Chipsmall
深圳市津利帝科技有限公司
Geelead Technology Limited
Chipsmall
采购部

SEMTECH

SX8645

Ultra Low Power, Capacitive Button and Wheel Touch Controller (12 sensors) with Enhanced LED Drivers

ADVANCED COMMUNICATIONS & SENSING DATASHEET

GENERAL DESCRIPTION

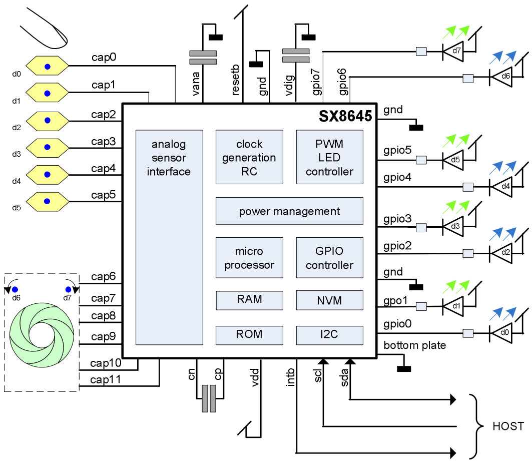
The SX8645 is an ultra low power, fully integrated 12-channel solution for capacitive touch-buttons and wheel applications. Unlike many capacitive touch solutions, the SX8645 features dedicated capacitive sense inputs (that requires no external components) in addition to 8 general purpose I/O ports (GPIO). Each GPIO is typically configured as LED driver with independent PWM source for enhanced lighting control such as intensity and fading.

The SX8645 includes a capacitive 10 bit ADC analog interface with automatic compensation up to 100pF. The high resolution capacitive sensing supports a wide variety of touch pad sizes and shapes and allows capacitive buttons to be created using thick overlay materials (up to 5mm) for an extremely robust and ESD immune system design.

The SX8645 incorporates a versatile firmware that was specially designed to simplify capacitive touch solution design and offers reduced time-to-market. Integrated multi-time programmable memory provides the ultimate flexibility to modify key firmware parameters (gain, threshold, scan period, auto offset compensation…) in the field without the need for new firmware development.

The SX8645 supports the 400 kHz I²C serial bus data protocol and includes a field programmable slave address. The tiny 5mm x 5mm footprint makes it an ideal solution for portable, battery powered applications where power and density are at a premium.

TYPICAL APPLICATION CIRCUIT



KEY PRODUCT FEATURES

- ♦ Complete Twelve Sensors Capacitive Touch Controller for Buttons and Wheel
  - ▪ Pre-configured for 6 Buttons and a Wheel
  - ▪ 8 LED Drivers with Individual Intensity, Fading Control and Autolight Mode
  - ▪ 256 steps PWM Linear and Logarithmic control
- ♦ High Resolution Capacitive Sensing
  - ▪ Up to 100pF of Offset Capacitance Compensation at Full Sensitivity
  - ▪ Capable of Sensing through Overlay Materials up to 5mm thick
- ♦ Extremely Low Power Optimized for Portable Application
  - ▪ 8uA (typ) in Sleep Mode
  - ▪ 80uA (typ) in Doze Mode (Scanning Period 195ms)
  - ▪ 220uA (typ) in Active Mode (Scanning Period 30ms)
- ♦ Programmable Scanning Period from 15ms to 1500ms
- ♦ Auto Offset Compensation
  - ▪ Eliminates False Triggers due to Environmental Factors (Temperature, Humidity)
  - ▪ Initiated on Power-up and Configurable Intervals
- ♦ Multi-Time In-Field Programmable Firmware Parameters for Ultimate Flexibility
  - ▪ On-chip user programmable memory for fast, self contained start-up
- ♦ "Smart" Wake-up Sequence for Easy Activation from Doze
- ♦ No External Components per Sensor Input
- ♦ Internal Clock Requires No External Components
- ♦ Differential Sensor Sampling for Reduced EMI
- ♦ 400 KHz Fast-Mode I²C Interface with Interrupt
- ♦ -40°C to +85°C Operation

APPLICATIONS

- ♦ Notebook/Netbook/Portable/Handheld computers
- ♦ Cell phones, PDAs
- ♦ Consumer Products, Instrumentation, Automotive
- ♦ Mechanical Button Replacement

ORDERING INFORMATION

Part Number	Temperature Range	Package
SX8645I05AWLTRT[1]	-40°C to +85°C	Lead Free MLPQ-W32

[1] 3000 Units/reel

* This device is RoHS/WEEE compliant and Halogen Free

Table of Contents

1 General Description

1.1 Pin Diagram

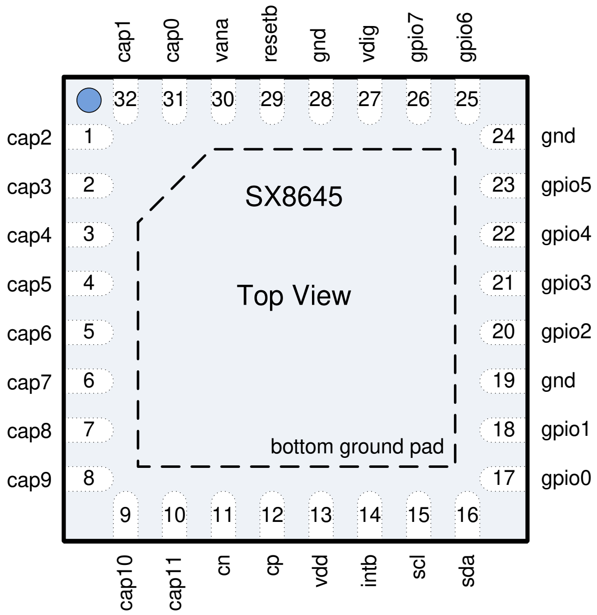
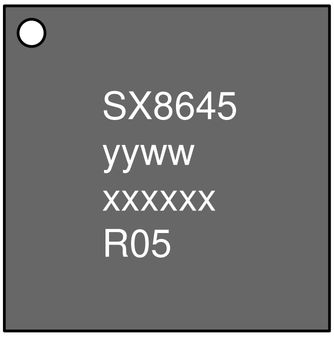


