Smart glasses HMI with PSOC™ 4 CAPSENSE™

About this document

Scope and purpose

This application note explains how to use PSOC™ 4 MCUs with the fifth-generation CAPSENSE™ technology and capacitive sensors to implement the Human-Machine Interface (HMI) for smart glasses.

Intended audience

This document is intended primarily for developers, engineers, and designers who are interested in implementing HMI solutions for smart glasses using Infineon CAPSENSE™ technology.

***Note**: This document is for advanced users of CAPSENSE™ technology who are familiar with PSOC™4 devices, CAPSENSE™ technology, and CAPSENSE™ middleware library. If you are new to CAPSENSE™ technology, see the AN64846 – Getting started with CAPSENSE™ application note. For more information on CAPSENSE™, see the AN85951 – PSOC™ 4 and PSOC™ 6 MCU CAPSENSE™ design guide.*

Table of contents

About this document . . . 1
Table of contents . . . 2
1 Introduction . . . 3
2 Hardware design guidelines . . . 4
2.1 General CAPSENSE™ sensor construction and stackup . . . 4
2.2 Narrow touchpad design considerations and guidelines . . . 5
2.2.1 Design considerations and formulas . . . 9
2.3 Example designs for sensor . . . 10
2.3.1 Smart glass stackup . . . 10
2.3.2 FPC/FR4 stackup . . . 11
2.3.3 Narrow touchpad patterns . . . 11
3 Firmware design considerations . . . 13
3.1 Sensing method . . . 13
3.2 Low-power aspects . . . 13
3.3 PSOC™ 4 firmware considerations . . . 13
3.4 Host driver considerations . . . 14
4 External factors affecting the system . . . 15
4.1 Noise effect . . . 15
4.2 Battery-powered applications . . . 15
References . . . 17
Revision history . . . 18
Trademarks . . . 19
Disclaimer . . . 20

1 Introduction

Developing HMI solutions for wearable applications, such as smart glasses, smart rings, and fitness bands poses a unique challenge because they need to be small, light weight, and power efficient. Touch HMI has been the go-to choice for such applications because of the ability to create designs that are aesthetically appealing, low profile, low cost, low power, robust, and easily integrated into existing user interfaces.

To meet these design challenges, PSOC™ 4 MCU family with fifth-generation CAPSENSE™ and multi-sense technology offers an ultra-low-power touch HMI solution based on an integrated “Always-on” sensing technology. The multi-sense converter (MSC) with best-in-class performance enables modern, sleek user interface solutions with superior liquid tolerance, and provides a robust and reliable touch HMI solution for harsh environments.

This application note is focused on the design of touch HMI for smart glasses. This category of products are generally thin and narrow, which does not leave space for bulky components like physical buttons. A narrow touchpad is the ideal solution for such applications as they can be integrated into the frame and can enable user interactions, such as taps, double taps, swipes, and so on. For smart glasses that leverage Infineon's PSOC™ 4 with CAPSENSE™, it is crucial to follow the design considerations and recommendations discussed in this document. This approach ensures optimal tuning, superior performance, and the best possible signal-to-noise ratio (SNR).

To aid the development of touch HMI for smart glasses, this application note will discuss the following topics:

- Hardware design guidelines for narrow touchpads
- External factors affecting the system
- Firmware development and power optimization
- Linux host driver implementation

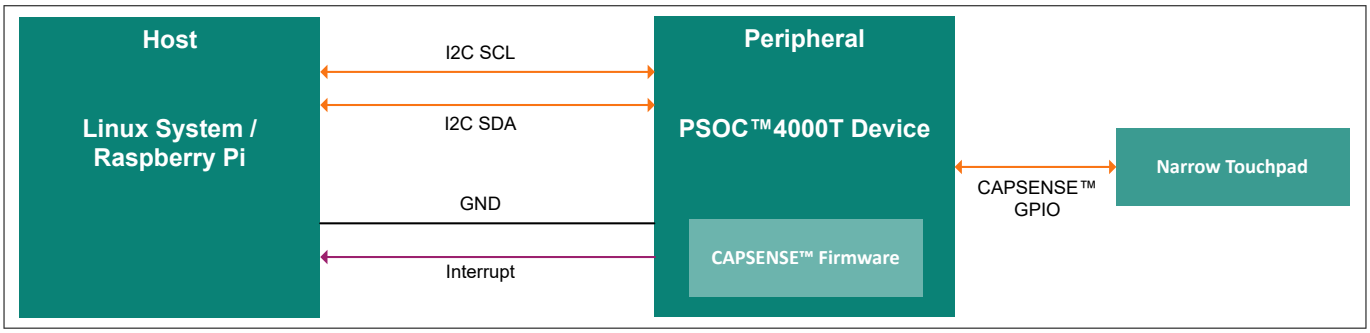


Figure 1 Smart glass narrow touchpad block diagram

2 Hardware design guidelines

This section explains the hardware design considerations and guidelines for Infineon CAPSENSE™.

2.1 General CAPSENSE™ sensor construction and stackup

A capacitive sensor can be constructed using different materials depending on the application requirement. In a typical sensor construction, a conductive pad is connected to the pin of the PSOC™ 4 MCU using a conductive trace. This whole arrangement is placed below a non-conductive overlay material and the user interacts on top of the overlay. Figure 2 shows the most common CAPSENSE™ sensor construction.

