UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

SOLAS OLED LTD., *Plaintiff*, vs. SAMSUNG ELECTRONICS CO., LTD., SAMSUNG ELECTRONICS AMERICA, INC., *Defendants*.	CASE NO. 2:21-cv-00105 **Complaint for Patent Infringement** **JURY DEMANDED**

Complaint for Patent Infringement

Plaintiff Solas OLED Ltd. ("Solas") files this complaint against Samsung Electronics Co., Ltd. ("SEC") and Samsung Electronics America, Inc. ("SEA") (each a "Defendant" and, collectively, "Defendants"), alleging infringement of U.S. Patent Nos. 9,292,144 and 8,526,767 ("Patents-in-Suit"). The Accused Products are the OLED panel displays made, used, offered for sale, sold, imported by Defendants in the United States and supplied by Defendants to customers and integrated into electronic devices sold in the United States.

Plaintiff Solas OLED and the Patents-in-Suit.

1. Plaintiff Solas is a technology licensing company organized under the laws of Ireland, with its headquarters at The Hyde Building, Suite 23, The Park, Carrickmines, Dublin 18, Ireland.

2. Solas is the owner of U.S. Patent No. 9,292,144, entitled "Touch-Sensor-Controller Sensor Hub," which issued March 22, 2016 (the "'144 patent"). A copy of the '144 patent is attached to this complaint as Exhibit 1.

3. Solas is the owner of U.S. Patent No. 8,526,767, entitled "Gesture Recognition," which issued September 3, 2013 (the "'767 patent"). A copy of the '767 patent is attached to this complaint as Exhibit 2.

Defendants and the Accused Products.

4. On information and belief, Defendant Samsung Electronics Co., Ltd. ("SEC") is a corporation organized under the laws of South Korea, with its principal place of business at 129, Samsung-Ro, YeongTong-Gu, Suwon-Si, Gyonggi-Do, 443-742, South Korea.

5. On information and belief, Defendant Samsung Electronics America, Inc. ("SEA") is a United States corporation organized under the laws of the State of New York, with its principal place of business at 85 Challenger Road, Ridgefield Park, New Jersey 07660. SEA is a wholly-owned subsidiary of SEC. SEA distributes certain Samsung consumer electronics products, including the Accused Products, in the United States.

6. The Accused Products are laptop computers, mobile phones and tablets made, used, offered for sale, sold, imported by Defendants in the United States, including without limitation the Samsung laptop computers and Galaxy mobile phones and tablet devices.

Jurisdiction and Venue.

7. This action arises under the patent laws of the United States, Title 35 of the United States Code. This Court has original subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a).

8. This Court has personal jurisdiction over Defendants in this action because Defendants have established minimum contacts with the United States as a whole such that the exercise of jurisdiction would not offend traditional notions of fair play and substantial justice. Defendants have purposefully directed activities at the United States, in particular, directing

Accused Products for sale to customers and distributors within the United States (including within this District) and engaging in sales and marketing efforts to generate and support such sales. Defendants have committed acts of infringement of Solas's patents giving rise to this action, such as by supplying to distributors and consumer device retailers the Accused Products in this District, including without limitation the Samsung ATIV and Galaxy laptop computers, tablets and phones accused of infringement in this case. Defendants, directly and through subsidiaries, intermediaries, and third parties, have committed and continues to commit acts of infringement in this District by, among other things, making, using, offering to sell, selling, and importing products that infringe the Asserted Patents.

9. Venue is proper in this District under 28 U.S.C. §§ 1391 and 1400(b). Defendant SEC is a foreign corporation. Venue is proper as to a foreign defendant in any district. 28 U.S.C. § 1391(c)(3). Defendant SEA has committed acts of infringement in this District and has regular and established places of business in this District.

Count 1 – Claim for infringement of the '144 patent.

10. Solas incorporates by reference each of the allegations in paragraphs 1–9 above and further alleges as follows:

11. On March 22, 2016, the United States Patent and Trademark Office issued U.S. Patent No. 9,292,144, entitled "Touch-Sensor-Controller Sensor Hub." Ex. 1.

12. Solas is the owner of the '144 patent with full rights to pursue recovery of royalties for damages for infringement, including full rights to recover past and future damages.

13. Each claim of the '144 patent is valid, enforceable, and patent-eligible.

14. Solas and its predecessors in interest have satisfied the requirements of 35 U.S.C. § 287(a) with respect to the '144 patent, and Solas is entitled to damages for Defendants' past infringement.

15. Defendants have directly infringed (literally and equivalently) and induced others to infringe the '144 patent by making, using, selling, offering for sale, or importing products that infringe the claims of the '144 patent and by inducing others to infringe the claims of the '144 patent without a license or permission from Solas, such as for example instructing users of the Accused Products to perform the patented methods of the '144 patent.

16. On information and belief, Defendants make, import, offer for sale, and sell certain infringing products in the United States. The Accused Products are, for example, consumer electronic devices manufactured by SEC and imported, sold, and offered for sale in the United States by SEA, including for example Samsung Galaxy mobile phones. The Accused Products all have touch controller chips for controlling one or more sensors in the Accused Products.

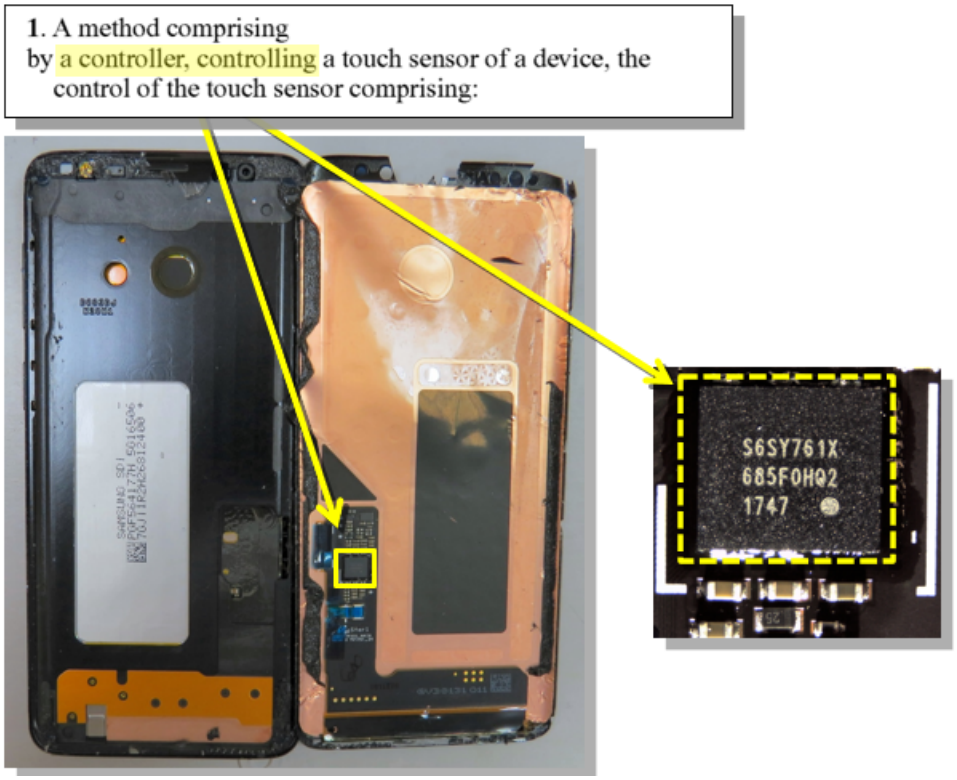
Samsung Galaxy S9

Samsung Galaxy S20

17. For example, claim 1 of the ’144 patent claim “a method” as follows:

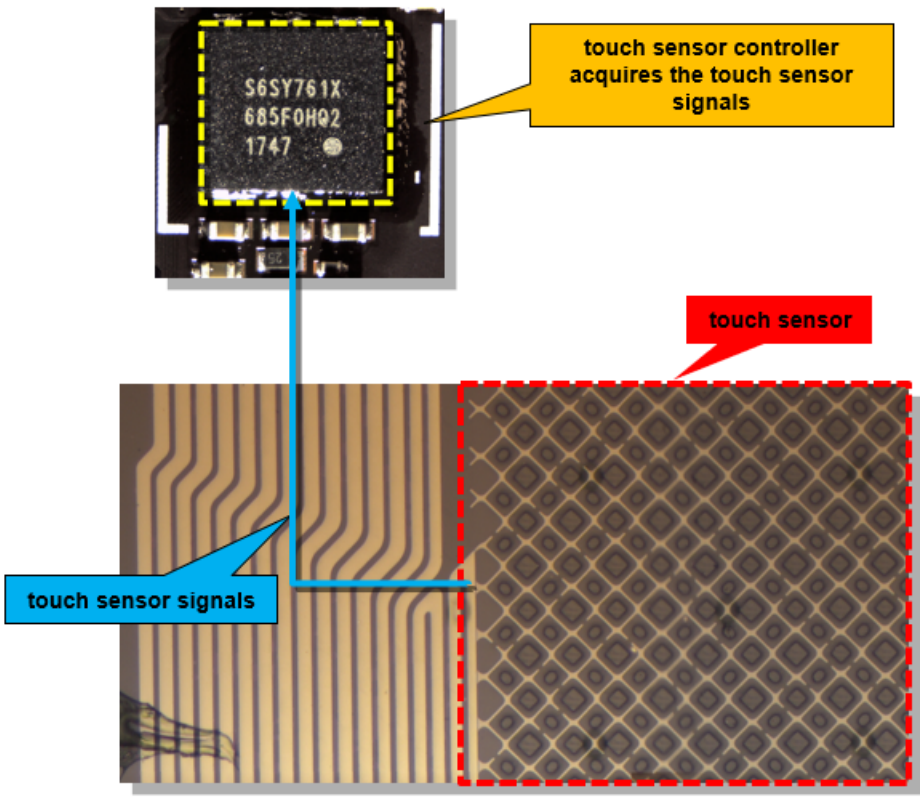
[1a] “by a controller, controlling a touch sensor of a device, the control of the touch sensor comprising;”

18. The Accused Products (such as the Galaxy S9, pictured below) comprise a touch controller which controls a touch sensor of a device.