Figure 1 Pinout Diagram

1.2 Marking information



yyww = Date Code
xxxxxx = Semtech lot number
R05 = Semtech Code

Figure 2 Marking Information

1.3 Pin Description

Number	Name	Type	Description
1	CAP2	Analog	Capacitive Sensor 2
2	CAP3	Analog	Capacitive Sensor 3
3	CAP4	Analog	Capacitive Sensor 4
4	CAP5	Analog	Capacitive Sensor 5
5	CAP6	Analog	Capacitive Sensor 6
6	CAP7	Analog	Capacitive Sensor 7
7	CAP8	Analog	Capacitive Sensor 8
8	CAP9	Analog	Capacitive Sensor 9
9	CAP10	Analog	Capacitive Sensor 10
10	CAP11	Analog	Capacitive Sensor 11
11	CN	Analog	Integration Capacitor, negative terminal (1nF between CN and CP)
12	CP	Analog	Integration Capacitor, positive terminal (1nF between CN and CP)
13	VDD	Power	Main input power supply
14	INTB	Digital Output	Interrupt, active LOW, requires pull up resistor (on host or external)
15	SCL	Digital Input	I2C Clock, requires pull up resistor (on host or external)
16	SDA	Digital Input/Output	I2C Data, requires pull up resistor (on host or external)
17	GPIO0	Digital Input/Output	General Purpose Input/Output 0
18	GPIO1	Digital Input/Output	General Purpose Input/Output 1
19	GND	Ground	Ground
20	GPIO2	Digital Input/Output	General Purpose Input/Output 2
21	GPIO3	Digital Input/Output	General Purpose Input/Output 3
22	GPIO4	Digital Input/Output	General Purpose Input/Output 4
23	GPIO5	Digital Input/Output	General Purpose Input/Output 5
24	GND	Ground	Ground
25	GPIO6	Digital Input/Output	General Purpose Input/Output 6
26	GPIO7	Digital Input/Output	General Purpose Input/Output 7
27	VDIG	Analog	Digital Core Decoupling, connect to a 100nF decoupling capacitor
28	GND	Ground	Ground
29	RESETB	Digital Input	Active Low Reset. Connect to VDD if not used.
30	VANA	Analog	Analog Core Decoupling, connect to a 100nF decoupling capacitor
31	CAP0	Analog	Capacitive Sensor 0
32	CAP1	Analog	Capacitive Sensor 1
bottom plate	GND	Ground	Exposed pad connect to ground

Table 1 Pin description

1.4 Simplified Block Diagram

The simplified block diagram of the SX8645 is illustrated in Figure 3.

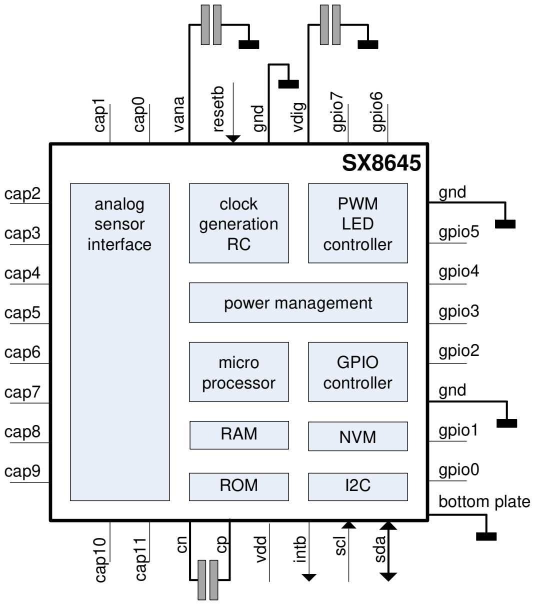


Figure 3 Simplified block diagram of the SX8645

1.5 Acronyms

ASI	Analog Sensor Interface
DCV	Digital Compensation Value
GPI	General Purpose Input
GPO	General Purpose Output
GPP	General Purpose PWM
MTP	Multiple Time Programmable
NVM	Non Volatile Memory
PWM	Pulse Width Modulation
QSM	Quick Start Memory
SPM	Shadow Parameter Memory

2 Electrical Characteristics

2.1 Absolute Maximum Ratings

Stresses above the values listed in “Absolute Maximum Ratings” may cause permanent damage to the device.

This is a stress rating only and functional operation of the device at these, or any other conditions beyond the “Recommended Operating Conditions”, is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Parameter	Symbol	Min.	Max.	Unit
Supply Voltage	VDD	-0.5	3.9	V
Input voltage (non-supply pins)	V_{IN}	-0.5	3.9	V
Input current (non-supply pins)	I_{IN}		10	mA
Operating Junction Temperature	T_{JCT}		125	°C
Reflow temperature	T_{RE}		260	°C
Storage temperature	T_{STOR}	-50	150	°C
ESD HBM (Human Body model)(i)	ESD_{HBM}	3		kV
Latchup(ii)	I_{LU}	± 100		mA

Table 2 Absolute Maximum Ratings

(i) Tested to JEDEC standard JESD22-A114

(ii) Tested to JEDEC standard JESD78

2.2 Recommended Operating Conditions

Parameter	Symbol	Min.	Max.	Unit
Supply Voltage	VDD	2.7V	3.6	V
Supply Voltage Drop(iii, iv, v)	VDD_{drop}		100	mV
Supply Voltage for NVM programming	VDD	3.0V	3.6	V
Ambient Temperature Range	T_A	-40	85	°C

Table 3 Recommended Operating Conditions

(iii) Performance for 2.6V < VDD < 2.7V might be degraded.

(iv) Operation is not guaranteed below 2.6V. Should VDD briefly drop below this minimum value, then the SX8645 may require;

- a hardware reset issued by the host using the RESETB pin
- a software reset issued by the host using the I2C interface

(v) In the event the host processor is reset or undergoes a power OFF/ON cycle, it is recommended that the host also resets the SX8645 and assures that parameters are re-written into the SPM (should these differ to the parameters held in NVM).

2.3 Thermal Characteristics

Parameter	Symbol	Min.	Max.	Unit
Thermal Resistance - Junction to Ambient (vi)	θ_{JA}		25	°C/W

Table 4 Thermal Characteristics

(vi) Static airflow

2.4 Electrical Specifications

All values are valid within the operating conditions unless otherwise specified.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Current consumption						
Active mode, average	$I_{OP,active}$	30ms scan period, 12 sensors enabled, minimum sensitivity		220	300	uA
Doze mode, average	$I_{OP,Doze}$	195ms scan period, 12 sensors enabled, minimum sensitivity		80	110	uA
Sleep	$I_{OP,sleep}$	I2C and GPI listening, sensors disabled		8	17	uA
GPIO, set as Input, RESETB, SCL, SDA						
Input logic high	V_{IH}		0.7*VDD		VDD + 0.3V	V
Input logic low	V_{IL}	VSS applied to GND pins	VSS - 0.3V		0.8	V
Input leakage current	L_I	CMOS input			±1	uA
Pull up resistor	R_{PU}	when enabled		660		kΩ
Pull down resistor	R_{PD}	when enabled		660		kΩ
GPIO set as Output, INTB, SDA						
Output logic high	V_{OH}	I_{OH} <4mA	VDD-0.4			V
Output logic low	V_{OL}	$I_{OL,GPIO}$<12mA $I_{OL,SDA,INTB}$<4mA			0.4	V
Start-up						
Power up time	t_{por}	time between rising edge VDD and rising INTB			150	ms
RESETB						
Pulse width	t_{res}		50			ns
External components						
Capacitor between VDIG, GND	C_{vdig}	type 0402, tolerance +/-50%		100		nF
Capacitor between VANA, GND	C_{vana}	type 0402, tolerance +/-50%		100		nF
Capacitor between CP, CN	C_{int}	type 0402, tolerance +/-10%		1		nF
Capacitor between VDD, GND	C_{vdd}	type 0402, tolerance +/-50%		100		nF

