

STM32 – Fieldscale SENSE™ [Use Case]

This document presents how an STM32 user can work with Fieldscale Sense to simulate the touch sensor performance on a system level. The user can first simulate the physical model of the sensor to extract its Equivalent Circuit in a netlist format and then use it on a system level analysis to simulate the Counts from the STM32 microcontroller just as they would be measured in the lab. Figure 1 shows the example board used and Figure 2 shows how the simulation results compare to the Counts measured.

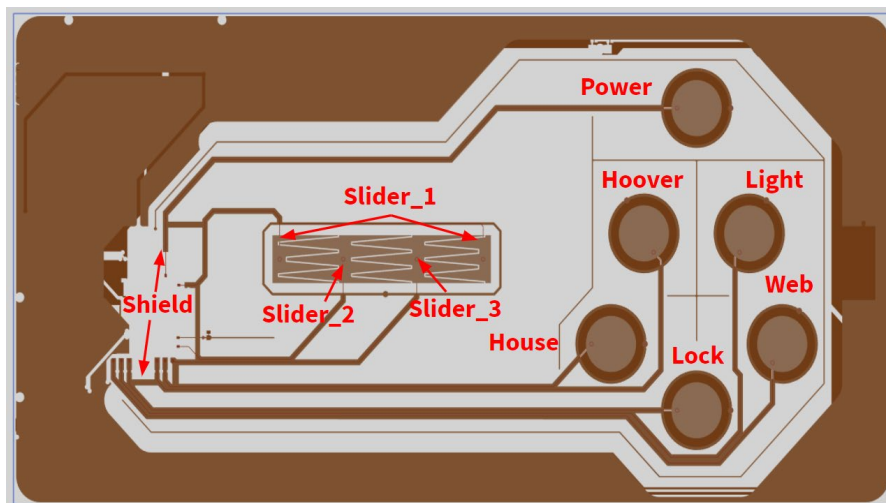


Figure 1: Example board layout and sensor name conventions

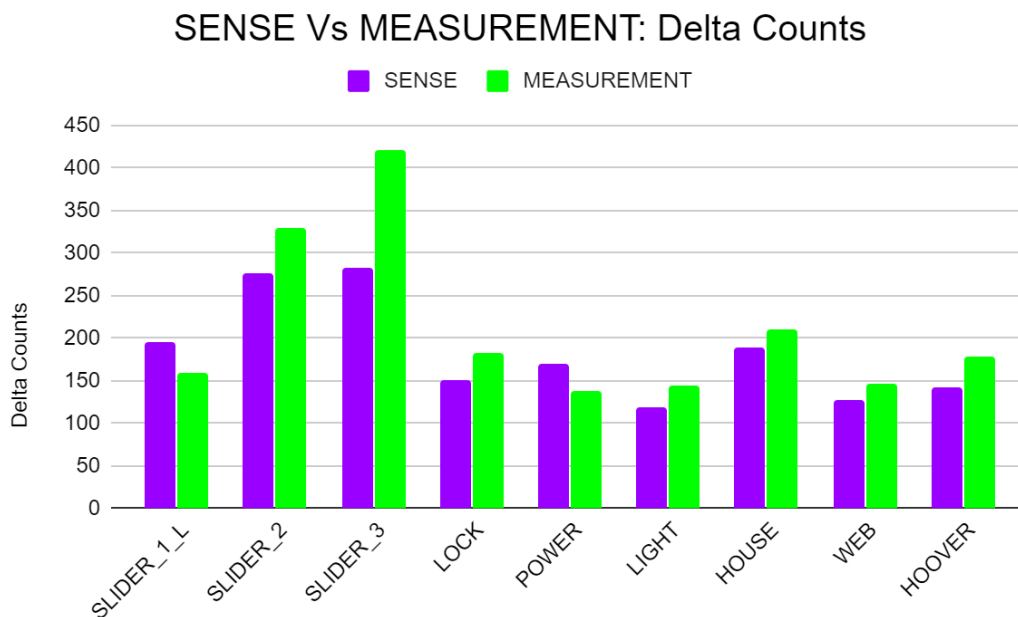


Figure 2. Delta Counts between no-touch and touch; simulated versus measured values.

A cylindrical pointer represents the human finger touch and multiple cases are investigated, one for each sensing element. The cylinder's diameter is 10 mm and its height is 20 mm. The Human Body Model is used as a connection between the finger and the "earth" which comprises a 1.5 kΩ resistor in series with a 100-pF capacitor. In each simulation a pointer is placed above one sensor and we extract the counts and delta counts for all sensors.

