleep. Thus the main controller is in indefinite sleep, and when the user finger gets closer to touch system activates the proximity sensor. This in turn interrupts and wakes the main controller from sleep to perform regular touch measurements.

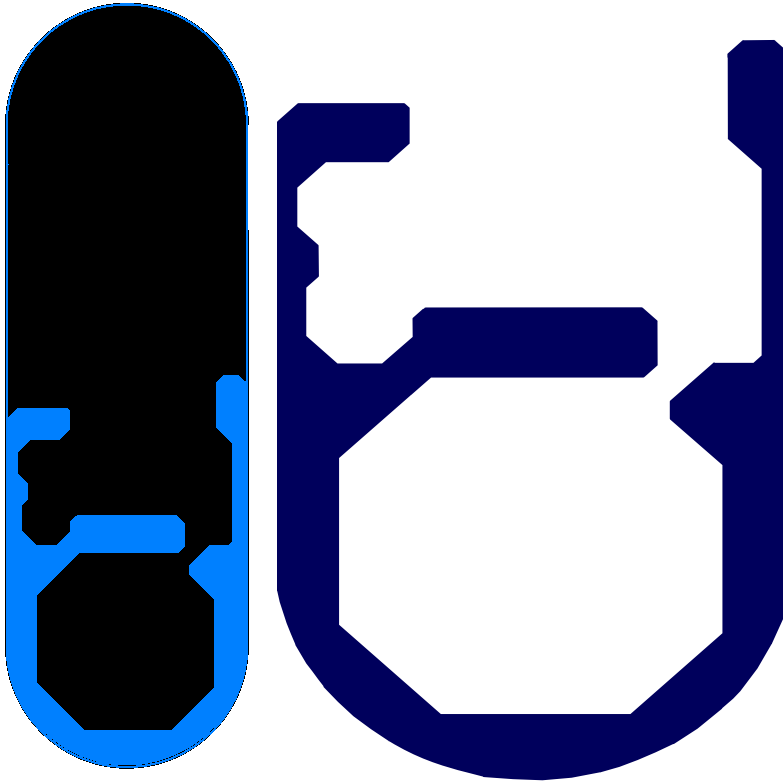
This method will also have an awake period, which will put the main controller back in sleep mode when there is no touch activity until the specified awake period.

Utilize this method where power saving is vital. The trade-off of this method is that this uses two devices.

**TIP:**

- Use a single channel Atmel device to design a proximity sensor which surrounds all the sensors exist in design
- Utilize the proximity detection as external interrupt for the actual touch device which is normally put in sleep

Figure 5-2. Proximity sensor used for wakeup.



## 5.4 Code optimization

Among the available compiler tool-chains GCC and IAR™, IAR compiler provides the best code optimization compared to GCC. It is preferable to choose IAR compiler where power requirements is critical. During project compilation it is advocated to select highest level of optimization for 'Speed', to reduce the power consumption.

**TIP:**

- Use IAR compiler Tool-chain for critical power requirements
- Select highest optimization level for Speed

## 5.5 External interrupts

The system which has two devices, say one is master and the other is responsible for touch activities. And the touch device sends touch measurement results to main device (master), for further action. External interrupts from master can be used to perform touch measurement periodically.

In this case, touch device will be kept in sleep indeterminately and the main device sends an interrupt to touch device to perform touch measurements periodically. Using this method power consumption of the touch device will be reduced greatly, as active time is reduced.

**TIP:** Use external interrupts generated by another master device for performing touch measurements.

## 6. Power Consumption Results

With the guidelines adopted, [Table 6-2](#) and [Table 6-4](#) represents average power consumption calculated for Atmel ATtiny88 device using various sensor and sleep period configurations. The test results compare power consumption figures obtained for both Atmel QTouch and Atmel QMatrix method designs.

**Table 6-1. Test setup details.**

Device	Atmel ATtiny88
Technology	Atmel QTouch
Operating voltage	3.3V
Operating frequency	4MHz
Library version	5.0
Sampling capacitor (Cs)	22nF
Series resistor (Rs)	1kΩ
WDT	Enabled
BOD	Disabled
Power save feature	Enabled
QT_DELAY_CYCLES	5
Sleep mode	Power down

**Table 6-2. Power consumption results.**

No. of channels [Keys]	Measurement cycle duration [ms]	Active period [ms]	Active current [μA]	Sleep period [ms]	Sleep current [μA]	Average current [μA]
8	32	5	1527.2	27	4.05	242.042
8	64	5	1527.2	59	4.05	123.046
8	125	5	1527.2	120	4.05	64.928
8	250	5	1527.2	245	4.05	34.464
8	500	5	1527.2	495	4.05	19.232
4	32	4	1506.8	28	4.05	191.894
4	64	4	1506.8	60	4.05	97.972
4	125	4	1506.8	121	4.05	52.138
4	250	4	1506.8	246	4.05	28.094
4	500	4	1506.8	496	4.05	16.072
1	32	1.96	1463.8	30.04	4.05	93.46
1	64	1.96	1463.8	62.04	4.05	48.755
1	125	1.96	1463.8	123.04	4.05	26.938
1	250	1.96	1463.8	248.04	4.05	15.494
1	500	1.96	1463.8	498.04	4.05	9.772