Table 5 Electrical Specifications

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
I2C Timing Specifications (i)						
SCL clock frequency	f_{SCL}				400	KHz
SCL low period	t_{LOW}		1.3			us
SCL high period	t_{HIGH}		0.6			us
Data setup time	$t_{SU;DAT}$		100			ns
Data hold time	$t_{HD;DAT}$		0			ns
Repeated start setup time	$t_{SU;STA}$		0.6			us
Start condition hold time	$t_{HD;STA}$		0.6			us
Stop condition setup time	$t_{SU;STO}$		0.6			us
Bus free time between stop and start	t_{BUF}		500			us
Input glitch suppression	t_{SP}				50	ns

Table 6 I2C Timing Specification

Notes:
(i) All timing specifications, Figure 4 and Figure 5, refer to voltage levels (V_{IL}, V_{IH}, V_{OL}) defined in Table 5.

The interface complies with slave F/S mode as described by NXP: “I2C-bus specification, Rev. 03 - 19 June 2007”

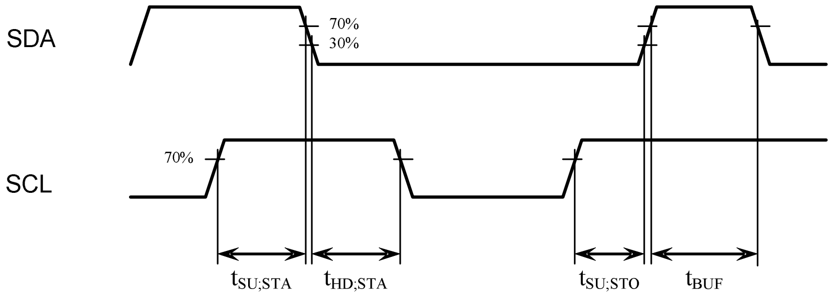


Figure 4 I2C Start and Stop timing

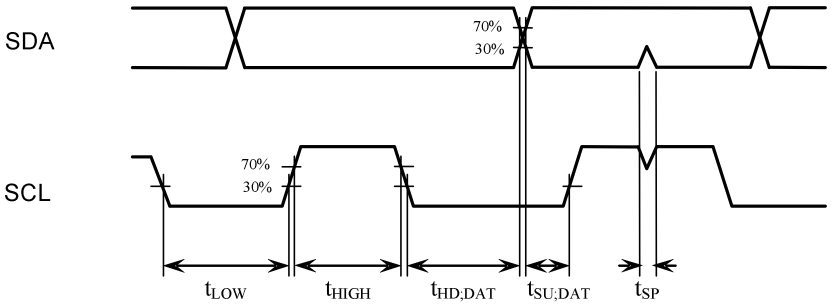


Figure 5 I2C Data timing

3 Functional Description

3.1 Quickstart Application

The SX8645 is preconfigured (Quickstart Application) for an application with 6 buttons, a wheel (consisting of 6 sensors) and 8 LED drivers using logarithmic PWM fading.

Implementing a schematic based on Figure 6 will be immediately operational after powering without programming the SX8645 (even without host).

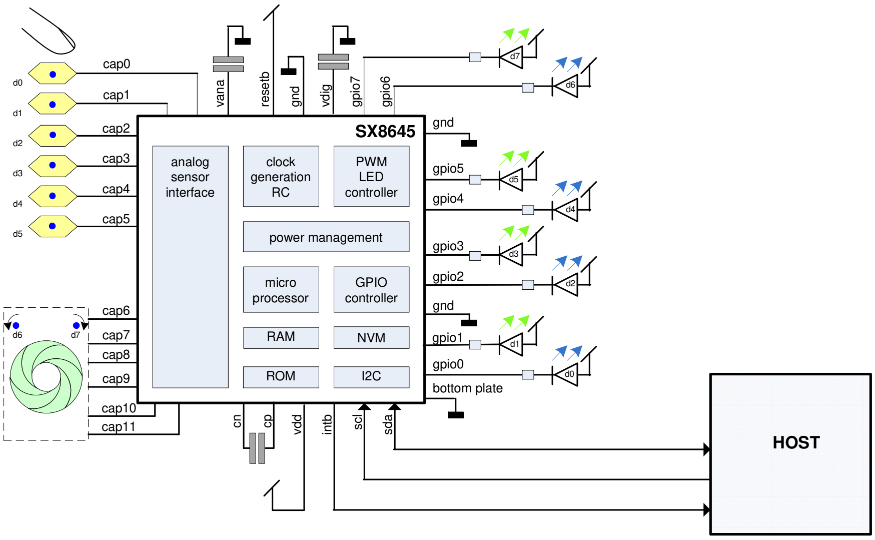


Figure 6 Quickstart Application

Touching the sensor on the CAP0 pin will enable automatically the LED connected to GPIO0. When the CAP0 sensor is released the LED on GPIO0 will slowly fade-out using smooth logarithmic fading.

All other sensors (CAP1 to CAP5) have their own LED associated on a GPIO pin showing a touch or a release.

The sensors on CAP6 to CAP11 are used in a wheel configuration. A moving finger on the wheel will enable the LED on GPIO6 or GPIO7 indicating the finger rotation direction.

The sensor detection and the LED fading described above are operational without any host interaction.

This is made possible using the SX8645 Autolight feature described in the following sections.

3.2 Introduction

3.2.1 General

The SX8645 is intended to be used in applications which require capacitive sensors covered by isolating overlay material. A finger approaching the capacitive sensors will change the charge that can be loaded on the sensors. The SX8645 measures the change of charge and converts that into digital values (ticks). The larger the charge on the sensors, the larger the number of ticks will be. The charge to ticks conversion is done by the SX8645 Analog Sensor Interface (ASI).
The ticks are further processed by the SX8645 and converted in a high level, easy to use information for the user's host.

The information between SX8645 and the user's host is passed through the I2C interface with an additional interrupt signal indicating that the SX8645 has new information. For buttons this information is simply touched or released.