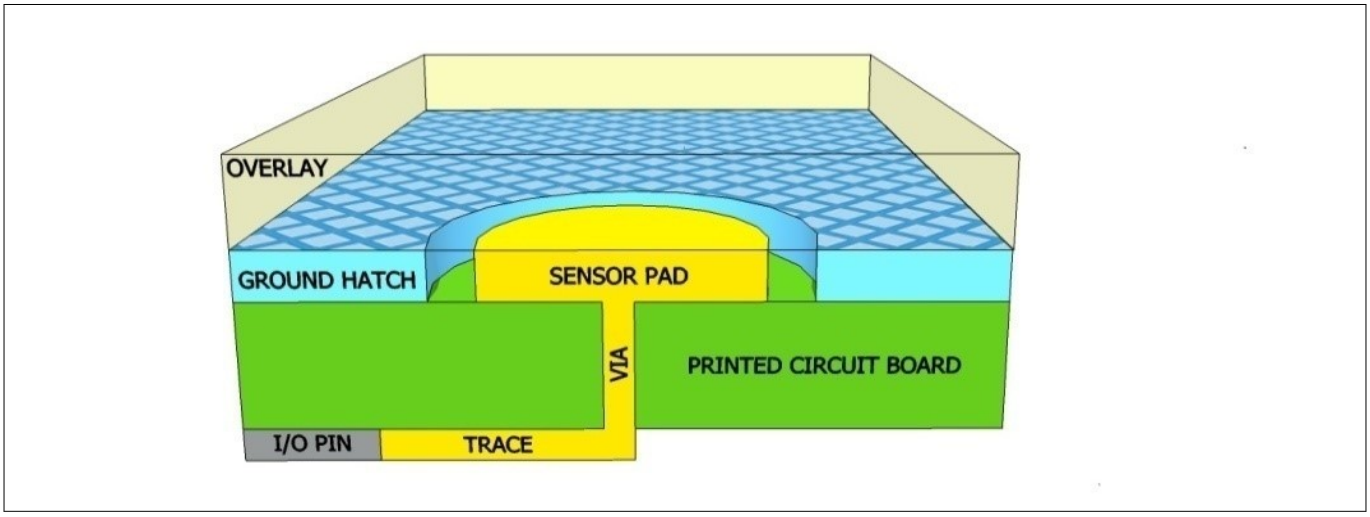


Figure 2 CAPSENSE™ sensor construction

The copper pad etched on the surface of the PCB acts as a CAPSENSE™ sensor. A non-conductive overlay serves as the touch surface. The overlay also protects the sensor from the environment and prevents direct finger contact. A ground hatch surrounding the sensor pad isolates the sensor from other sensors and PCB traces.

The simplest CAPSENSE™ PCB design is a two-layer board with sensor pads and hatched ground plane on the top, and the electrical components on the bottom. Figure 3 shows an exploded view of the CAPSENSE™ hardware and Figure 4 shows the typical PCB stackup. Sensors can also be constructed using materials other than copper, such as Indium-Tin-Oxide (ITO), or printed ink on substrates such as glass or a flex PCB.

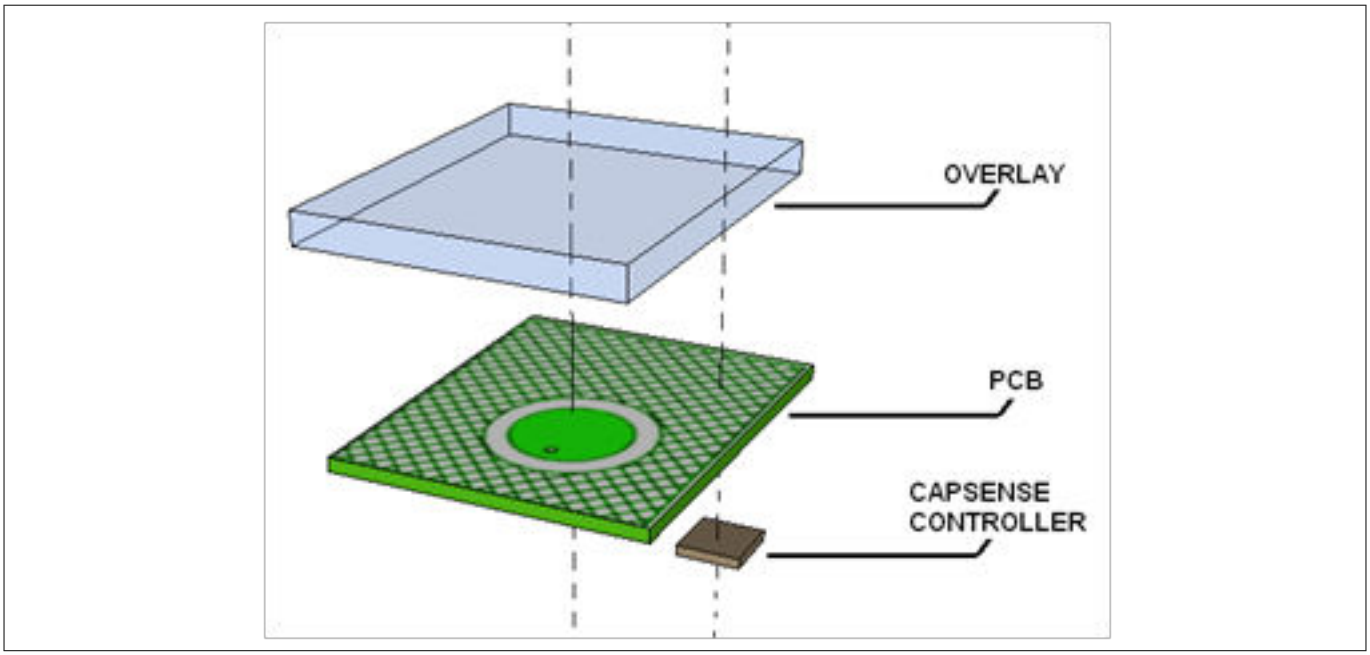


Figure 3 CAPSENSE™ hardware overview

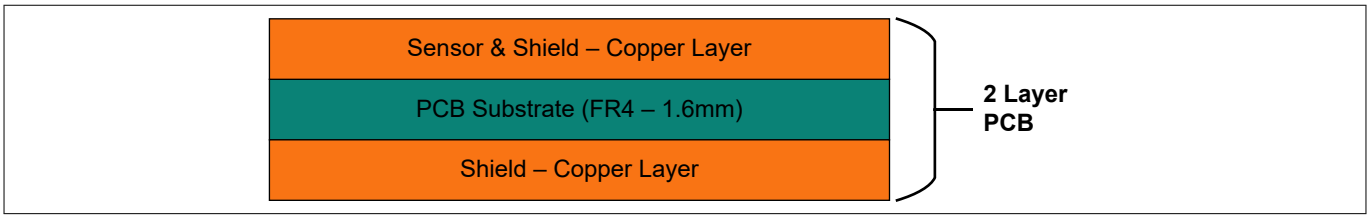


Figure 4 General CAPSENSE™ PCB stackup - 2 layer

2.2 Narrow touchpad design considerations and guidelines

As the touchpad HMI for smart glasses needs to be accommodated on the frame as shown in Figure 5, it needs some special considerations to be made in the hardware and firmware design. This section will discuss the hardware design details. For general touchpad design guidelines, see the AN234185 – PSOC™ 4 CAPSENSE™ touchpad design guide.