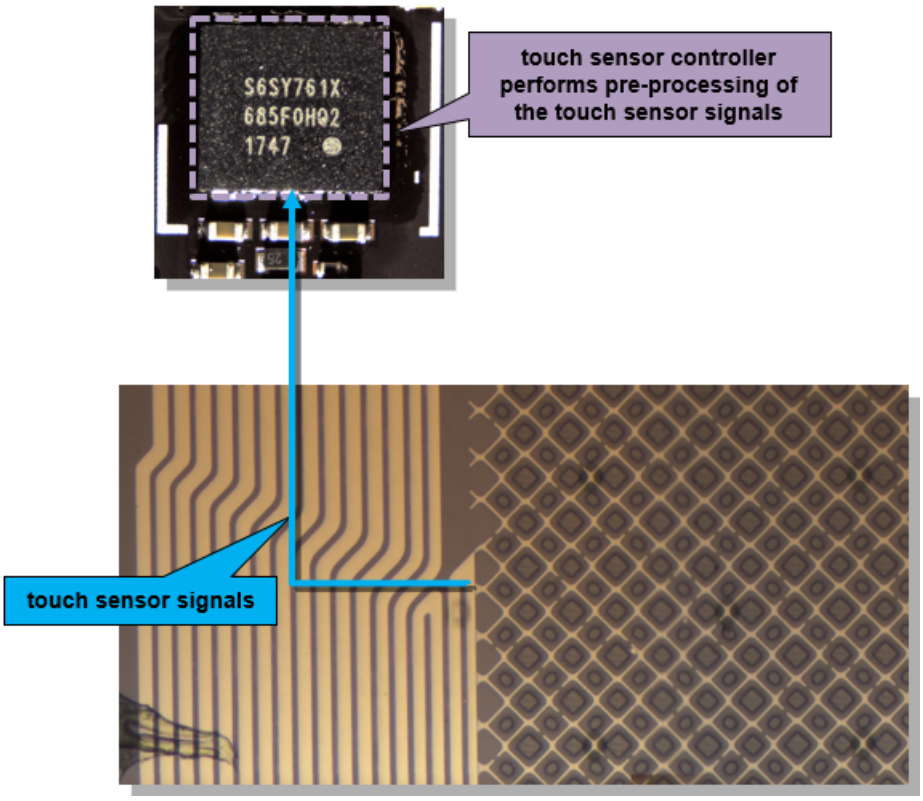
[1b] "acquisition of touch-sensor signals from the touch sensor;"

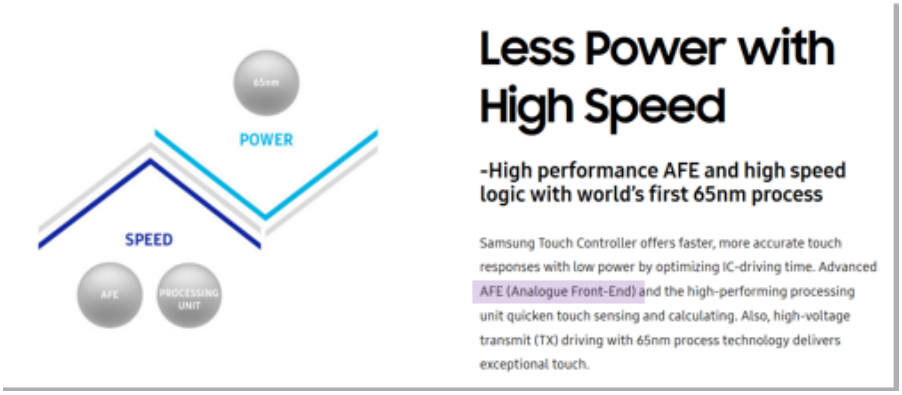
19. The Accused Products comprise a touch controller which controls a touch sensor by "acquisition of touch-sensor signals from the touch sensor":



[1c] "pre-processing of the touch-sensor signals;"

20. The Accused Products pre-process touch-sensor signals:

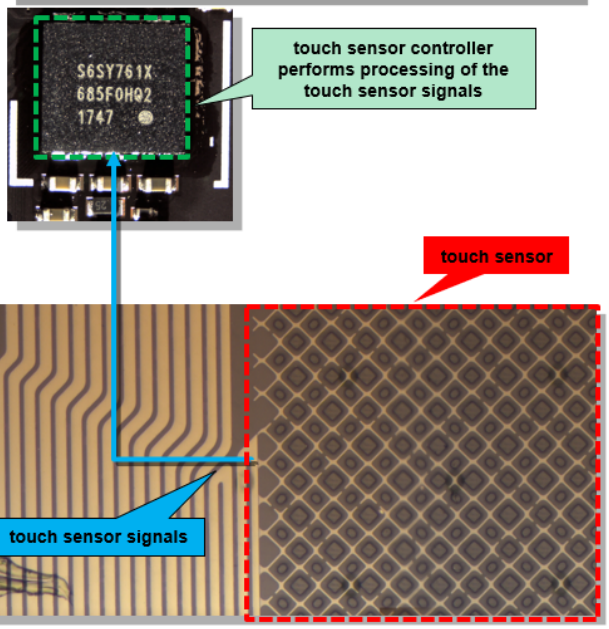


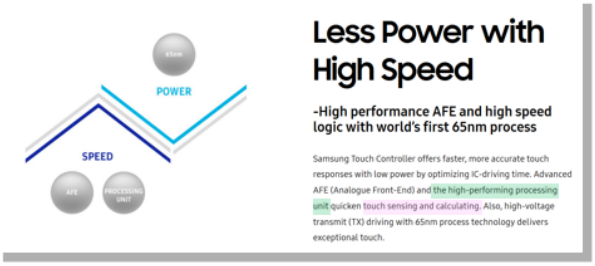


[1d] "and processing of the touch-sensor signals to determine: whether a touch or proximity input has occurred with respect to the touch sensor;"

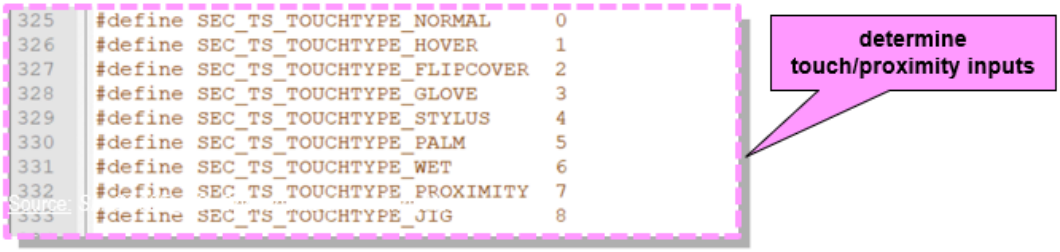
21. The Accused Products process the touch-sensor signals to determine: whether a touch or proximity input has occurred with respect to the touch sensor:

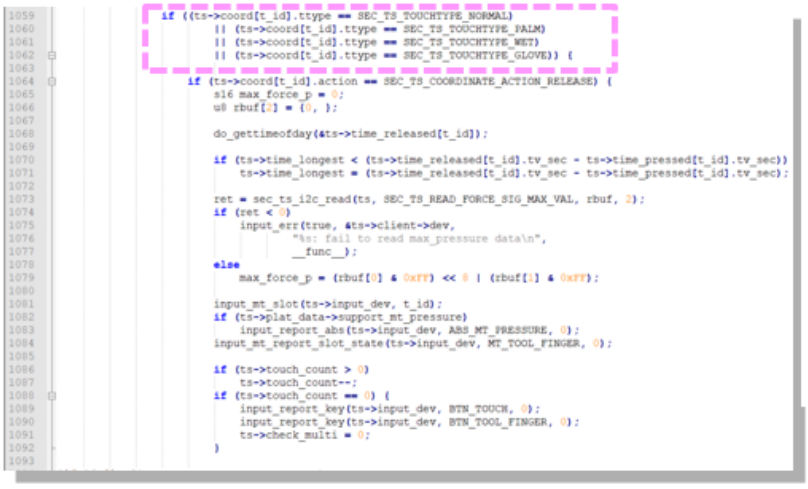
and
processing of the touch-sensor signals to determine:
whether a touch or proximity input has occurred with
respect to the touch sensor;





The touch controller performs processing of the touch-sensors signals to determine if a touch has occurred on the touch sensor.



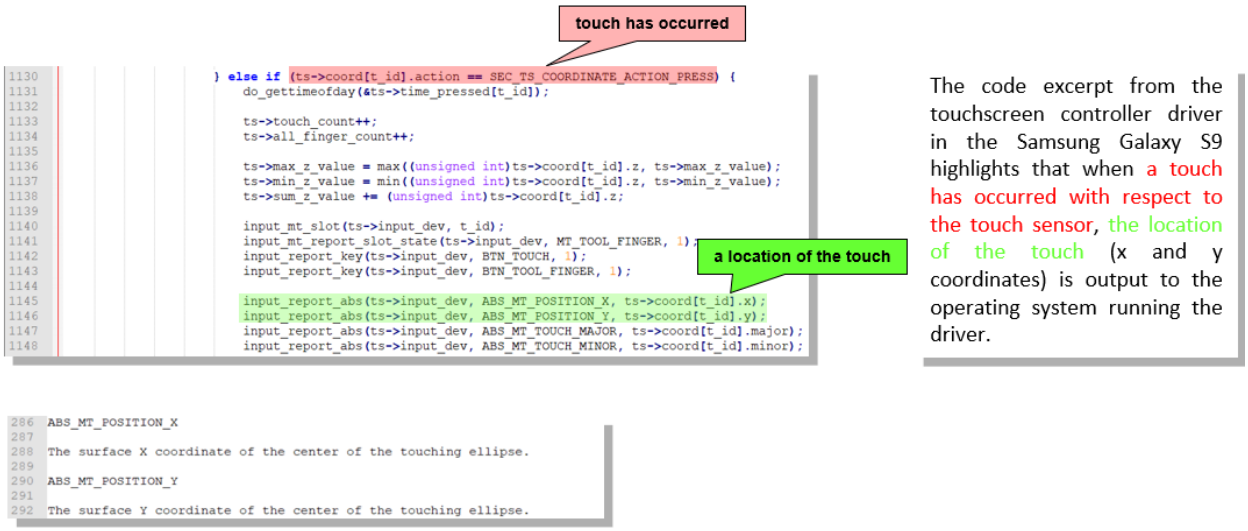


```
if ((ts->coord[t_id].ttype == SEC_TS_TOUCHTYPE_NORMAL)
        || (ts->coord[t_id].ttype == SEC_TS_TOUCHTYPE_PALM)
        || (ts->coord[t_id].ttype == SEC_TS_TOUCHTYPE_WET)
        || (ts->coord[t_id].ttype == SEC_TS_TOUCHTYPE_GLOVE)) {
    if (ts->coord[t_id].action == SEC_TS_COORDINATE_ACTION_RELEASE) {
        s16 max_force_p = 0;
        u8 rbuf[2] = {0, };

        do_gettimeofday(&ts->time_released[t_id]);

        if (ts->time_longest < (ts->time_released[t_id].tv_sec - ts->time_pressed[t_id].tv_sec))
            ts->time_longest = (ts->time_released[t_id].tv_sec - ts->time_pressed[t_id].tv_sec);

        ret = sec_ts_i2c_read(ts, SEC_TS_READ_FORCE_SIG_MAX_VAL, rbuf, 2);
        if (ret < 0)
            input_err(true, &ts->client->dev,
                      "%s: fail to read max_pressure data\n",
                      __func__);
        else
            max_force_p = (rbuf[0] & 0xFF) << 8 | (rbuf[1] & 0xFF);

        input_mt_slot(ts->input_dev, t_id);
        if (ts->plat_data->support_mt_pressure)
            input_report_abs(ts->input_dev, ABS_MT_PRESSURE, 0);
        input_mt_report_slot_state(ts->input_dev, MT_TOOL_FINGER, 0);

        if (ts->touch_count > 0)
            ts->touch_count--;
        if (ts->touch_count == 0) {
            input_report_key(ts->input_dev, BTN_TOUCH, 0);
            input_report_key(ts->input_dev, BTN_TOOL_FINGER, 0);
            ts->check_multi = 0;
        }
```

The code excerpt from the touchscreen controller driver in the Samsung Galaxy S9 highlights the different modes of touch which the touchscreen controller can detect. This is achieved through processing of the touch-sensor signals from the touch sensor.