3.2.2 GPIOs

A second path of feedback to the user is using General Purpose Input Output (GPIO) pins. The SX8645 offers eight individual configurable GPIO pins. The GPIO can e.g. be set as a LED driver which slowly fade-in when a finger touches a button and slowly fade-out when the button is released. Fading intensity variations can be logarithmic or linear. Interval speed and initial and final light intensity can be selected by the user. The fading is done using a 256 steps PWM. The SX8645 has eight individual PWM generators, one for each GPIO pin.
The LED fading can be initiated automatically by the SX8645 by setting the SX8645 Autolight feature. A simple touch on a sensor and the corresponding LED will fade-in without any host interaction over the I2C.
In case the Autolight feature is disabled then the host will decide to start a LED fading-in period, simply by setting the GP0 pin to 'high' using one I2C command. The SX8645 will then slowly fade-in the LED using the PWM autonomously.
In case the host needs to have full control of the LED intensity then the host can set the GPIO in GPP mode. The host is then able to set the PWM pulse width freely at the expense of an increased I2C occupation.
The GPIOs can be set further in the digital standard Input mode (GPI).

3.2.3 Parameters

The SX8645 has many low level built-in, fixed algorithms and procedures. To allow a lot of freedom for the user and adapt the SX8645 for different applications these algorithms and procedures can be configured with a large set of parameters which will be described in the following sections. Examples of parameters are which sensors are buttons or which sensors are parts of a wheel, which GPIO is used for outputs or LEDs and which GPIO is mapped to which button.
Sensitivity and detection thresholds of the sensors are part of these parameters. Assuming that overlay material and sensors areas are identical then the sensitivities and thresholds will be the same for each sensor. In case sensors are not of the same size then sensitivities or thresholds might be chosen individually per sensor.
So a smaller size sensor can have a larger sensitivity while a big size sensor may have the lower sensitivity.

3.2.4 Configuration

During a development phase the parameters can be determined and fine tuned by the users and downloaded over the I2C in a dynamic way. The parameter set can be downloaded over the I2C by the host each time the SX8645 boots up. This allows a flexible way of setting the parameters at the expense of I2C occupation.
In case the parameters are frozen they can be programmed in Multiple Time Programmable (MTP) Non Volatile Memory (NVM) on the SX8645. The programming needs to be done once (over the I2C). The SX8645 will then boot up from the NVM and additional parameters from the host are not required anymore.
In case the host desires to overwrite the boot-up NVM parameters (partly or even complete) this can be done by additional I2C communications.

3.3 Scan Period

The basic operation Scan period of the SX8645 sensing interface can be split into three periods over time.

In the first period (Sensing) the SX8645 is sensing all enabled CAP inputs, from CAP0 towards CAP11.

In the second period (Processing) the SX8645 processes the sensor data, verifies and updates the GPIO and I2C status registers.

In the third period (Timer) the SX8645 is set in a low power mode and waits until a new cycle starts.

Figure 7 shows the different SX8645 periods over time.

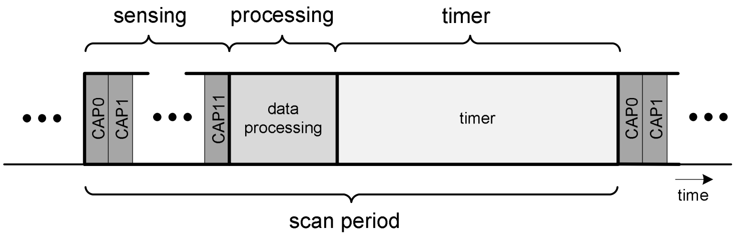


Figure 7 Scan Period

The scan period determines the minimum reaction time of the SX8645. The scan period can be configured by the host from 15ms to values larger than a second.
The reaction time is defined as the interval between a touch on the sensor and the moment that the SX8645 generates the interrupt on the INTB pin. The shorter the scan period the faster the reaction time will be.
Very low power consumption can be obtained by setting very long scan periods with the expense of having longer reaction times.

Important: All external events like GPIO, I2C and INTB are updated in the processing period, so once every scan period. If e.g. a GPI would change state directly after the processing period then this will be reported with a delay of one scan period later in time.

3.4 Operation modes

The SX8645 has 3 operation modes. The main difference is found in the reaction time (corresponding to the scan period) and power consumption.

Active mode offers fast scan periods. The typical reaction time is 30ms. All enabled sensors are scanned and information data is processed within this interval.

Doze mode increases the scan period time which increases the reaction time to 195ms typical and at the same time reduces the operating current.

Sleep mode turns the SX8645 OFF, except for the I2C and GPI peripheral, minimizing operating current while maintaining the power supplies. In Sleep mode the SX8645 does not do any sensor scanning.

The user can specify other scan periods for the Active and Doze mode and decide for other compromises between reaction time and power consumption.

In most applications the reaction time needs to be fast when fingers are present, but can be slow when no person uses the application. In case the SX8645 is not used for a specific time it can go from Active mode into Doze mode and power will be saved. This time-out is determined by the Passive Timer which can be configured by the user or turned OFF if not required.

To leave Doze mode and enter Active mode this can be done by a simple touch on any button.
For some applications a single button touch might cause undesired wakening up and Active mode would be entered too often.
The SX8645 offers therefore a smart wake-up sequence feature in which the user needs to touch and release a correct sequence of buttons before Active mode will be entered. This is explained in more detail in the Wake-Up Sequence section.

The host can decide to force the operating mode by issuing commands over the I2C (using register CompOpMode) and take fully control of the SX8645.

The diagram in Figure 8 shows the available operation modes and the possible transitions.