Touchpads can be designed using various sensor patterns, such as diamond, manhattan, interlace, etc. This document will discuss a touchpad design, which uses an interlace sensor pattern. A narrow touchpad will require a minimum of two rows. The number of columns is determined by the required length of the touchpad.

2 Hardware design guidelines

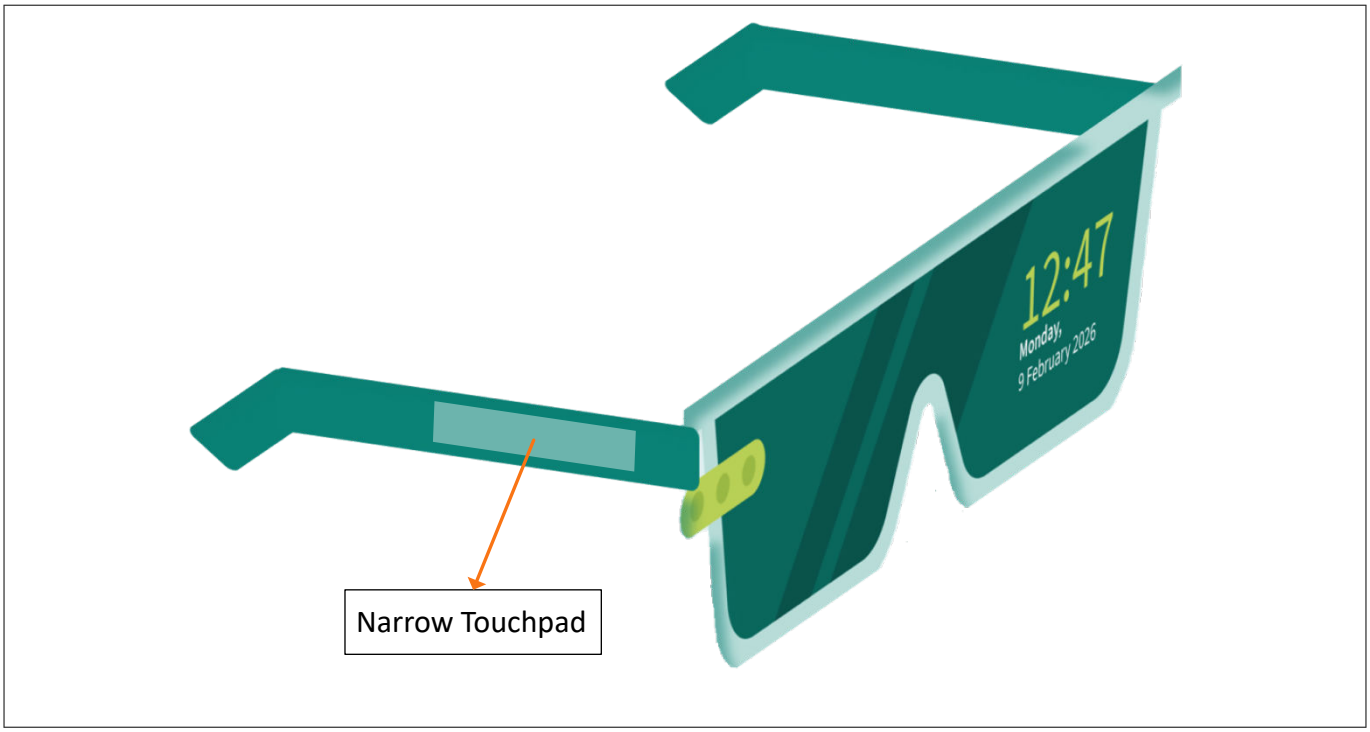


Figure 5 Smart glass

Figure 6 and Figure 7 illustrates the key design terms essential for understanding the design considerations and for developing the HMI design presented in this document. The following design considerations outlined specifically apply to touchpad designs featuring an interlaced sensor pattern. Detailed guidelines are mentioned in Design considerations and formulas.