Finally, the user selects the computer power that will be used for the simulation, i.e. the number of CPU cores and RAM used, and click "Run" even from their tablet.

There are only a few steps needed in order to counter the problem of evaluating a touch sensor without going through the costly and time-consuming procedure of manufacturing and testing prototypes. With Sense, the user can effortlessly create the virtual representation of their sensor, virtually configure the STM32 MCU, and simply click "Run".

Sensor Virtual Model

The first step towards creating the virtual representation of the sensor is to define the shape of the sensor. The user can use elements from the software library, e.g. circular buttons, or import their own design. Sense supports the import of DXF and Gerber files without limitations on the number of conductive layers of the sensor either it is PCB-based or not. Figure 3 below shows a part of the process of importing gerber files describing an example board; traces, shields, sensing electrodes, everything is available and defined here.

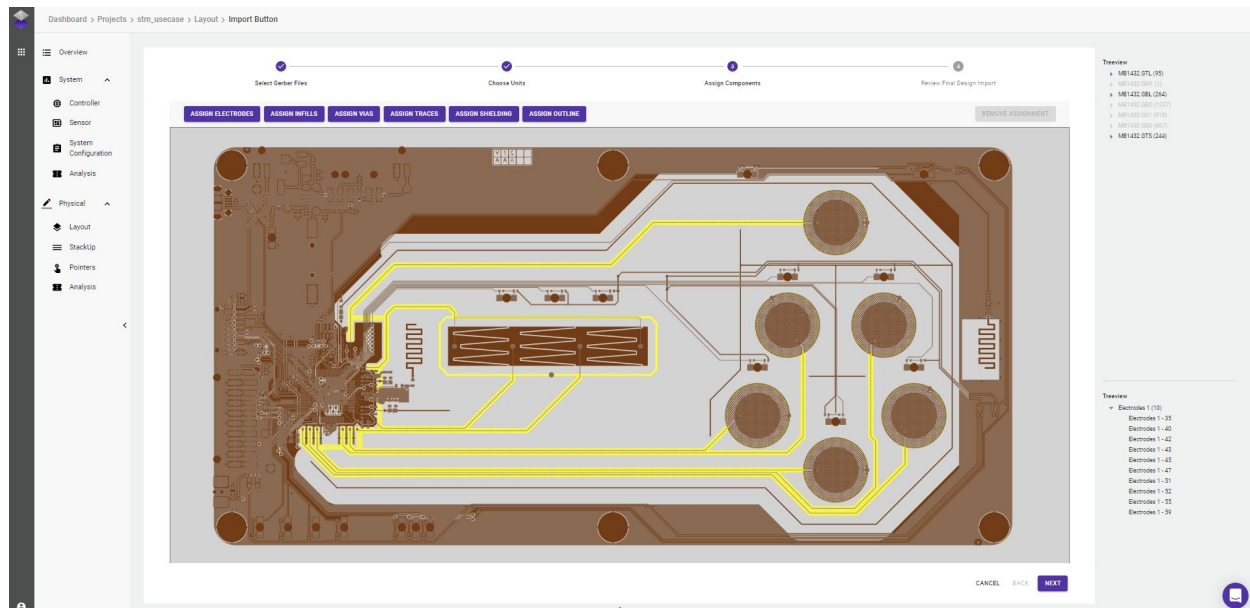


Figure 3. Importing gerber files in Fieldscale Sense

After completing the import of the sensor layout, the user needs to create the whole 3D model of the sensor by simply clicking your way through material selection. In the StackUp tab the user defines the thickness and relative permittivity of each layer in their sensor, designate where the

imported entities (electrodes, traces, shields, etc.) are placed and take care of all the via connections. In StackUp the user can explore the influence of an air-gap or a thick cover glass in the sensor performance.

The touch that can be detected by the MCU with the Counts difference is simulated by placing a Pointer in the model; this Pointer represents a human finger. The pointer that will be used for the analysis is created with a few clicks and is automatically included in the simulation models. In the Pointer tab, you can define a stylus, a test-probe, or a finger and you can also select where the touch should occur. How sensitive is button A when a finger is nearby in button B and what happens when a user touches the sensor on an abstract point? You can also analyze the performance of the sensor when gloves are being used.