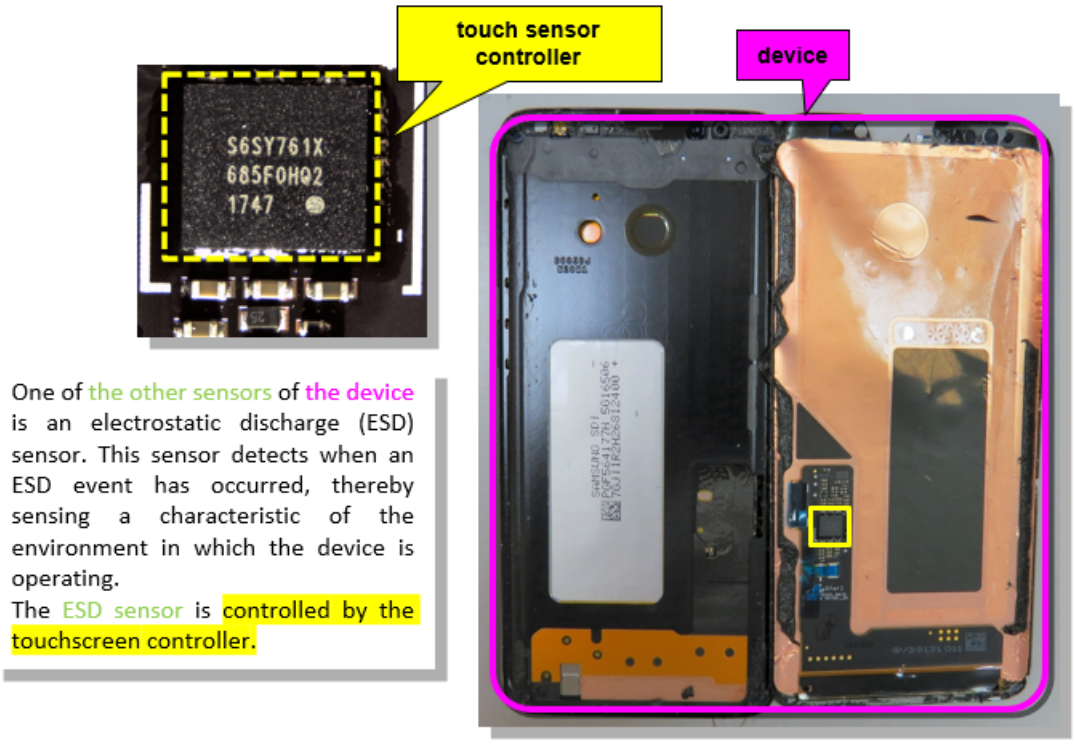
[1e] "if the touch or proximity input has occurred with respect to the touch sensor, a location of the touch or proximity input; and"

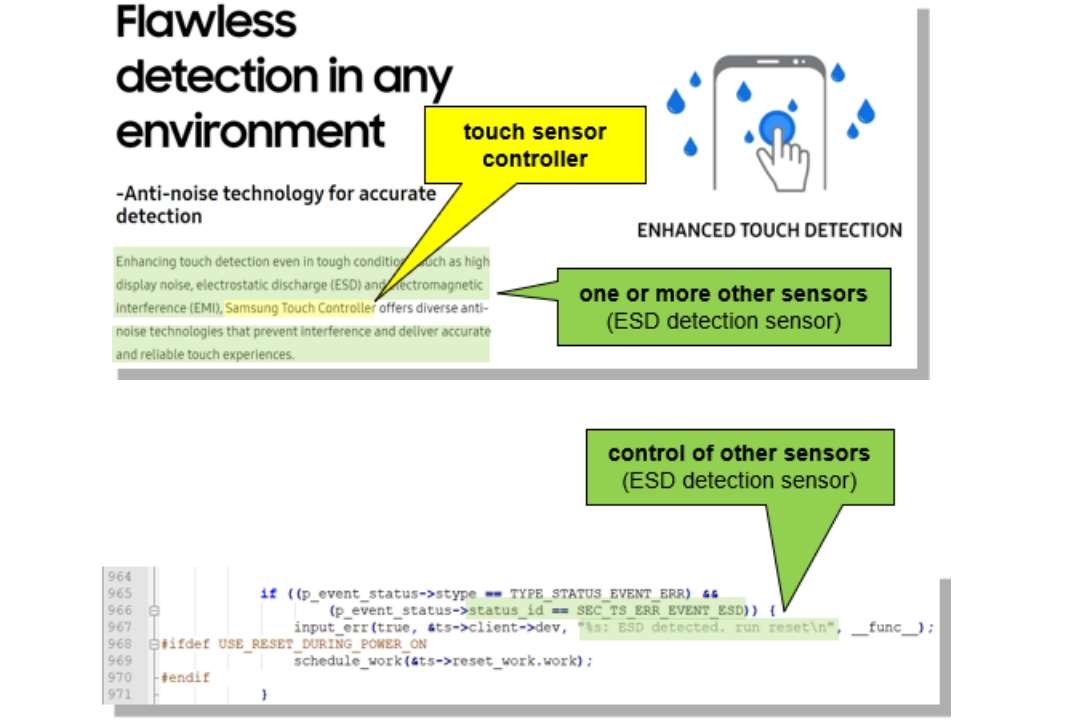
22. The Accused Products determine, if the touch or proximity input has occurred with respect to the touch sensor, a location of the touch or proximity input:



[1f] "by the controller, controlling one or more other sensors of the device, the control of the other sensors occurring at least in part concurrently with the acquisition of the touch-sensor signals from the touch sensor or the pre-processing of the touch-sensor signals."

23. The Accused Products' touch controllers control one or more other sensors of the device, the control of the other sensors occurring at least in part concurrently with the acquisition of the touch-sensor signals from the touch sensor or the pre-processing of the touch-sensor signals:





24. Defendants also knowingly and intentionally induce and contribute to infringement of the '144 patent in violation of 35 U.S.C. §§ 271(b) and 271(c). Through the

filing and service of this Complaint, Defendants have had knowledge of the '144 patent and the infringing nature of the Accused Products. Despite this knowledge of the '144 patent, Defendants continue to actively encourage and instruct its customers to use and integrate the accused products in ways that directly infringe the '144 patent. Defendants do so knowing and intending that their customers will commit these infringing acts. Defendants also continue to make, use, offer for sale, sell, and/or import the Accused Products, despite their knowledge of the '144 patent, thereby specifically intending for and inducing its customers to infringe the '144 patent through the customers' normal and customary use of the Accused Products.

25. Defendants have infringed multiple claims of the '144 patent, including independent claim 1. By way of example only, the normal and customary use of the accused Samsung Galaxy S9 phone infringes an exemplary claim of the '144 patent, as in the description set forth above, which Solas provides without the benefit of information about the Accused Products obtained through discovery.

26. Defendants have known how the Accused Products are made and have known, or have been willfully blind to the fact, that making, using, offering to sell, and selling the Accused Products to their customers, would constitute willful infringement of the '144 patent. Those products imported into and sold within the United States include, without limitation, Samsung laptop computers, Galaxy tablets and phones.

27. Defendants have induced, and continue to induce, infringement of the '144 patent by actively encouraging others (including its customers) to use, offer to sell, sell, and import the Accused Products. On information and belief, these acts include providing information and instructions on the use of the Accused Products; providing information, education and

instructions to its customers; providing the Accused Products to customers; and indemnifying patent infringement within the United States.

28. Solas has been damaged by Defendant's infringement of the '144 patent and is entitled to damages as provided for in 35 U.S.C. § 284, including reasonable royalty damages.

Count 2 – Claim for infringement of the '767 patent.

29. Solas incorporates by reference each of the allegations in paragraphs 1–28 above and further alleges as follows:

30. On September 3, 2013, the United States Patent and Trademark Office issued U.S. Patent No. 8,526,767, entitled "Gesture Recognition." Ex. 2.

31. Solas is the owner of the '767 patent with full rights to pursue recovery of royalties for damages for infringement, including full rights to recover past and future damages.

32. Each claim of the '767 patent is valid, enforceable, and patent-eligible.

33. Solas and its predecessors in interest have satisfied the requirements of 35 U.S.C. § 287(a) with respect to the '767 patent, and Solas is entitled to damages for Defendants' past infringement.

34. Defendants have directly infringed (literally and equivalently) and induced others to infringe the '767 patent by making, using, selling, offering for sale, or importing products that infringe the claims of the '767 patent and by inducing others to infringe the claims of the '767 patent without a license or permission from Solas, such as for example instructing users of the Accused Products to perform the patented methods of the '767 patent.

35. On information and belief, Defendants make, import, offer for sale, and sell certain infringing products in the United States. The Accused Products are, for example, consumer electronic devices manufactured by SEC and imported, sold, and offered for sale in the

United States by SEA, including for example Samsung Galaxy mobile phones. The Accused Products all have touch controller chips for controlling one or more sensors in the Accused Products.