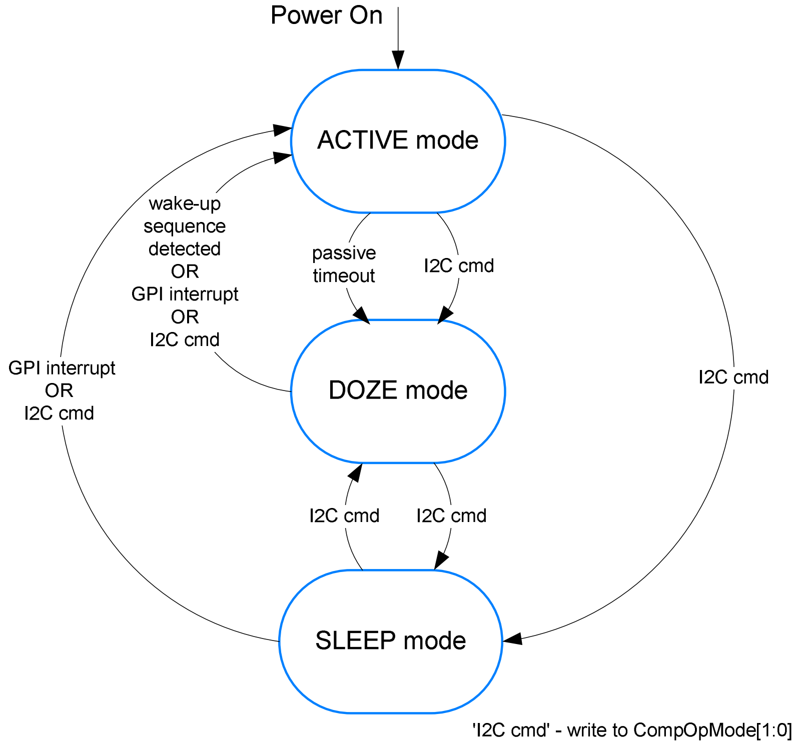


Figure 8 Operation modes

3.5 Sensors on the PCB

The capacitive sensors are relatively simple copper areas on the PCB connected to the twelve SX8645 capacitive sensor input pins (CAP0…CAP11).The sensors are covered by isolating overlay material (typically 1mm...3mm). The area of a sensor is typically one square centimeter which corresponds about to the area of a finger touching the overlay material.
The capacitive sensors can be setup as ON/OFF buttons (see example Figure 9) or arranged in a wheel configuration (see example Figure 10) for e.g. menu scrolling or volume control applications.

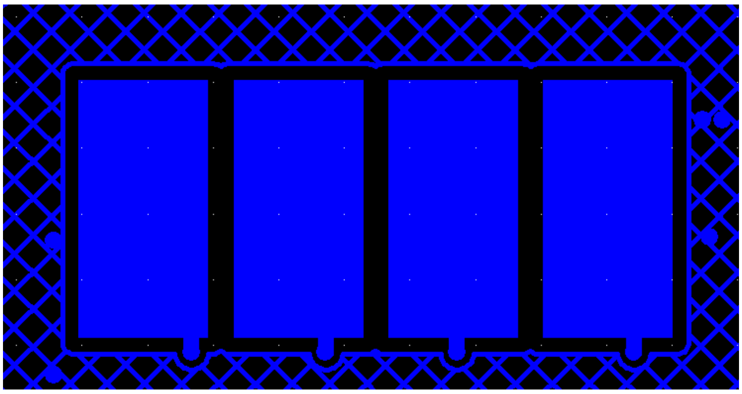


Figure 9 PCB top layer of four sensors for buttons (surrounded by a ground plane)

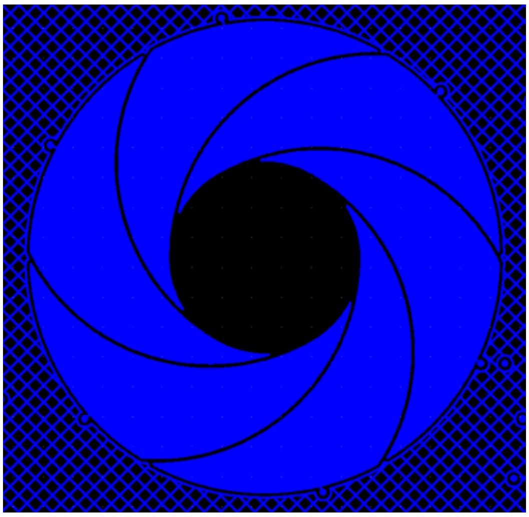


Figure 10 PCB top layer of one wheel using six sensors (surrounded by ground plane)

3.6 Button and Wheel Information

3.6.1 Button Information

The buttons have two simple states (see Figure 11): ON (touched by finger) and OFF (released and no finger press).

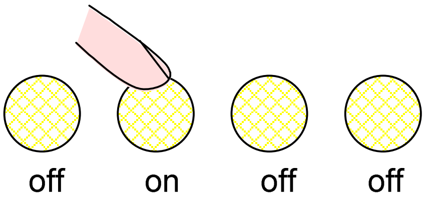


Figure 11 Buttons

A finger is detected as soon as the number of ticks from the ASI reaches a user-defined threshold plus a hysteresis.

A release is detected if the ticks from the ASI go below the threshold minus a hysteresis. The hysteresis around the threshold avoids rapid touch and release signaling during transients.

3.6.2 Wheel Information

In case sensors are arranged in a wheel configuration the ON, OFF information remains available as if it would be a single sensor button.

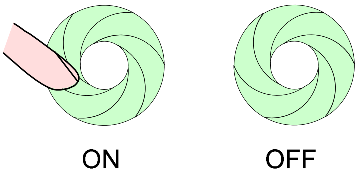


Figure 12 Wheel ON, OFF

Wherever the wheel is touched the information will be set to ON. If no finger is present the wheel information will be OFF.

Due to the 2 dimensional character of the wheel more information can be derived by processing the ticks.
During a touch a finger will influence most of the time the charge on one or two sensors but never all of the sensors at the same time. Some sensor ticks will be larger than others based on the finger position.
The processing algorithms can therefore determine where the finger is positioned on the wheel.
Interpolation between sensors increases the resolution beyond the number of sensors in the wheel.
The interpolation can be done already on the PCB sensor structures (analog, like the wheel in Figure 10) and as well by SX8645 digital processing of the ticks using center of gravity calculations.

The position of the finger on the PCB structures varies between the minimum zero and a user defined maximum (Figure 13).

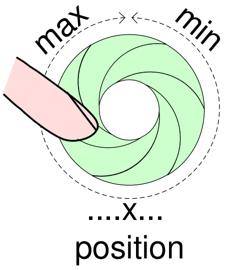


Figure 13 Wheel Position

The position belonging to the minimum and associated to a sensor is defined arbitrarily. The SX8645 defines the minimum position to the sensor with the lowest CAP pin index. E.g. if CAP0 is a button (or disabled) and CAP1 to CAP7 are the sensors of the wheel then the position 'zero' starts at CAP1 and the maximum is found at CAP7.

In addition to the wheel position, the SX8645 allows to detect finger rotation. The rotation occurs if the finger position changes a certain step size between two succeeding scan periods. A very slow moving finger will not be considered as a rotation as the changing position will be minor. The SX8645 allows detecting a rotate clockwise (direction min to max) (see Figure 14) and a rotate counter clockwise (direction max to min) (see Figure 15).

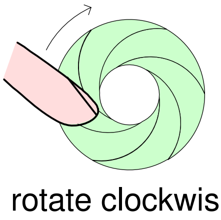


Figure 14 Wheel rotate clockwise

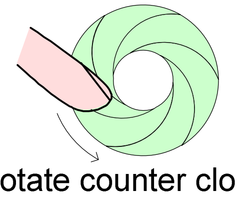


Figure 15 Wheel rotate counter clockwise

3.7 Analog Sensing Interface

The Analog Sensing Interface (ASI) converts the charge on the sensors into ticks which will be further digitally processed. The basic principle of the ASI will be explained in this section.