***Note**: These guidelines are applicable only for 2xN touchpad designs where two is Rx (row) and N is Tx (column).*

2 Hardware design guidelines

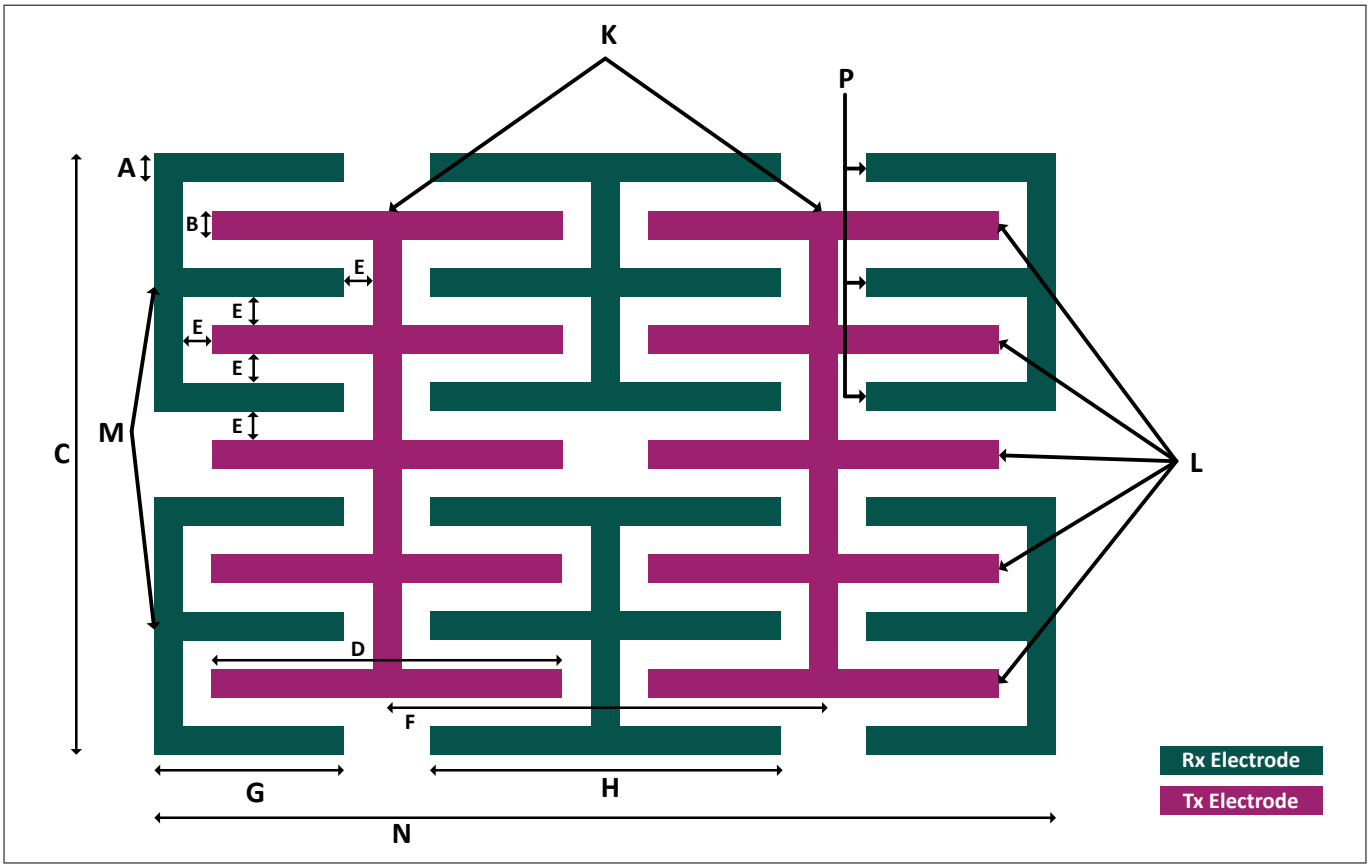


Figure 6 **Smart glass HMI design terms**

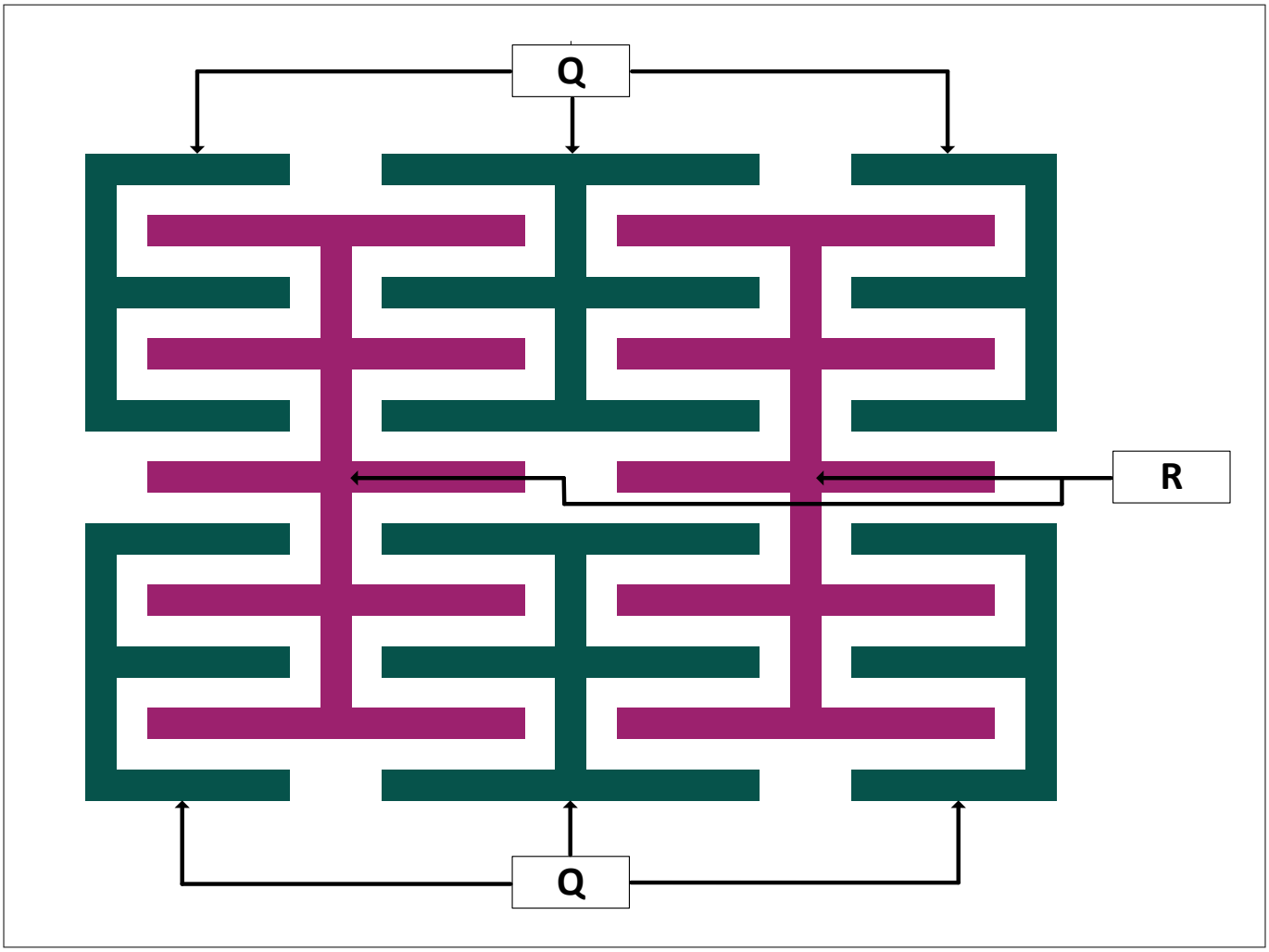


Figure 7 **Smart glass HMI design terms columns**

Table 1 **Smart glass HMI design terms details**

Design term	Description
A	Rx width
B	Tx width
C	Touchpad height
D	Rx prong length
E	Spacing
F	Rx prong pitch
G	Edge Tx prong length
H	Tx prong length
K	No. of Rx electrodes
L	No. of Rx prongs
M	No. of Tx electrodes
N	Touchpad length

(table continues...)

Table 1 (continued) Smart glass HMI design terms details

Design term	Description
P	No. of Tx prongs
Q	No. of Tx columns
R	No. of Rx columns

It is recommended to keep the Tx electrodes on the perimeter of the sensor as shown in Figure 8. Once Tx electrode patterns are formed, they need to be stitched. The central prongs must be used to stitch the Tx electrodes, which is shown in Figure 8.