**Table 6-3. Test setup details.**

Device	Atmel ATtiny88
Technology	Atmel QMatrix
Operating voltage	3.3V
Operating frequency	4MHz
Library version	5.0
Sampling capacitor (Cs)	10nF
X-line series resistor (Rx)	1k $\Omega$
Y-line series resistance (Ry)	1k $\Omega$
Sampling resistance (R <sub>sm</sub> )	470k $\Omega$
WDT	Enabled
BOD	Disabled
QT_DELAY_CYCLES	4

**Table 6-4. Power consumption results.**

No. of channels [Keys]	Measurement cycle duration [ms]	Active period [ms]	Active current [ $\mu$ A]	Sleep period [ms]	Sleep current [ $\mu$ A]	Average current [ $\mu$ A]
8	32	6	1542.8	26	4.05	292.566
8	64	6	1542.8	58	4.05	148.308
8	125	6	1542.8	119	4.05	77.91
8	250	6	1542.8	244	4.05	40.98
8	500	6	1542.8	494	4.05	22.515
4	32	3	1534.8	29	4.05	147.558
4	64	3	1534.8	61	4.05	75.804
4	125	3	1534.8	122	4.05	40.788
4	250	3	1534.8	247	4.05	22.419
4	500	3	1534.8	497	4.05	13.235
1	32	0.8	1608	31.2	4.05	44.149
1	64	0.8	1608	63.2	4.05	24.099
1	125	0.8	1608	124.2	4.05	14.315
1	250	0.8	1608	249.2	4.05	9.183
1	500	0.8	1608	499.2	4.05	6.616

## 7. References

- 1) QTAN0079: Buttons, Sliders and Wheels Sensor Design Guide  
<http://www.atmel.com/images/doc10752.pdf>
- 2) QTAN0062: QTouch and QMatrix Sensitivity Tuning for Keys, Slider and Wheels  
<http://www.atmel.com/images/QTAN0062.pdf>
- 3) AN-KD02: Secrets of a Successful QTouch Design  
[http://www.atmel.com/images/an-kd02\\_103-touch\\_secrets.pdf](http://www.atmel.com/images/an-kd02_103-touch_secrets.pdf)
- 4) Atmel QTouch Library User Guide  
<http://www.atmel.com/Images/doc8207.pdf>
- 5) Atmel AVR3000: QTouch Conducted Immunity  
<http://www.atmel.com/images/doc8425.pdf>

## Appendix A. Acronyms

### **QTouch method**

A type of capacitive touch sensing technology using self capacitance – each channel has only one electrode.

### **QMatrix method**

A type of capacitive touch sensing technology using mutual capacitance – each channel has a drive electrode (X) and a receive electrode (Y).

### **Signals**

Raw measurement for capacitance change on a channel.

### **Reference**

Long term average measurement of Signal data on a channel.

### **Touch Delta**

Difference between Reference and Signal values.

### **Detect Threshold**

Determines the delta required for a valid touch.

### **Intra-port Configuration**

A configuration for QTouch acquisition method libraries, when the sensor SNS and SNSK pins are available on the same port.

### **Inter-port Configuration**

A configuration for QTouch acquisition method libraries, when the sensor SNS and SNSK pins are available on distinct ports.

### **Touch Acquisition**

A single capacitive touch measurement process.

### **Electrode**

Electrodes are typically areas of copper on a printed circuit board but can also be areas of clear conductive indium tin oxide (ITO) on a glass or plastic touch screen.



## Appendix B. Revision History

Doc. Rev.	Date	Comments
42056A	12/2012	Initial document release

**Atmel Corporation**

1600 Technology Drive  
San Jose, CA 95110  
USA

**Tel:** (+1)(408) 441-0311

**Fax:** (+1)(408) 487-2600

[www.atmel.com](http://www.atmel.com)

**Atmel Asia Limited**

Unit 01-5 & 16, 19F  
BEA Tower, Millennium City 5  
418 Kwun Tong Road  
Kwun Tong, Kowloon  
HONG KONG

**Tel:** (+852) 2245-6100

**Fax:** (+852) 2722-1369

**Atmel Munich GmbH**

Business Campus  
Parking 4  
D-85748 Garching b. Munich  
GERMANY

**Tel:** (+49) 89-31970-0

**Fax:** (+49) 89-3194621

**Atmel Japan G.K.**

16F Shin-Osaki Kangyo Building  
1-6-4 Osaki  
Shinagawa-ku, Tokyo 141-0032  
JAPAN

**Tel:** (+81)(3) 6417-0300

**Fax:** (+81)(3) 6417-0370

© 2012 Atmel Corporation. All rights reserved. / Rev.: 42056A-AVR-12/2012

Atmel®, Atmel logo and combinations thereof, AVR®, Enabling Unlimited Possibilities®, megaAVR®, picoPower®, tinyAVR®, QTouch®, XMEGA®, and others are registered trademarks or trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be trademarks of others.

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