Using the unique Equivalent Circuit analysis, you extract the circuit representation of your touch sensor in a netlist file. This netlist also describes the inputs and outputs of your sensor that are going to be connected to the controller.

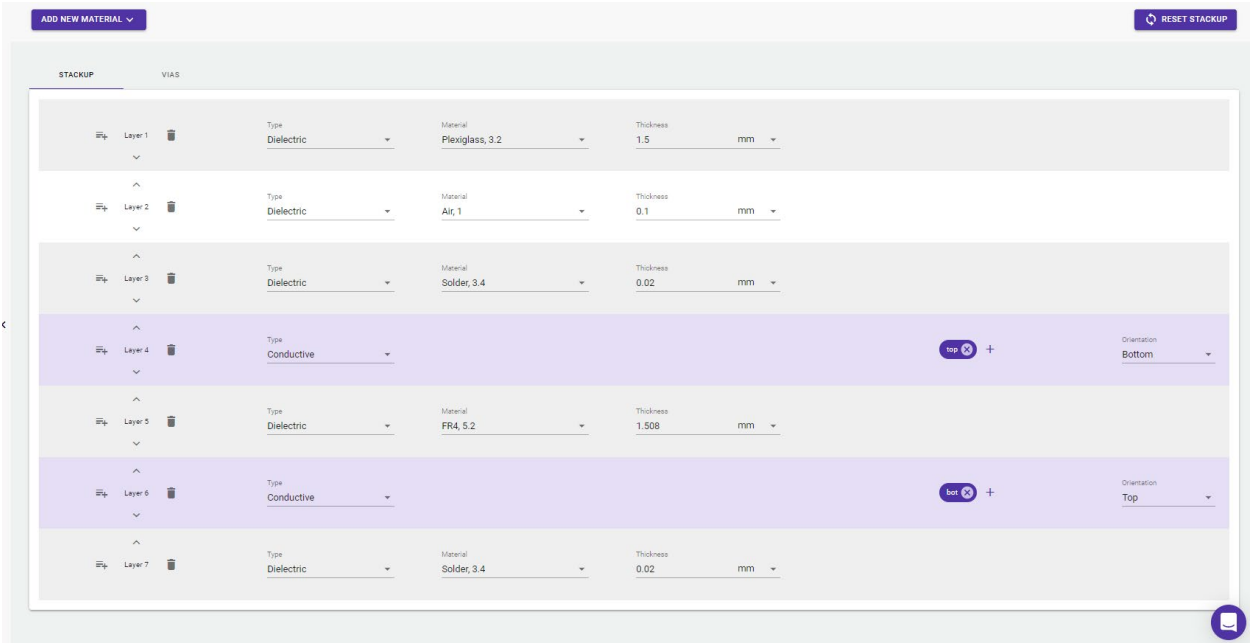


Figure 4. Defining the materials in the stack-up of the sensor.

Controller Virtual Model

Controller Circuit Name
STM32_config1

NETLIST PARAMETERS PINS

Hardware Clock
48000000 Hz

div
4 ▾

cth
6 ▾

ctl
8 ▾

max_counts
10000

delay
0.000002 sec

The STM32 microcontroller can be found in SENSE library and is used automatically for the simulation. All the user has to do is to define its parameters as shown in Figure 5.

Hardware Clock is the frequency of the clock used on the charging pulse applied to the sensing electrodes; the frequency of the applied pulses results from dividing the clock frequency by the number in Div. parameter. The parameters cth and ctl represent the number of periods that will be used for the charging (cth) and transferring (ctl) phases of the sensing.

The final step is to virtually connect the touch sensor circuit to the STM32. Here you can define the complete connectivity interface, including the operation voltage and lumped elements.

Figure 5. STM32 configuration parameters.

System Configuration Name
controller1_sensor1_config3

CONTROLLER AND SENSOR CONNECTIVITY INTERFACE POINTER

Controller Pins	Connection	Connection Options
VDD	Voltage ▾	Voltage 3.45 Threshold 1.725 V
G1_I01	Capacitor ▾	Capacitance 47 nF
G1_I02	Touch Sensor ▾	Sensor Pin <input checked="" type="checkbox"/> Resistance 10 kΩ
G1_I03	Floating ▾	Sensor Pin
G1_I04	Touch Sensor ▾	Sensor Pin

Figure 6. Connectivity interface between STM32 and touch sensor.