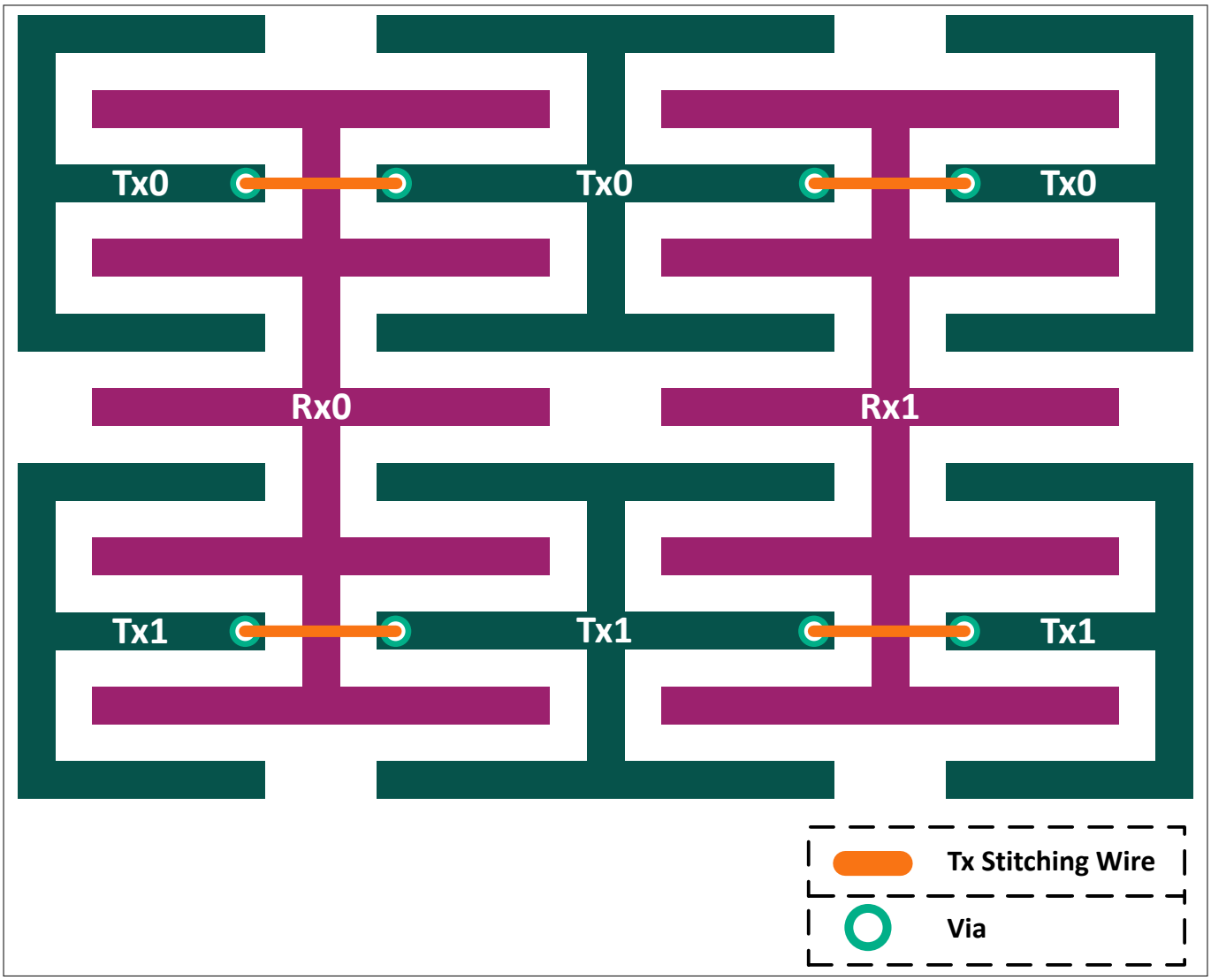


Figure 8 Smart glass HMI - Tx stitching

2.2.1 Design considerations and formulas

You must follow the design considerations for smart glass HMI listed in Table 2.

Table 2 Smart glass HMI design considerations and formulas

Category	Design consideration/formula
Spacing	$Spacing(E) \geq 0.25mm$
	$No.\ of\ spacing = 2 \times No.\ of\ Rx\ Prong(L)$
Tx electrodes	$No.\ of\ Tx\ Prong(P) = No.\ of\ Rx\ Prong(L) + 1$
	$No.\ of\ Tx\ Column(Q) = No.\ of\ Rx\ Column(R) + 1$
Rx electrodes	Number of Rx prongs should be an odd number
	$Rx\ Width(B) \geq Tx\ Width(A) \geq 0.25mm$
	$No.\ of\ Rx\ Column(R) = No.\ of\ Rx\ Pins(K)$
	$Rx\ Prong\ Pitch(F) = Rx\ Prong\ Length(D) + (2 \times Spacing(E)) + Tx\ Width(A)$
	$Rx\ Prong\ Length(D) = (Touchpad\ Length(N) - (Tx\ Width(A) \times No.\ of\ Tx\ Column(Q)) - (Spacing \times 2 \times No.\ of\ Rx\ Column(R))) / No.\ of\ Rx\ Column(R)$
Touchpad dimension	$Touchpad\ Height(C) = (No.\ of\ Tx\ Prong(P) \times Tx\ Width(A)) + (No.\ of\ Rx\ Prong(L) \times Rx\ Width(B)) + (No.\ of\ spacing \times Spacing(E))$

2.3 Example designs for sensor

CAPSENSE™ touchpads offer significant flexibility, allowing them to be integrated into a variety of form factors based on the specific requirements of the end application.

The narrow touchpad design supports a range of standard finger sizes, including 6 mm, 8 mm, and 10 mm fingers, allowing for accurate touch detection and gesture recognition. The touchpad can detect multiple fingers and recognize various gestures, including single-touch detection, multi-touch detection (up to two fingers), and gesture recognition (e.g., swipe, tap, and pinch). Multi-finger detection is only supported by the CSX touchpad configuration. The supported gestures include single-click, double-click, swipe (up, down, right, left), touch down, and lift off.

2.3.1 Smart glass stackup

The general stackup for smart glass applications consists of overlay, FPCs, and shielding components. The stackup consists of two outer overlay layers, facing outwards, which provide protection and aesthetic appeal. The FPC with the touchpad is attached to the overlay using adhesive tape. The other side of the PCB is connected to the battery using electromagnetic shielding film tape, ensuring electromagnetic interference (EMI) protection and noise reduction. The other surface of the battery is connected to the overlay's inner surface, also using shielded tape or black shielded tape. Figure 9 shows the overall stackup details.