Samsung Galaxy S9

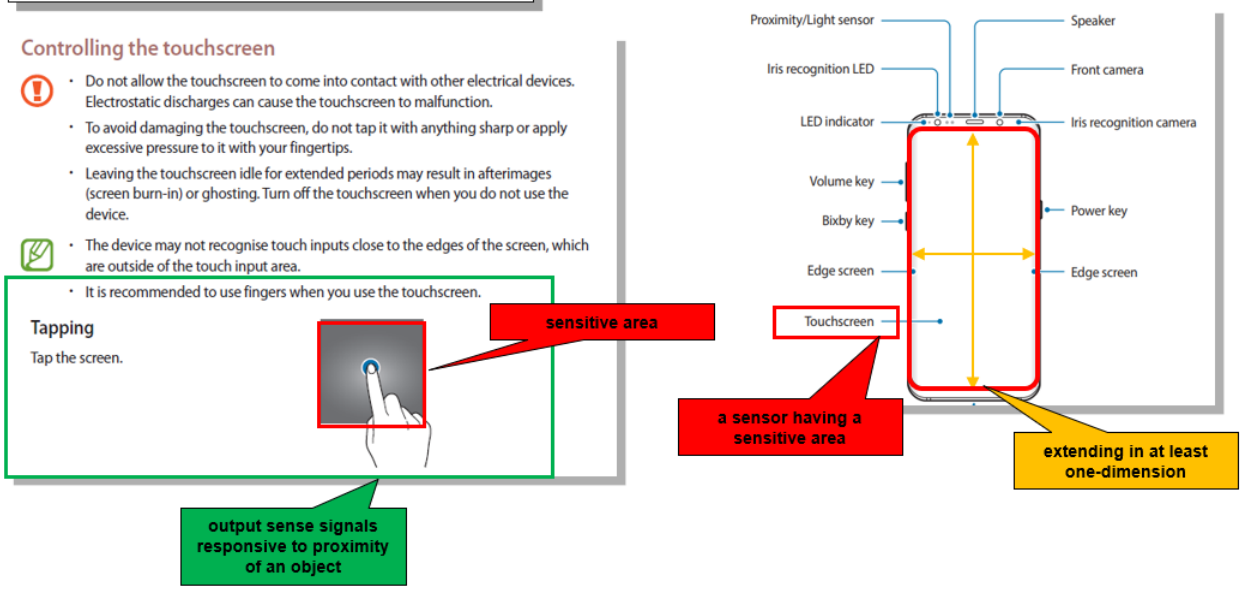
Samsung Galaxy S20

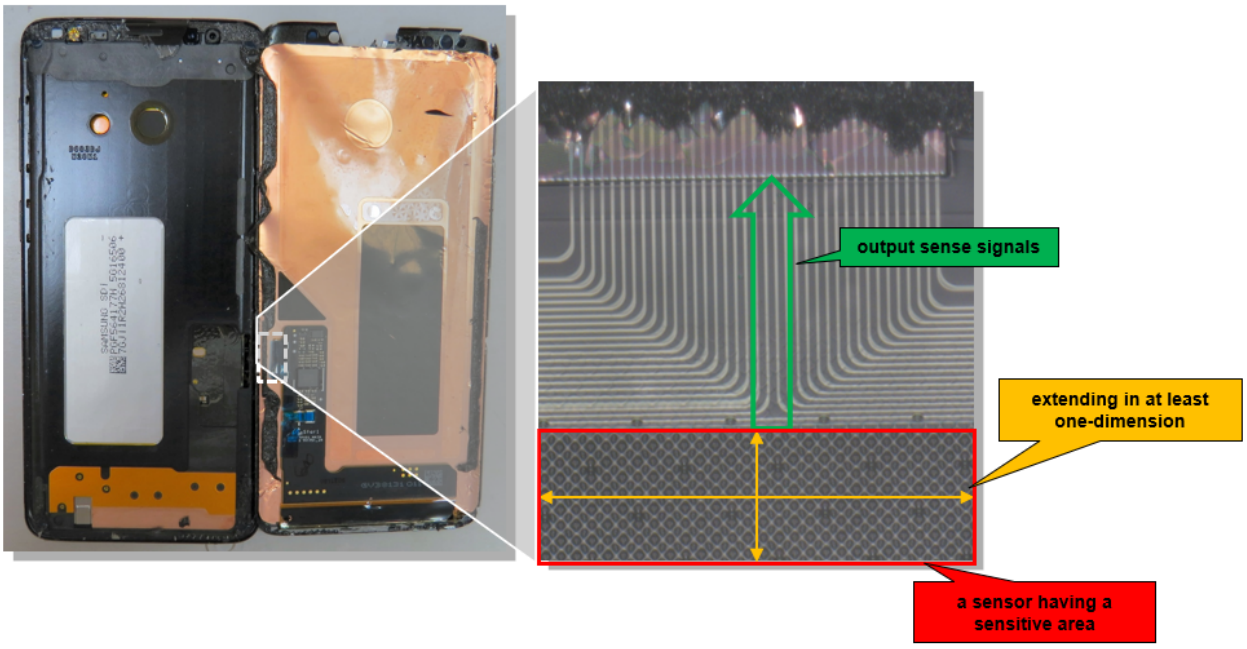
36. For example, claim 1 of the '767 patent claim a "touch sensor device" as follows:

[1a] "a sensor having a sensitive area extending in at least one-dimension and arranged to output sense signals responsive to proximity of an object to the sensitive area;"

37. The Accused Products (such as the Galaxy S9, pictured below) comprise a sensor having a sensitive area extending in at least one-dimension and arranged to output sense signals responsive to proximity of an object to the sensitive area:

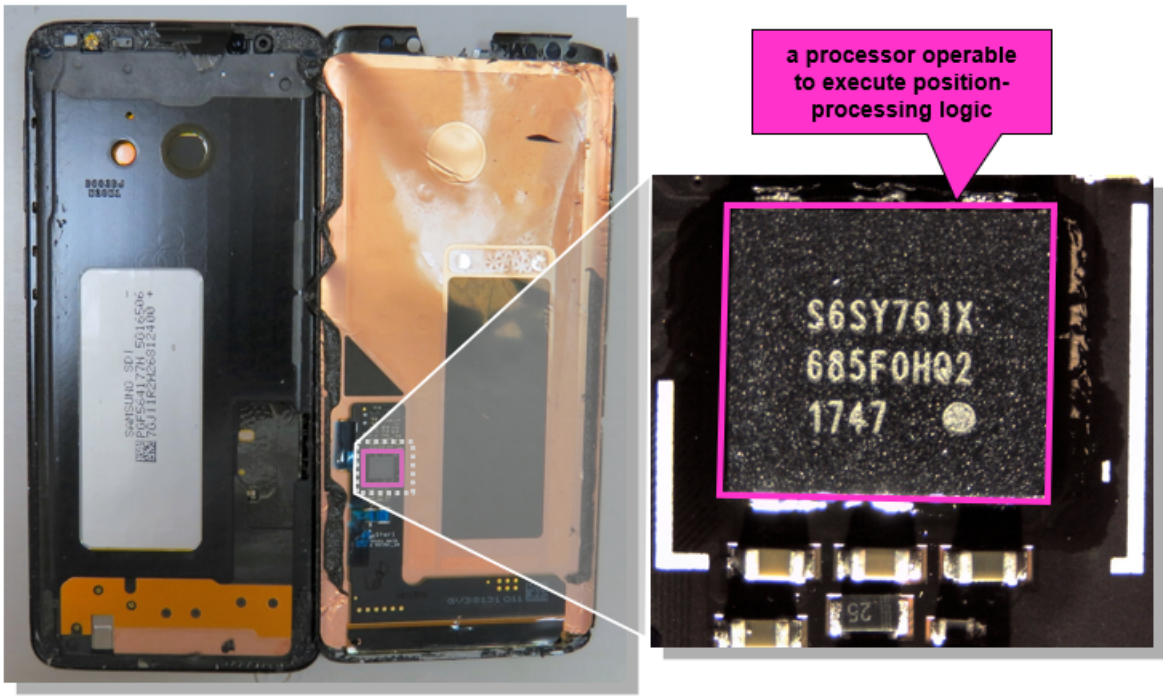
a sensor having a sensitive area extending in at least one-dimension and arranged to output sense signals responsive to proximity of an object to the sensitive area;

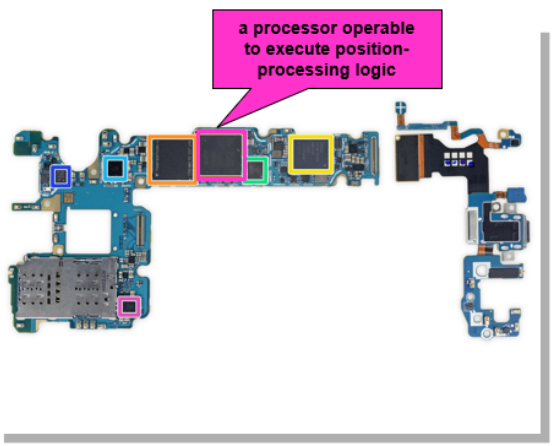


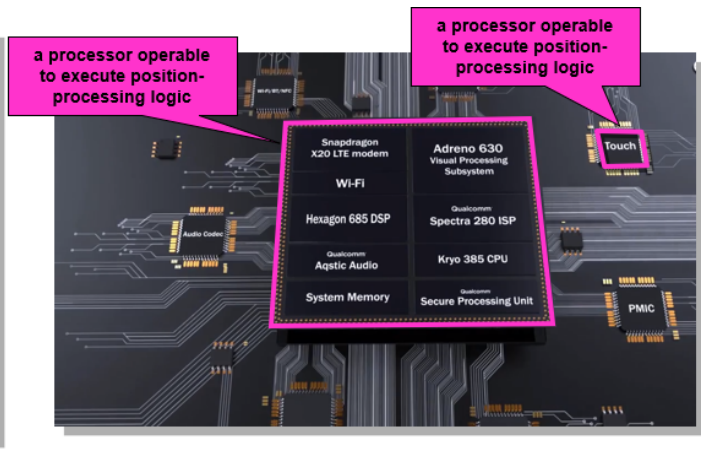


[1b] "a processor operable to execute position-processing logic stored in one or more tangible media, the position-processing logic, when executed by the processor, configured to:"

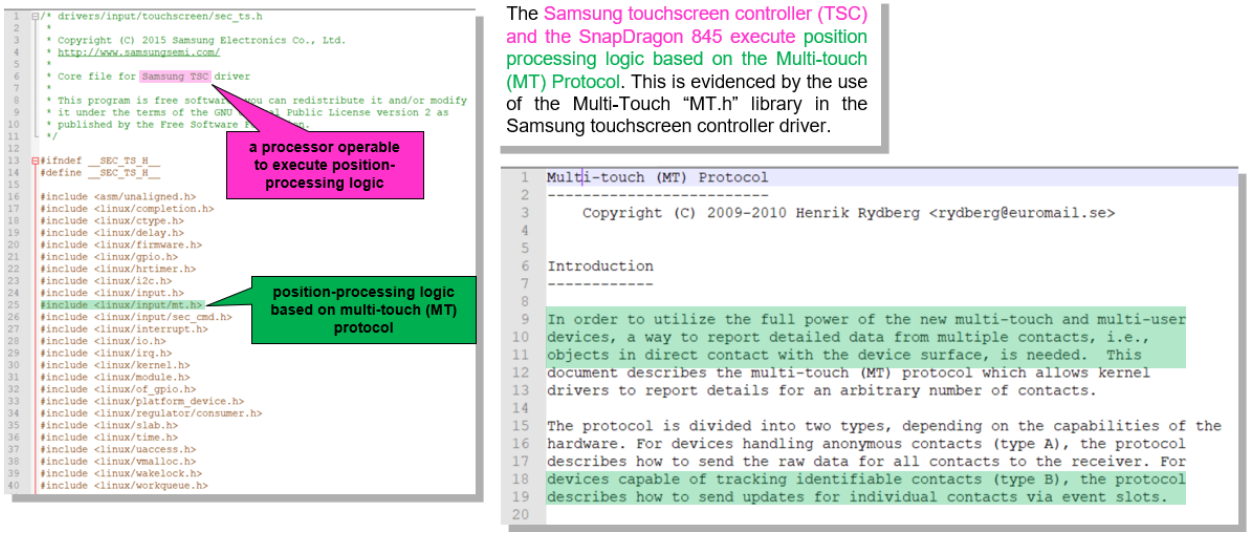
38. The Accused Products comprise a processor operable to execute position-processing logic stored in one or more tangible media, the position-processing logic:







- Samsung K3UH5H5-OMMAGCJ 32 Gb (4 GB) LPDDR4X DRAM, layered over a Qualcomm Snapdragon 845

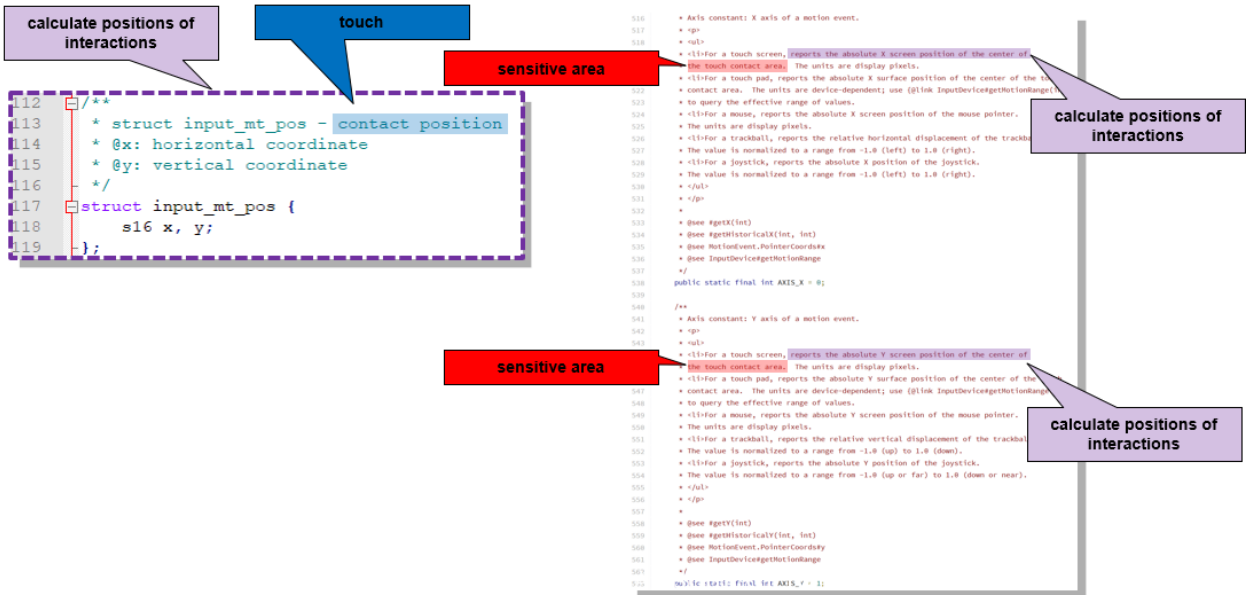


[1c] "calculate positions of interactions with the sensitive area from an analysis of the sense signals; and output a times series of data indicative of the interaction positions on the sensor, the interactive positions corresponding to the touches; and"

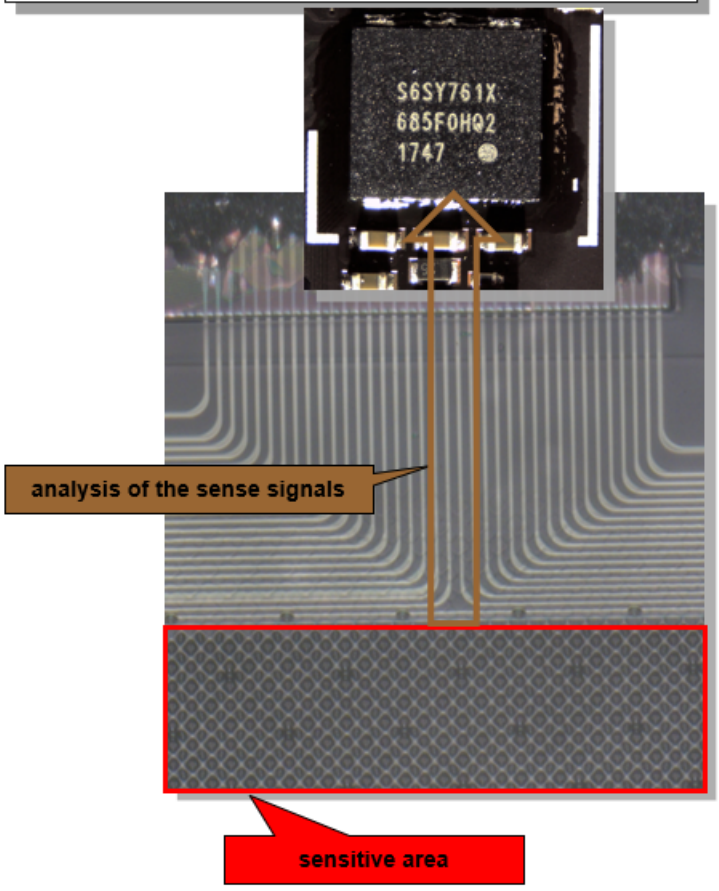
39. The Accused Products calculate positions of interactions with the sensitive area from an analysis of the sense signals, and output a times series of data indicative of the interaction positions on the sensor, the interactive positions corresponding to the touches:

calculate positions of interactions with the sensitive area from an analysis of the sense signals; and

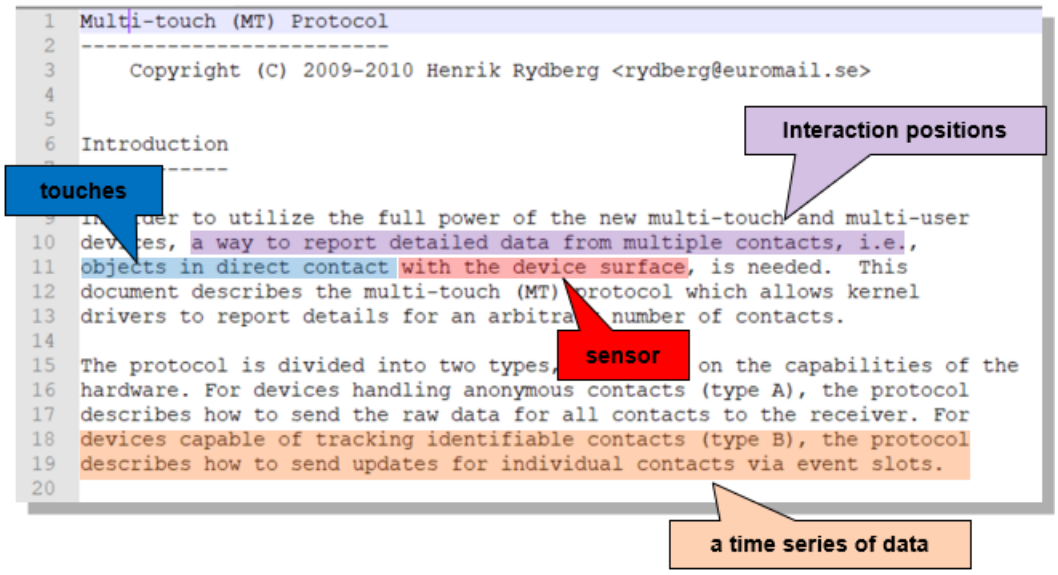
output a times series of data indicative of the interaction positions on the sensor, the interaction positions corresponding to touches; and



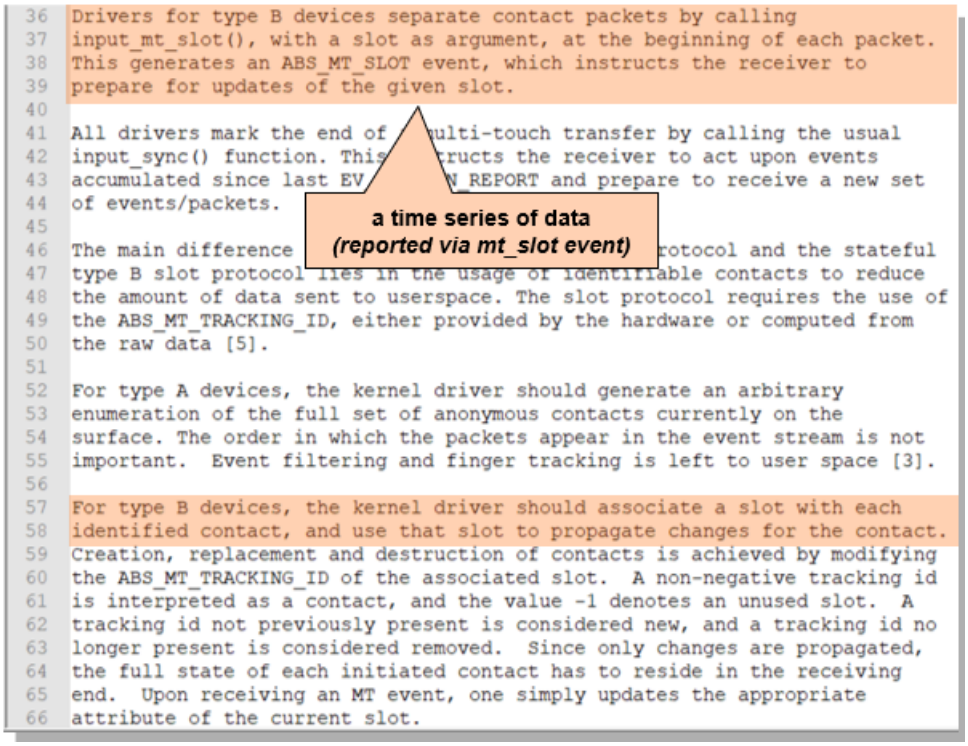
calculate positions of interactions with the sensitive area from an analysis of the sense signals; and
output a times series of data indicative of the interaction positions on the sensor, the interaction positions corresponding to touches; and