The ASI consists of a multiplexer selecting the sensor, analog switches, a reference voltage, an ADC sigma delta converter, an offset compensation DAC and an external integration capacitor (see Figure 16).

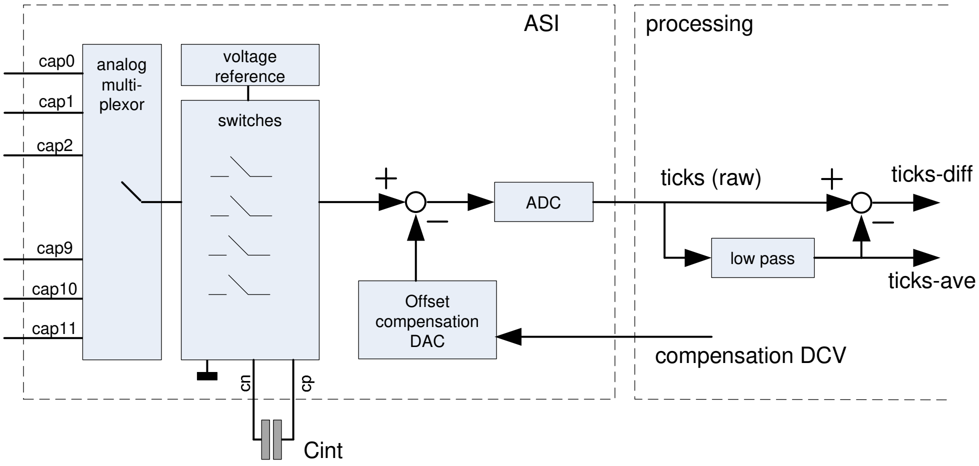


Figure 16 Analog Sensor Interface

To get the ticks representing the charge on a specific sensor the ASI will execute several steps.

The charge on a sensor cap (e.g. CAP0) will be accumulated multiple times on the external integration capacitor, Cint.

This results in an increasing voltage on Cint proportional to the capacitance on CAP0.

At this stage the offset compensation DAC is enabled. The compensation DAC generates a voltage proportional to an estimation of the external capacitance. The estimation is obtained by the offset compensation procedure executed e.g. at power-up.

The difference between the DAC output and the charge on Cint is the desired signal. In the ideal case the difference of charge will be converted to zero ticks if no finger is present and the number of ticks becomes high in case a finger is present.

The difference of charge on Cint and the DAC output will be transferred to the ADC (Sigma Delta Integrator).

After the charge transfer to the ADC the steps above will be repeated.

The larger the number the cycles are repeated the larger the signal out of the ADC with improved SNR. The sensitivity is therefore directly related to the number of cycles.

The SX8645 allows setting the sensitivity for each sensor individually in applications which have a variety of sensors sizes or different overlays or for fine-tuning performances. The optimal sensitivity is depending heavily on the final application. If the sensitivity is too low the ticks will not pass the thresholds and it is not possible to detect fingers. In case the sensitivity is set too large a finger hovering above the sensors will already be detected before the finger really touches the overlay resulting in false detections.

Once the ASI has finished the first sensor, the ticks are stored and the ASI will start measuring the next sensor until all (enabled) sensors pins have been treated.

In case some sensors are disabled then these result in lower power consumption simply because the ASI is active for a shorter period and the following processing period will be shorter.

The ticks from the ASI will then be handled by the digital processing.

3.8 Offset Compensation

The capacitance at the CAP pins is determined by an intrinsic capacitance of the integrated circuit, the PCB traces, ground coupling and the sensor planes. This capacitance is relatively large and might become easily some tens of pF. This parasitic capacitance will vary only slowly over time due to environmental changes.

A finger touch is in the order of one pF. If the finger approaches the sensor this occurs typically fast.

The ASI has the difficult task to detect and distinguish a small, fast changing capacitance, from a large, slow varying capacitance. This would require a very precise, high resolution ADC and complicated, power consuming, digital processing.

The SX8645 features a 16 bit DAC which compensates for the large, slow varying capacitance already in front of the ADC. In other words the ADC converts only the desired small signal. In the ideal world the ADC will put out zero ticks even if the external capacitance is as high as 100pF.

At each power-up of the SX8645 the Digital Compensation Values (DCV) are estimated by the digital processing algorithms. The algorithm will adjust the compensation values such that zero ticks will be generated by the ADC.

Once the correct compensation values are found these will be stored and used to compensate each CAP pin.

If the SX8645 is shut down the compensation values will be lost. At a next power-up the procedure starts all over again. This assures that the SX8645 will operate under any condition. Powering up at e.g. different temperatures will not change the performance of the SX8645 and the host does not have to do anything special.

The DCVs do not need to be updated if the external conditions remain stable.

However if e.g. temperature changes this will influence the external capacitance. The ADC ticks will drift then slowly around zero values basically because of the mismatch of the compensation circuitry and the external capacitance.

In case the average value of the ticks become higher than the positive noise threshold (configurable by user) or lower than the negative threshold (configurable by user) then the SX8645 will initiate a compensation procedure and find a new set of DCVs.

Compensation procedures can as well be initiated by the SX8645 on periodic intervals. Even if the ticks remain within the positive and negative noise thresholds the compensation procedure will then estimate new sets of DCVs.

Finally the host can initiate a compensation procedure by using the I2C interface (in Active or Doze mode). This is e.g. required after the host changed the sensitivity of sensors.

3.9 Processing

The first processing step of the raw ticks, coming out of the ASI, is low pass filtering to obtain an estimation of the average capacitance: tick-ave (see Figure 17).
This slowly varying average is important in the detection of slowly changing environmental changes.

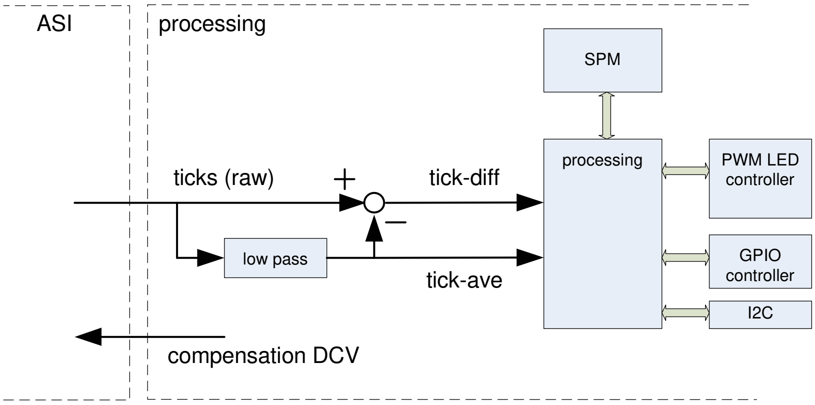


Figure 17 Processing

The difference of the tick average and the raw ticks, tick-diff, is a good estimation of rapid changing input capacitances.