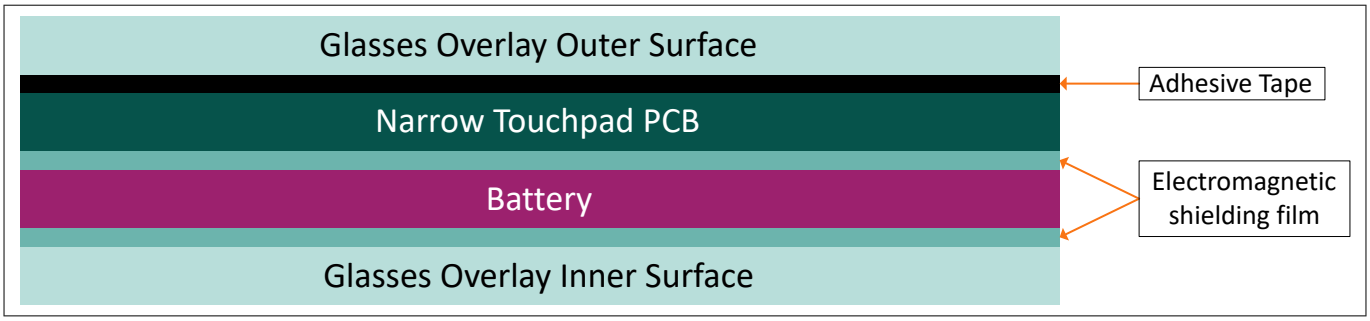


Figure 9 Overall stackup

2.3.2 FPC/FR4 stackup

For space-constrained designs, FPCs are commonly used. For narrow touchpad designs, a four-layer stackup is recommended. Figure 10 shows the layer details for this stackup.

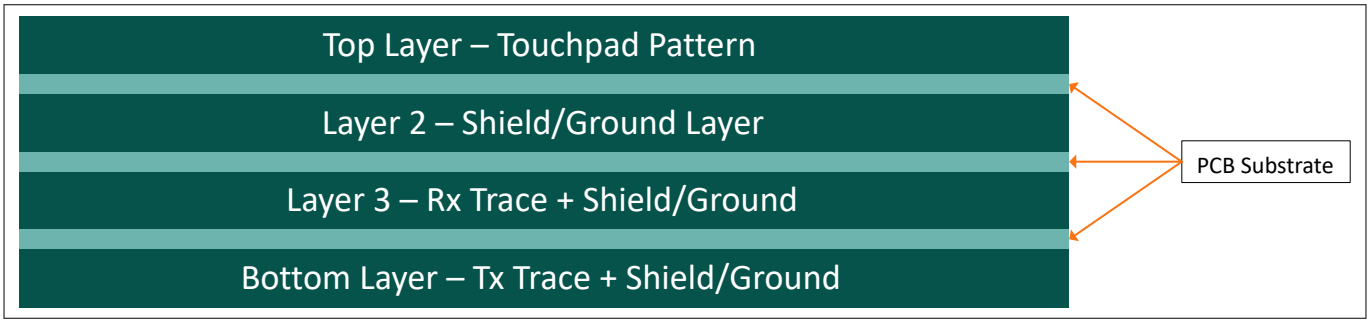


Figure 10 FPC/FR4 stackup

2.3.3 Narrow touchpad patterns

In this section, two examples are shown that would be suitable for smart glasses. These examples use the interlace sensor pattern but a similar touchpad can be created using diamond sensor patterns as well.

- **Rectangular touchpad:** Figure 11 shows a narrow rectangular touchpad. This is a 2X10 touchpad, which contains 2 Rx rows and 10 Tx columns. The dimensions of this touchpad are 50 mm width and 10 mm height

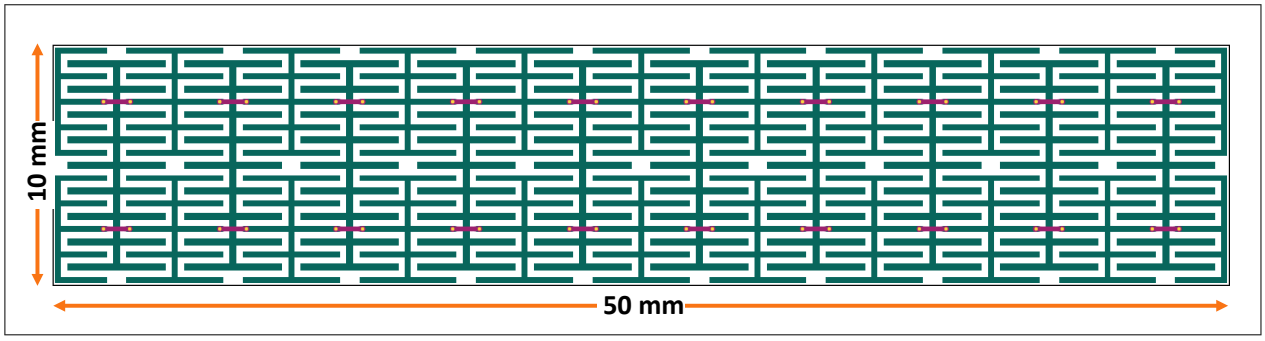


Figure 11 Straight touchpad

- **Trapezoidal touchpad:** Figure 12 shows a narrow trapezoidal touchpad. This is a 2X10 touchpad which contains 2 Rx rows and 10 Tx columns. The dimensions of this touchpad are 50 mm width with 10 mm height on the right and 8 mm height on the left

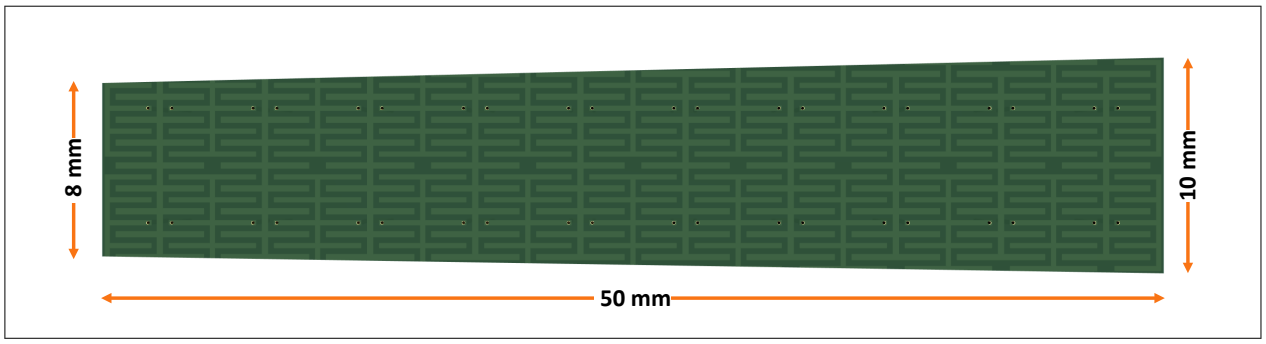


Figure 12 Trapezoid touchpad

See the Rectangular touchpad design DXF file for reference.