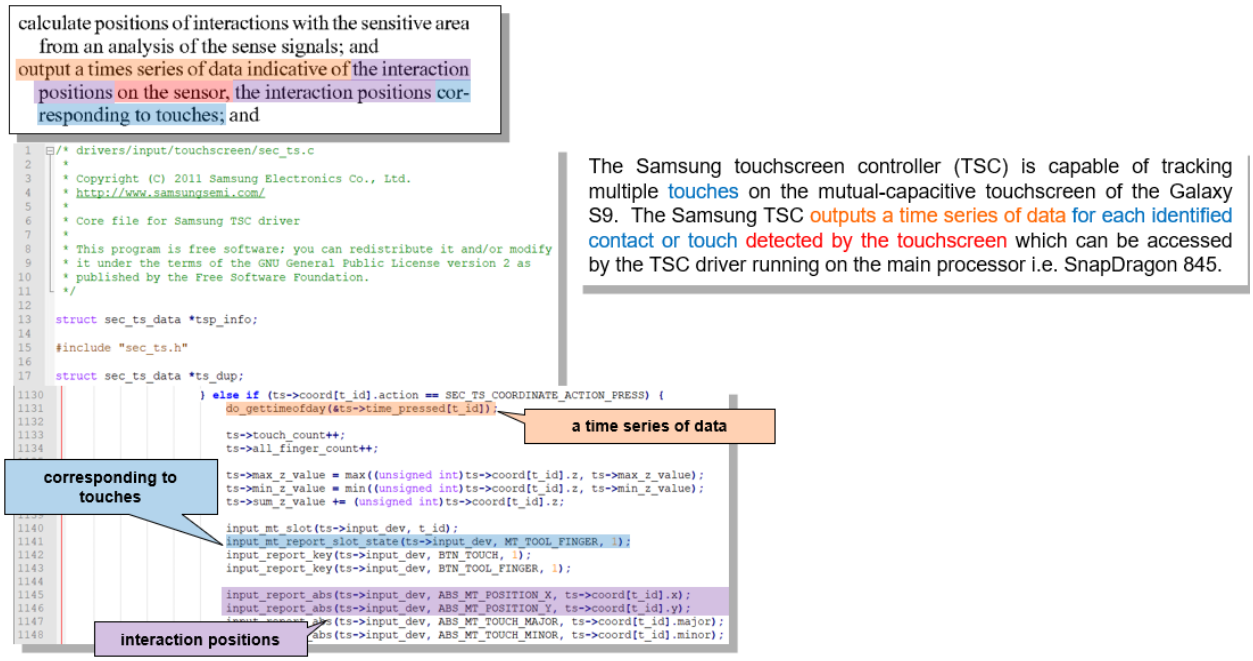
calculate positions of interactions with the sensitive area from an analysis of the sense signals; and
output a times series of data indicative of the interaction positions on the sensor, the interaction positions corresponding to touches; and




```
Multi-touch (MT) Protocol
-------------------------
    Copyright (C) 2009-2010 Henrik Rydberg <rydberg@euromail.se>


Introduction
------------

In order to utilize the full power of the new multi-touch and multi-user
devices, a way to report detailed data from multiple contacts, i.e.,
objects in direct contact with the device surface, is needed.  This
document describes the multi-touch (MT) protocol which allows kernel
drivers to report details for an arbitrary number of contacts.

The protocol is divided into two types, depending on the capabilities of the
hardware. For devices handling anonymous contacts (type A), the protocol
describes how to send the raw data for all contacts to the receiver. For
devices capable of tracking identifiable contacts (type B), the protocol
describes how to send updates for individual contacts via event slots.
```

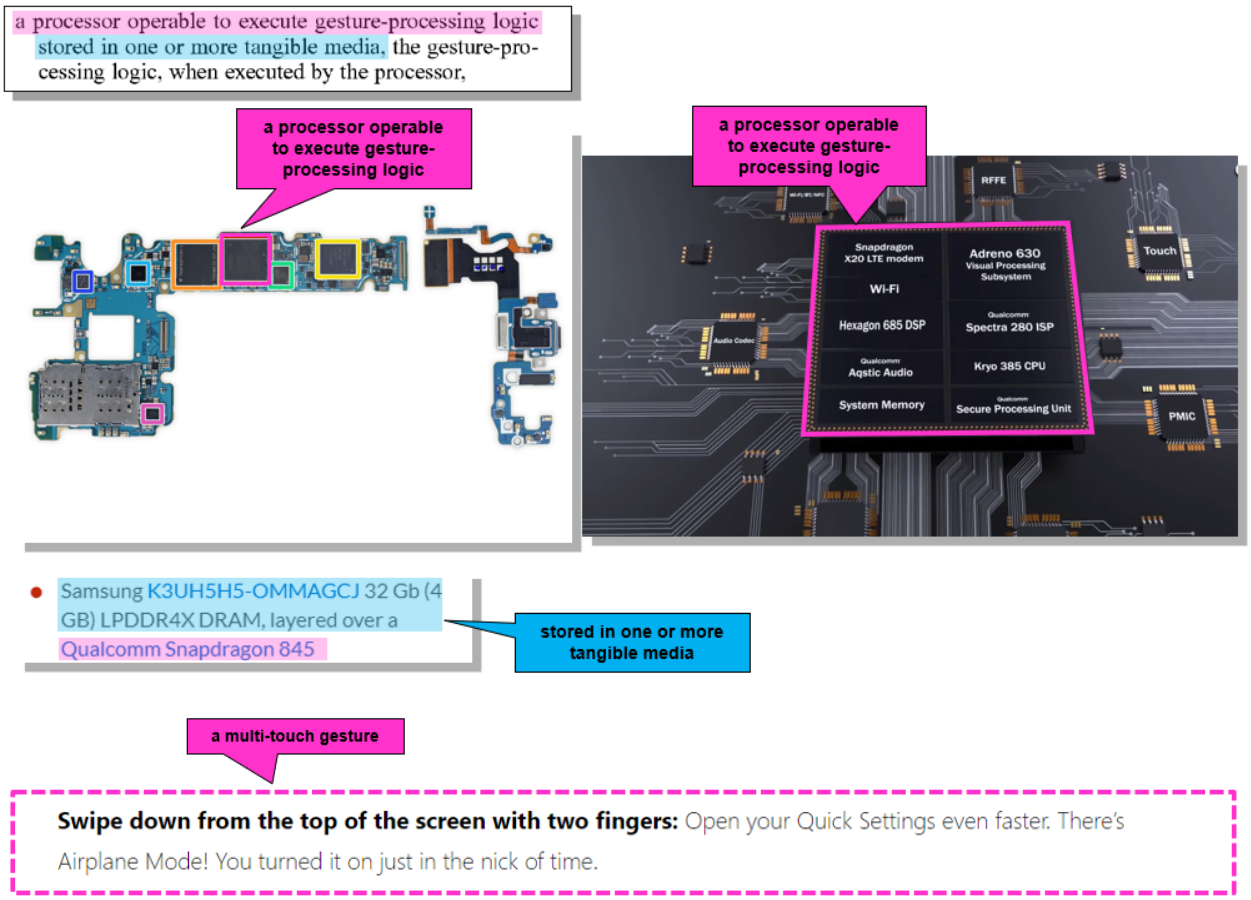



```
Drivers for type B devices separate contact packets by calling
input_mt_slot(), with a slot as argument, at the beginning of each packet.
This generates an ABS_MT_SLOT event, which instructs the receiver to
prepare for updates of the given slot.

All drivers mark the end of a multi-touch transfer by calling the usual
input_sync() function. This instructs the receiver to act upon events
accumulated since last EV_SYN/SYN_REPORT and prepare to receive a new set
of events/packets.

The main difference between the stateless type A protocol and the stateful
type B slot protocol lies in the usage of identifiable contacts to reduce
the amount of data sent to userspace. The slot protocol requires the use of
the ABS_MT_TRACKING_ID, either provided by the hardware or computed from
the raw data [5].

For type A devices, the kernel driver should generate an arbitrary
enumeration of the full set of anonymous contacts currently on the
surface. The order in which the packets appear in the event stream is not
important.  Event filtering and finger tracking is left to user space [3].

For type B devices, the kernel driver should associate a slot with each
identified contact, and use that slot to propagate changes for the contact.
Creation, replacement and destruction of contacts is achieved by modifying
the ABS_MT_TRACKING_ID of the associated slot.  A non-negative tracking id
is interpreted as a contact, and the value -1 denotes an unused slot.  A
tracking id not previously present is considered new, and a tracking id no
longer present is considered removed.  Since only changes are propagated,
the full state of each initiated contact has to reside in the receiving
end.  Upon receiving an MT event, one simply updates the appropriate
attribute of the current slot.
```



[1d] "a processor operable to execute gesture-processing logic stored in one or more tangible media, the gesture-processing logic"

40. The Accused Products have a processor operable to execute gesture-processing logic stored in one or more tangible media, the gesture-processing logic, when executed by the processor, configured to analyze the time series of data to distinguish one or more gesture inputs from the time series of data, the gesture-processing logic being coded with gesture-recognition code comprising a plurality of state-machine modules:



https://r2.community.samsung.com/t5/Tech-Talk/Guide-to-Touchscreen-Gestures/td-p/4095444

[1e] "when executed by the processor, configured to analyze the time series of data to distinguish one or more gesture inputs from the time series of data, the gesture-processing logic being coded with gesture-recognition code comprising a plurality of state-machine modules, the plurality of state-machine modules comprising:"

41. The Accused Products have a logic "when executed by the processor, configured to analyze the time series of data to distinguish one or more gesture inputs from the time series of data, the gesture-processing logic being coded with gesture-recognition code comprising a plurality of state-machine modules:

config-
ured to analyze the time series of data to distinguish one or more gesture inputs from the time series of data, the gesture-processing logic being coded with gesture-recognition code comprising a plurality of state-machine modules, the plurality of state-machine modules comprising:

The Samsung touchscreen controller (TSC) driver runs on the SnapDragon845 processor which is the receiver of the time series of data for each individual touch detected by the TSC.

a time series of data *(reported via mt_slot event)*

analyze the time series of data (mt_slot_events provided)

```
Drivers for type B devices separate contact packets by calling
input_mt_slot(), with a slot as argument, at the beginning of each packet.
This generates an ABS_MT_SLOT event, which instructs the receiver to
prepare for updates of the given slot.

All drivers mark the end of a multi-touch transfer by calling the usual
input_sync() function. This instructs the receiver to act upon events
accumulated since last EV_SYN/SYN_REPORT and prepare to receive a new set
of events/packets.

The main difference between the stateless type A protocol and the stateful
type B slot protocol lies in the usage of identifiable contacts to reduce
the amount of data sent to userspace. The slot protocol requires the use of
the ABS_MT_TRACKING_ID, either provided by the hardware or computed from
the raw data [5].

For type A devices, the kernel driver should generate an arbitrary
enumeration of the full set of anonymous contacts currently on the
surface. The order in which the packets appear in the event stream is not
important.  Event filtering and finger tracking is left to user space [3].

For type B devices, the kernel driver should associate a slot with each
identified contact, and use that slot to propagate changes for the contact.
Creation, replacement and destruction of contacts is achieved by modifying
the ABS_MT_TRACKING_ID of the associated slot.  A non-negative tracking id
is interpreted as a contact, and the value -1 denotes an unused slot.  A
tracking id not previously present is considered new, and a tracking id no
longer present is considered removed.  Since only changes are propagated,
the full state of each initiated contact has to reside in the receiving
end.  Upon receiving an MT event, one simply updates the appropriate
attribute of the current slot.
```