The tick-diff, tick-ave and the configuration parameters in the SPM are then processed and determines the sensor information, I2C registers status and PWM control.

3.10 Configuration

Figure 18 shows the building blocks used for configuring the SX8645.

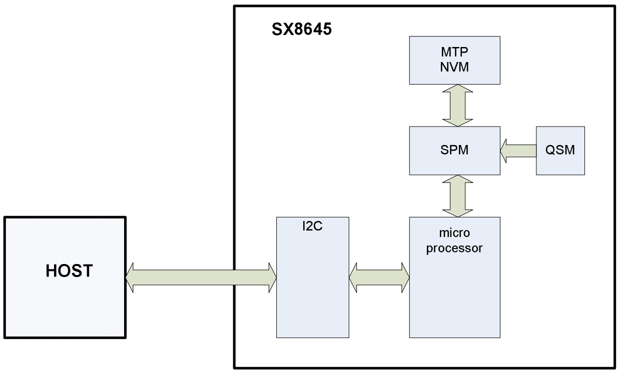


Figure 18 Configuration

The default configuration parameters of the SX8645 are stored in the Quick Start Memory (QSM). This configuration data is setup to a very common application for the SX8645 with 6 buttons and a wheel. Without any programming or host interaction the SX8645 will startup in the Quick Start Application.

The QSM settings are fixed and can not be changed by the user.

In case the application needs different settings than the QSM settings then the SX8645 can be setup and/or programmed over the I2C interface.

The configuration parameters of the SX8645 can be stored in the Multiple Time Programmable (MTP) Non Volatile Memory (NVM). The NVM contains all those parameters that are defined and stable for the application. Examples are the number of sensors enabled, sensitivity, active and Doze scan period. The details of these parameters are described in the next chapters.

At power up the SX8645 checks if the NVM contains valid data. In that case the configuration parameter source becomes the NVM. If the NVM is empty or non-valid then the configuration source becomes the QSM. In the next step the SX8645 copies the configuration parameter source (QSM or NVM) into the Shadow Parameter Memory (SPM). The SX8645 is operational and uses the configuration parameters of the SPM.

During power down or reset event the SPM loses all content. It will automatically be reloaded (from QSM or NVM) following power up or at the end of the reset event.

The host will interface with the SX8645 through the I2C bus.

The I2C of the SX8645 consists of 16 registers. Some of these I2C registers are used to read the status and information of the button and the wheel. Other I2C registers allow the host to take control of the SX8645. The host can e.g. decide to change the operation mode from Active mode to Doze mode or go into Sleep (according to Figure 8).

Two additional modes allow the host to have an access to the SPM or indirect access to the NVM.

These modes are required during development, can be used in real time or in-field programming.

Figure 19 shows the Host SPM mode. In this mode the host can decide to overwrite the SPM. This is useful during the development phases of the application where the configuration parameters are not yet fully defined and as well during the operation of the application if some parameters need to be changed dynamically.

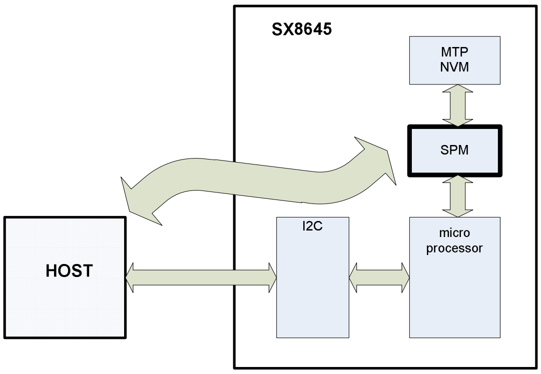


Figure 19 Host SPM mode

The content of the SPM remains valid as long as the SX8645 is powered and no reset is performed. After a power down or reset the host needs to re-write the SPM if relevant for the application.

Figure 20 shows the Host NVM mode. In this mode the host will be able to write the NVM.

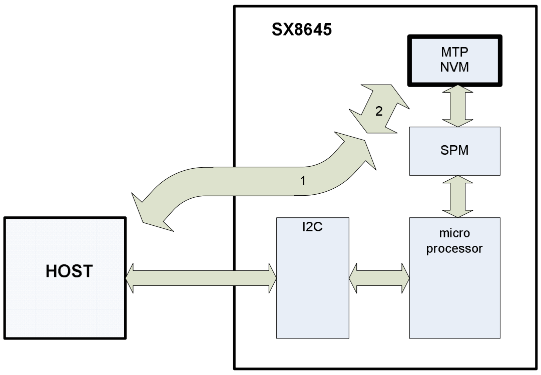


Figure 20 Host NVM mode

The writing of the host towards the NVM is not done directly but done in 2 steps (Figure 20).

In the first step the host writes to the SPM (as in Figure 19). In the second step the host signals the SX8645 to copy the SPM content into the NVM.

Initially the NVM memory is empty and it is required to determine a valid parameter set for the application. This can be done during the development phase using dedicated evaluation hardware representing the final application. This development phase uses probably initially the host SPM mode which allows faster iterations.

Once the parameter set is determined this can be written to the NVM over the I2C using the 2 steps approach by the host or a dedicated programmer for large volumes production (as described in the paragraphs 6.6 and 6.7).

3.11 Power Management

The SX8645 uses on-chip voltage regulators which are controlled by the on-chip microprocessor. The regulators need to be stabilized with an external capacitor between VANA and ground and between VDIG and ground (see Table 5). Both regulators are designed to only drive the SX8645 internal circuitry and must not be loaded externally.

3.12 Clock Circuitry

The SX8645 has its own internal clock generation circuitry that does not require any external components. The clock circuitry is optimized for low power operation and is controlled by the on-chip microprocessor. The typical operating frequency of the oscillating core is 16.7MHz from which all other lower frequencies are derived.

3.13 I2C interface

The I2C interface allows the communication between the host and the SX8645.

The I2C slave implemented on the SX8645 is compliant with the standard (100kb/s) and fast mode (400kb/s)
The default SX8645 I2C address equals 0b010 1011.
A different I2C address can be programmed by the user in the NVM.

3.14 Reset

The reset can be performed by 3 sources:

- power up,
- RESETB pin,
- software reset.

3.14.1 Power up

During power up the INTB is kept low. Once the power up sequence is terminated the INTB is released autonomously. The SX8645 is then ready for operation.

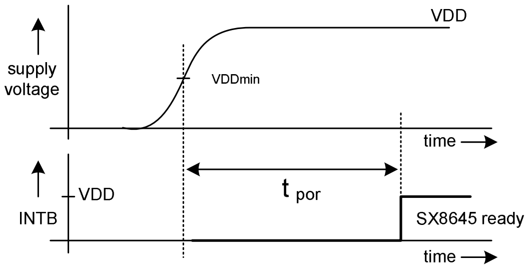


Figure 21 Power Up vs. INTB

During the power on period the SX8645 stabilizes the internal regulators, RC clocks and the firmware initializes all registers.