3 Firmware design considerations

This section assumes that the user is familiar with PSOC™ 4 devices, CAPSENSE™ technology, CAPSENSE™ middleware library, and software tools, including the CAPSENSE™ Configurator and Tuner. For more information, see the References section.

The development of firmware for a low-power CSX touchpad is a critical aspect of the smart glass system. The PSOC™ 4 device is used as a peripheral device, responsible for processing touch data and reporting it to the host. To enable communication between the host and peripheral, I2C is used along with an interrupt feature. The peripheral generates an interrupt on a GPIO, which is sensed by the host and processed for user applications. Additionally, a host driver is required to manage the communication between the host and the PSOC™ 4 peripheral, ensuring seamless data transfer and processing.

For detailed information on developing the PSOC™ 4 firmware, see the PSOC™ 4 firmware considerations section. For information on developing the host driver, see the Host driver considerations section.

3.1 Sensing method

The CAPSENSE™ CSX method is generally considered the most appropriate choice for smart glass HMI designs. However, depending on the specific end application, it is possible to utilize either the CSD or CSX method. As per application requirements, if only single-touch position or one-finger gestures are required, then CAPSENSE™ CSD can be used, and if multi-touch with two-finger gestures are required, then CAPSENSE™ CSX can be used.

3.2 Low-power aspects

A CAPSENSE™ scan does not require the CPU to be in Active mode. If the firmware does not have any additional tasks other than waiting for the scan to finish, you can put the device to Sleep mode after initiating a scan, which will save power. When the CSD/CSX hardware completes the scan, it generates an interrupt to return the device to the Active mode. To achieve low power operation with a PSOC™ 4 CAPSENSE™ design, see the AN234231 – PSOC™ 4 CAPSENSE™ ultra-low-power capacitive sensing techniques.

For details on how to develop the firmware to achieve low power operation, see the PSOC™ 4: MSCLP low-power multitouch mutual-capacitance touchpad or PSOC™ 4: MSCLP multitouch mutual-capacitance touchpad tuning code example.

3.3 PSOC™ 4 firmware considerations

Smart glasses often rely on touch position and gesture detection as the primary means of user input. The user interacts with the device through a touchpad, which can have as few as two rows of sensors due to space constraints. To compensate for the limited sensor array, the device must accurately detect touch positions and respond accordingly.

The system relies on a PSOC™ 4 device that independently manages touch detection. The device continuously monitors the sensor array for touch inputs and calculates the corresponding position coordinates when a touch event is detected. Once the new touch position is determined, the device generates an interrupt signal via a GPIO, prompting the host to fetch the updated data over the I2C interface. This approach ensures that touch events are detected and communicated to the host quickly and efficiently.

To support narrow touchpads, utilize CAPSENSE™ middleware. See the instructions in the Smart glass CSX touchpad code example. Apart from the touch position and gestures, users can access many attributes, such as firmware version, reset reason, interrupt status, sensor raw count, and so on. For more information on the firmware, see the PSOC™ 4: MSCLP low-power smart glass CSX touchpad code example.

3.4 Host driver considerations

Smart glass applications typically require a host for post-processing. The PSOC™ 4 device is used for touch position and gesture detection. A serial interface is required to transmit data from the PSOC™ 4 device to the host. This document uses an I2C interface, along with an interrupt from the PSOC™ 4 device to demonstrate communication with a Raspberry Pi 4 host.

The PSOC™ 4 touchpad device provides access to a wide range of attributes and data, including but not limited to touch position, gestures, firmware version, interrupt status, and other relevant information, and allows to initiate various tests and actions, such as capacitive testing and software resets, as described in the SYSFS Attributes and DebugFS Attributes sections of the Linux host driver code example, which provide a detailed description of the available attributes and their usage.

To utilize the driver, see the setup instructions provided in the documentation file located in the `Doc` directory of the Linux host driver code example, which provides a comprehensive guide for building and setting up the driver on a Linux system or Raspberry Pi.

Before rebooting, ensure that the PSOC™ 4 device is properly connected to the Raspberry Pi via the I2C lines, with the Raspberry Pi GPIO17 is connected to the PSOC™ 4 MCU GPIO P4.0, as this is crucial for guaranteeing proper functionality and preventing potential issues.

After rebooting, see the detailed guide provided in the `Doc` directory of the Linux host driver code example for information on retrieving touchpad data, utilizing supported Sysfs Attributes. Upon driver's successful integration with the Linux input subsystem, use the `sudo evtest` command to test the host driver with the PSOC™ 4 MCU narrow touchpad and observe touch positions, gestures, and all supported attributes.

For more details, see the Linux host driver code example.

4 External factors affecting the system

In any system, there are multiple external factors that affect system performance. These aspects are described in details in the "**Noise in CAPSENSE™ system**" section in the PSOC™ 4 and PSOC™ 6 CAPSENSE™ design guide. This section discusses a few additional considerations that need to be made for narrow touchpads in particular.

4.1 Noise effect

Wearable devices are usually very dense in design and have multiple metal or conductive objects that are in close proximity to the CAPSENSE™ sensor and the PSOC™ 4 device. As part of sensor and system design, it is recommended to keep metal objects and battery as far as possible from the CAPSENSE™ sensor, since these can act as high noise sources for CAPSENSE™ sensors. In some devices magnetic objects, such as speakers are present, which must be kept as far from the CAPSENSE™ sensors as possible.

CAPSENSE™ sensors are susceptible to noise because of their large sensor area or high sensitivity setting. High noise makes it difficult to achieve a good signal-to-noise ratio (SNR) (typically greater than 5:1), which is required for the touchpad sensing. Table 3 lists the common sources that contribute to noise and their respective recommended mitigation techniques.