Detect common gestures

distinguish gesture inputs

A "touch gesture" occurs when a user places one or more fingers on the touch screen, and your application interprets that pattern of touches as a particular gesture. There are correspondingly two phases to gesture detection:

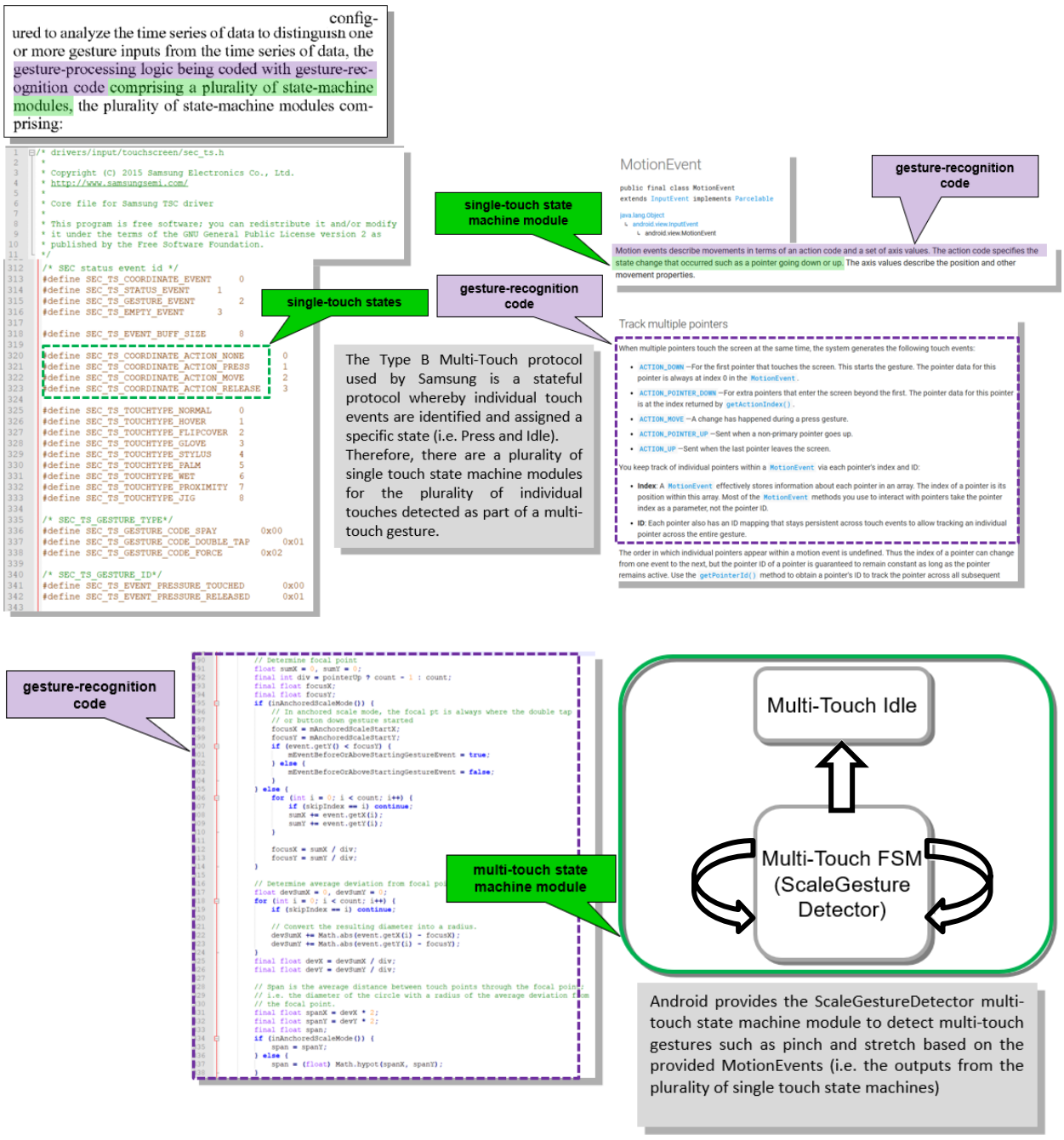
1. Gather data about touch events.

the time series of data

2. Interpret the data to see if it meets the criteria for any of the gestures your app supports.

analyze

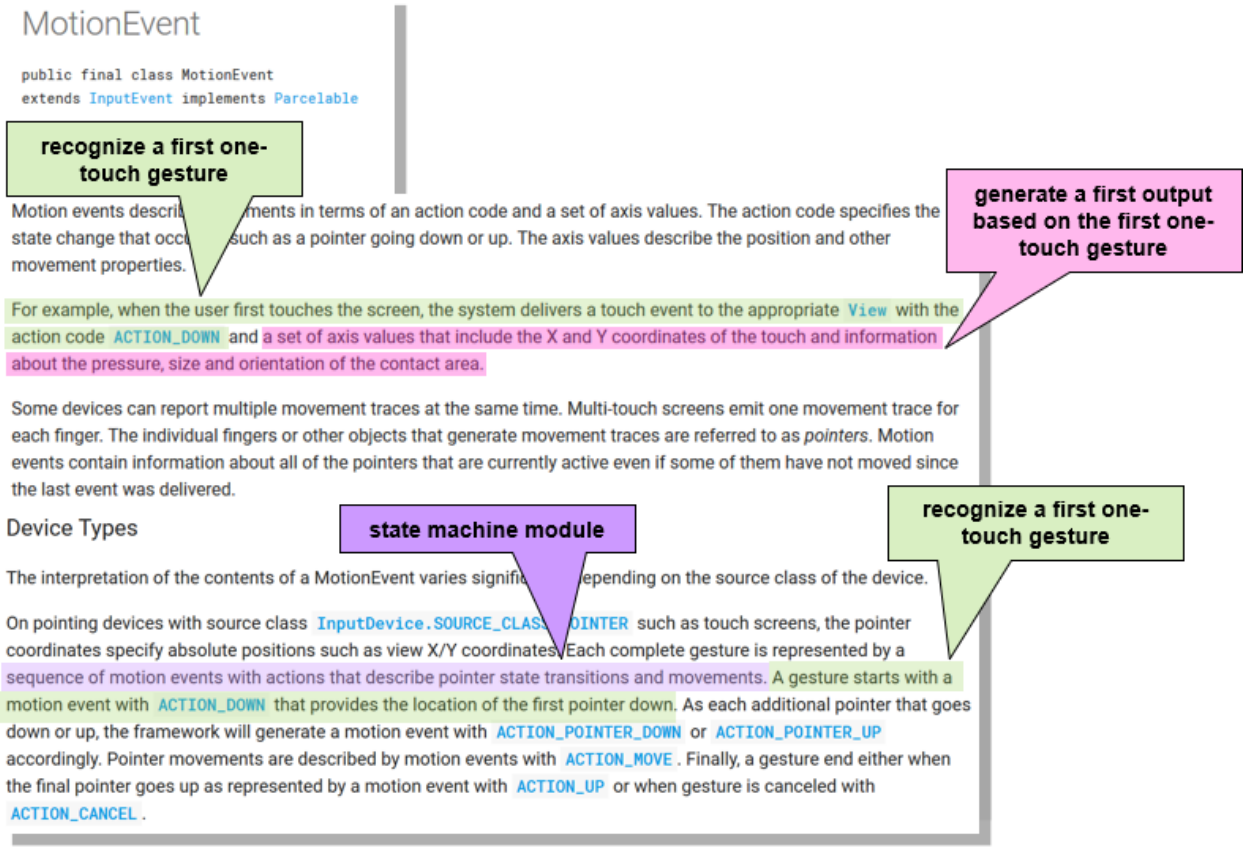
Android runs on the SnapDragon845 processor which provides a convention for distinguishing single and multi-touch gestures. The SnapDragon845 processor analyses the time series of data received for each individual touch detected to distinguish gestures (i.e. spreading and pinching) made on the touchscreen.

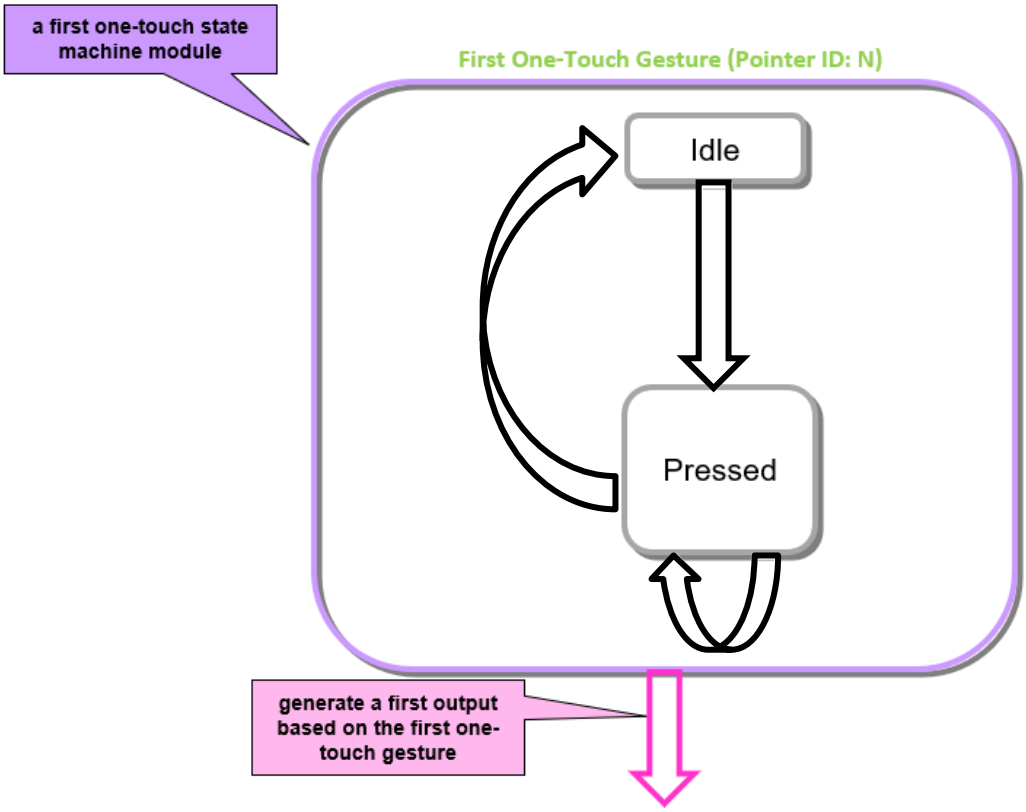


[1f] "a first one-touch state-ma chine module, the first one-touch state-machine module being operable to recognize at least a first one-touch gesture and generate a first output based on the first one-touch gesture;"

42. The Accused Products have a first one-touch state-machine module, the first one-touch state-machine module being operable to recognize at least a first one-touch gesture and generate a first output based on the first one-touch gesture:

a first one-touch state-machine module, the first one-touch state-machine module being operable to recognize at least a first one-touch gesture and generate a first output based on the first one-touch gesture;





[1g] "a second one-touch state-machine module, the second one-touch state-machine module being operable to recognize at least a second one-touch gesture and generate a second output based on the second one-touch gesture; and"

43. The Accused Products have a second one-touch state-machine module, the second one-touch state-machine module being operable to recognize at least a second one-touch gesture and generate a second output based on the second one-touch gesture:

a second one-touch state-machine module, the second one-touch state-machine module being operable to recognize at least a second one-touch gesture and generate a second output based on the second one-touch gesture; and

MotionEvent

```
public final class MotionEvent
extends InputEvent implements Parcelable
```

java.lang.Object
↳ android.view.InputEvent
↳ android.view.MotionEvent

Motion events describe movements in terms of an action code and a set of axis values. The action code specifies th state change that occurred such as a pointer going down or up. The axis values describe the position and other movement properties.