During the power up the SX8645 is not accessible and I2C communications are forbidden.
As soon as the INTB rises the SX8645 will be ready for I2C communication.

3.14.2 RESETB

When RESETB is driven low the SX8645 will reset and start the power up sequence as soon as RESETB is driven high or pulled high.

In case the user does not require a hardware reset control pin then the RESETB pin can be connected to VDD.

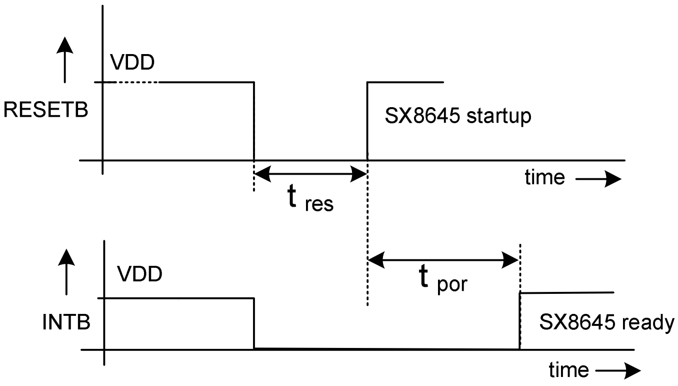


Figure 22 Hardware Reset

3.14.3 Software Reset

To perform a software reset the host needs to write 0xDE followed by 0x00 at the SoftReset register at address 0xB1.

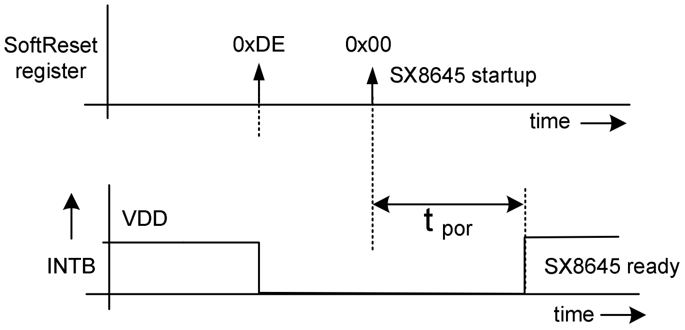


Figure 23 Software Reset

3.15 Interrupt

3.15.1 Power up

During power up the INTB is kept low. Once the power up sequence is terminated the INTB is released autonomously. The SX8645 is then ready for operation.

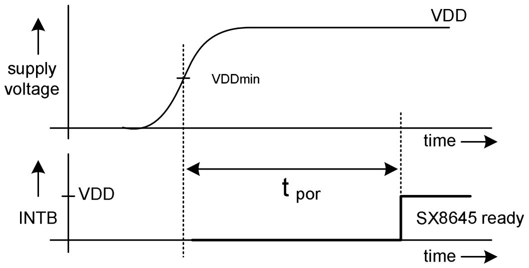


Figure 24 Power Up vs. INTB

During the power on period the SX8645 stabilizes the internal regulators, RC clocks and the firmware initializes all registers.

During the power up the SX8645 is not accessible and I2C communications are forbidden.
As soon as the INTB rises the SX8645 will be ready for I2C communication.

3.15.2 Assertion

INTB is updated in Active or Doze mode once every scan period.

The INTB will be asserted: at the following events:

- if a Button event occurred (touch or release if enabled). I2C registers CapStatMsb and CapStatLsb show the detailed status of the Buttons,
- if a Wheel event occurred (touch, release, rotate clockwise, rotate counter clockwise or position change). I2C registers CapStatMsb, WhlPosMsb and WhlPosLsb show the detailed status of the Wheel,
- if a GPI edge occurred (rising or falling if enabled). I2C register GpiStat shows the detailed status of the GPI pins,
- when actually entering Active or Doze mode either through automatic wakeup or via host request (may be delayed by 1 scan period). I2C register CompOpmode shows the current operation mode,
- once compensation procedure is completed either through automatic trigger or via host request (may be delayed by 1 scan period),
- once SPM write is effective (may be delayed by 1 scan period),
- once NVM burn procedure is completed (may be delayed by 1 scan period),
- during reset (power up, hardware RESETB, software reset).

3.15.3 Clearing

INTB is updated in Active or Doze mode once every scan period.

The clearing of the INTB is done as soon as the host performs a read to the IrqSrc I2C register or reset is completed

3.15.4 Example

A typical example of the assertion and clearing of the INTB and the I2C communication is shown in Figure 25.

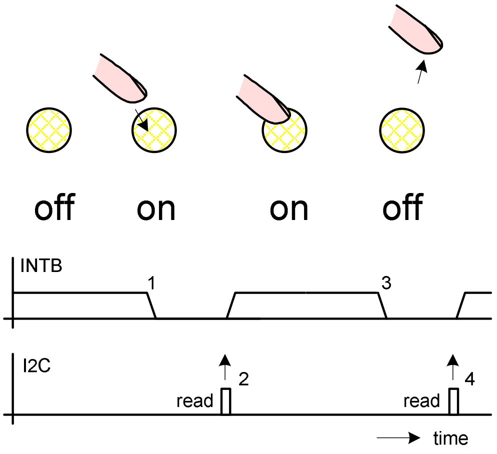


Figure 25 Interrupt and I2C

When a button is touched the SX8645 will assert the interrupt (1). The host will read the IrqSrc information over the I2C and this clears the interrupt (2).

If the finger releases the button the interrupt will be asserted (3). The host reading the IrqSrc information will clear the interrupt (4).

In case the host does not react to an interrupt this results in a missing touch.

3.16 General Purpose Input and Outputs

3.16.1 Introduction and Definitions

The SX8645 offers eight General Purpose Input and Outputs (GPIO) pins which can be configured in any of these modes:

- GPI (General Purpose Input)
- GPP (General Purpose PWM)
- GPO (General Purpose Output)

Each of these modes is described in more details in the following sections.

The polarity of the GPP and GPO pins is defined as in figure below, driving an LED as example. It has to be set accordingly in SPM parameter GpioPolarity.

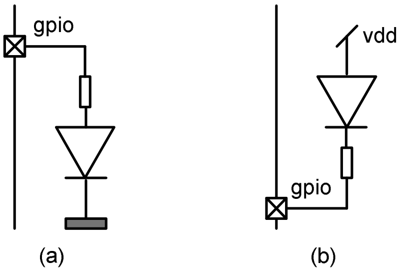


Figure 26 Polarity definition, (a) normal, (b) inverted