Table 3 Sources of noise and mitigation techniques

Noise source	Examples	Recommended noise mitigation technique
PWM driven devices	LEDs, motors	• Design to keep these sources as far as possible from the sensor • Add a ground hatch between these devices and sensors
Switching power converter	AC-DC, DC-DC, DC-AC	
High-speed communication interfaces	USB, Ethernet	• Use shielded twisted pair wires • Design to keep these sources as far as possible from the sensor • Add a ground hatch between these devices and sensors
AC supply lines	Relays and switches	Use shielded twisted pair wires when power lines are passing near the sensors

4.2 Battery-powered applications

When configuring CAPSENSE™ sensors on wearable devices that run on battery power, consider a few following key points:

In battery-powered portable applications, device ground and earth ground are lightly coupled, so, capacitance between device ground and earth ground (C_{GE}) is small. The resulting equivalent circuit is shown in Figure 13. So, in this condition, you get a lower sensitivity; that means you will get a lower signal for a finger touch, which is due to a decrease in capacitance seen at the device.

4 External factors affecting the system

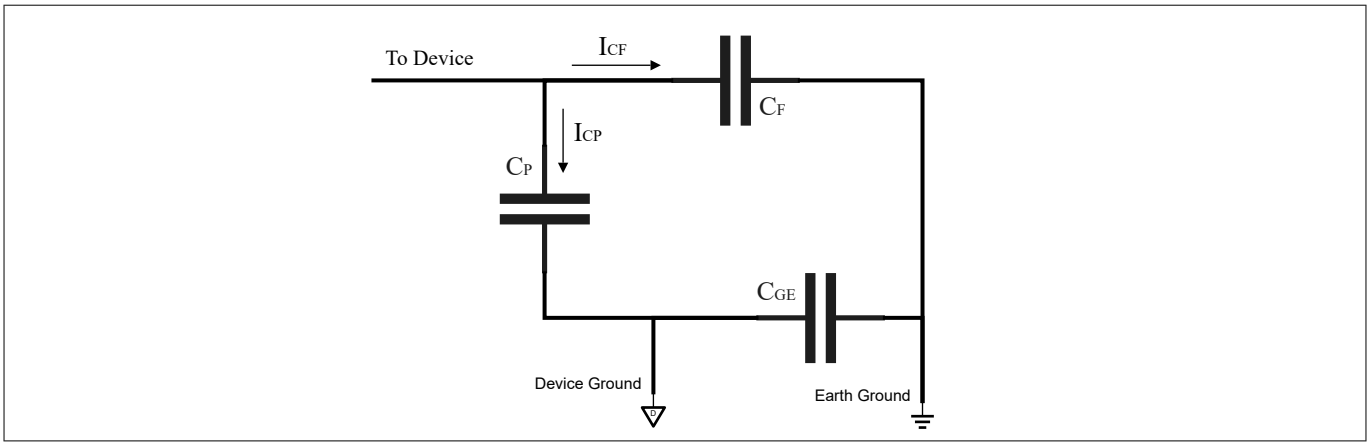


Figure 13 Equivalent circuit of the CSD for battery-powered application

Following are the recommendations for a CSD system design in a portable application powered by a battery:

- Add a large ground plane to the system. The ground plane should be away from the sensing element such that it does not increase the parasitic capacitance of the sensor. Follow the best practices for the PCB layout guidelines described in the "PCB layout guidelines" section in AN85951 – PSOC™ 4 and PSOC™ 6 CAPSENSE™ design guide
- Use a driven shield to improve the sensitivity of portable devices. For more information, see the "Layout guidelines for shield electrode" section in AN85951 – PSOC™ 4 and PSOC™ 6 CAPSENSE™ design guide
- Reduce the thickness of the overlay material or use an overlay with better dielectric value to improve sensitivity
- Place the system ground such that it can refer more closely with the user. This can also improve the ground reference

References

Webpage

[1] Infineon Technologies AG: *PSOC™ 4000T webpage*; Available online

Supported kits

[2] Infineon Technologies AG: *CY8CPROTO-040T: PSOC™ 4000T CAPSENSE™ Prototyping Kit*; Available online
[3] Infineon Technologies AG: *CY8CKIT-040T: PSOC™ 4000T CAPSENSE™ Evaluation Kit*; Available online
[4] Infineon Technologies AG: *CY8CPROTO-040T-MS: PSOC™ 4000T Multi-Sense Prototyping Kit*; Available online

Related resources

[5] Infineon Technologies AG: *CAPSENSE™ sensing controller webpage*; Available online
[6] Infineon Technologies AG: *ModusToolbox™ software*; Available online
[7] Infineon Technologies AG: *ModusToolbox™ software on GitHub*; Available online
[8] Infineon Technologies AG: *Eclipse IDE for ModusToolbox™ user guide*; Available online
[9] Infineon Technologies AG: *ModusToolbox™ CAPSENSE™ Configurator user guide*; Available online
[10] Infineon Technologies AG: *ModusToolbox™ CAPSENSE™ Tuner user guide*; Available online
[11] Infineon Technologies AG: *CAPSENSE™ Middleware Library*; Available online
[12] Infineon Technologies AG: *AN64846 – Getting started with CAPSENSE™*; Available online
[13] Infineon Technologies AG: *AN79953 – Getting started with PSOC™ 4 MCU*; Available online
[14] Infineon Technologies AG: *AN85951 – PSOC™ 4 and PSOC™ 6 CAPSENSE™ design guide*; Available online
[15] Infineon Technologies AG: *AN234185 – PSOC™ CAPSENSE™ touchpad design guide*; Available online
[16] Infineon Technologies AG: *AN234231 – PSOC™ 4 CAPSENSE™ ultra-low-power capacitive sensing techniques*; Available online

Revision history

Document revision	Date	Description of changes
**	2025-12-11	Initial release

Trademarks

PSOC™, formerly known as PSoC™, is a trademark of Infineon Technologies. Any references to PSoC™ in this document or others shall be deemed to refer to PSOC™.

Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2025-12-11

Published by

Infineon Technologies AG

81726 Munich, Germany

© 2025 Infineon Technologies AG

All Rights Reserved.

Do you have a question about any aspect of this document?

Email: erratum@infineon.com

Document reference
IFX-ezh1751543210718

Important notice

The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

Warnings

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.