For example, when the user first touches the screen, the system delivers a touch event to the appropriate `View` with the action code `ACTION_DOWN` and a set of axis values that include the X and Y coordinates of the touch and information about the pressure, size and orientation of the contact area.

Some devices can report multiple movement traces at the same time. Multi-touch screens emit one movement trace for each finger. The individual fingers or other objects that generate movement traces are referred to as *pointers*. Motion events contain information about all of the pointers that are currently active even if some of them have not moved since the last event was delivered.

Device Types

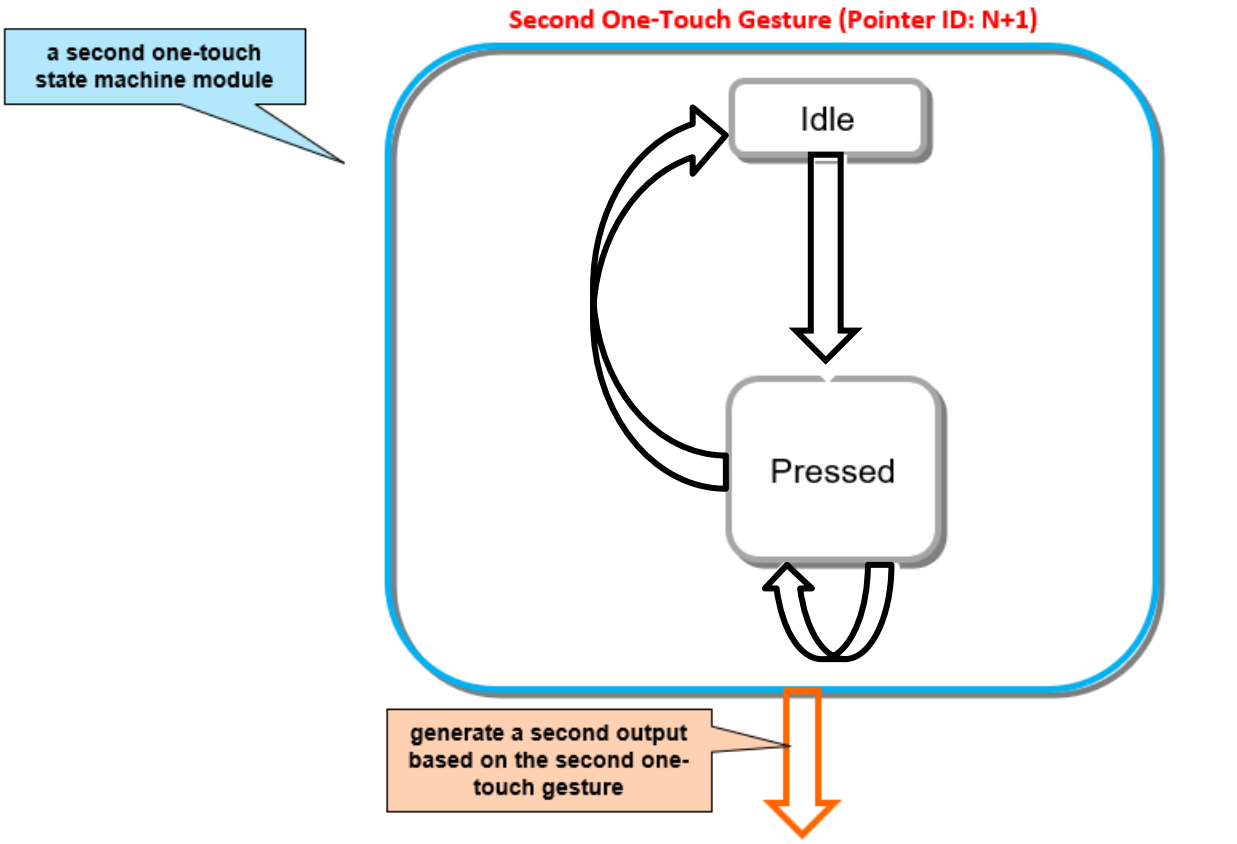
The interpretation of the contents of a MotionEvent varies signifi epending on the source class of the device.

On pointing devices with source class `InputDevice.SOURCE_CLAS OINTER` such as touch screens, the pointer coordinates specify absolute positions such as view X/Y coordinates. Each complete gesture is represented by a sequence of motion events with actions that describe pointer state transitions and movements. A gesture starts with a motion event with `ACTION_DOWN` that provides the location of the first pointer down. As each additional pointer that goes down or up, the framework will generate a motion event with `ACTION_POINTER_DOWN` or `ACTION_POINTER_UP` accordingly. Pointer movements are described by motion events with `ACTION_MOVE`. Finally, a gesture end either when the final pointer goes up as represented by a motion event with `ACTION_UP` or when gesture is canceled with `ACTION_CANCEL`.

generate a second output based on the second one-touch gesture

state machine module

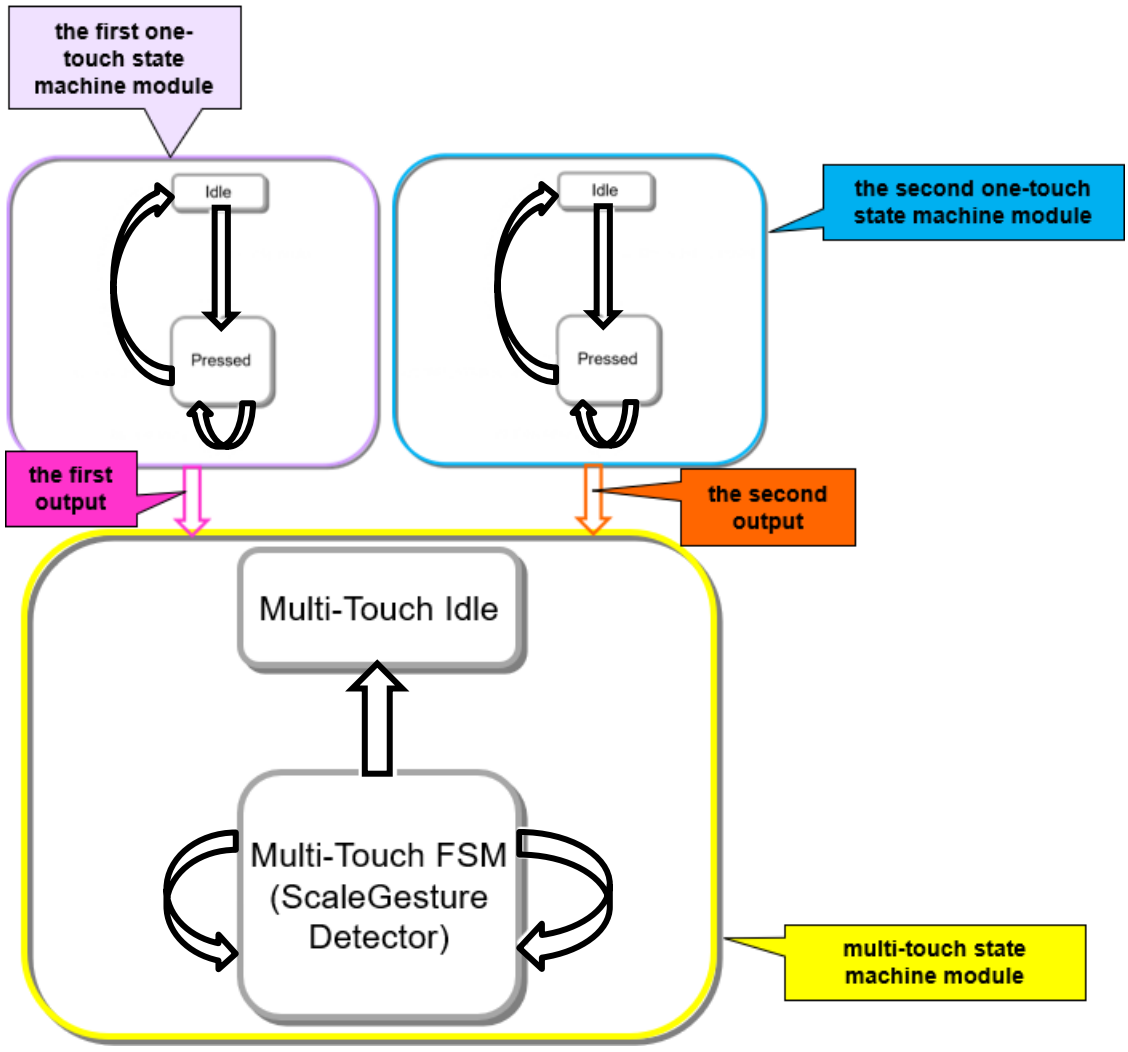
recognize a second one-touch gesture



[1h] "a multi-touch state-machine module operable to: receive, directly from the first one-touch state-machine module, the first output; receive, directly from the second one-touch state-machine module, the second output; and"

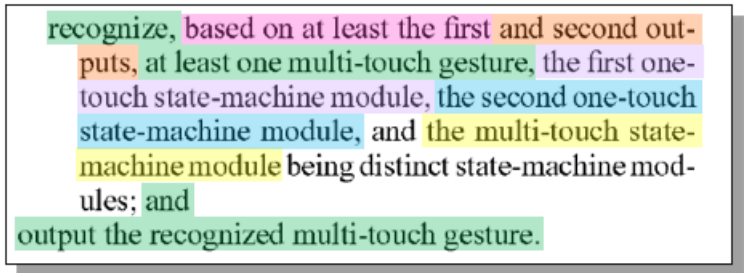
44. The Accused Products have a multi-touch state-machine module operable to: receive, directly from the first one-touch state-machine module, the first output; receive, directly from the second one-touch state-machine module, the second output:

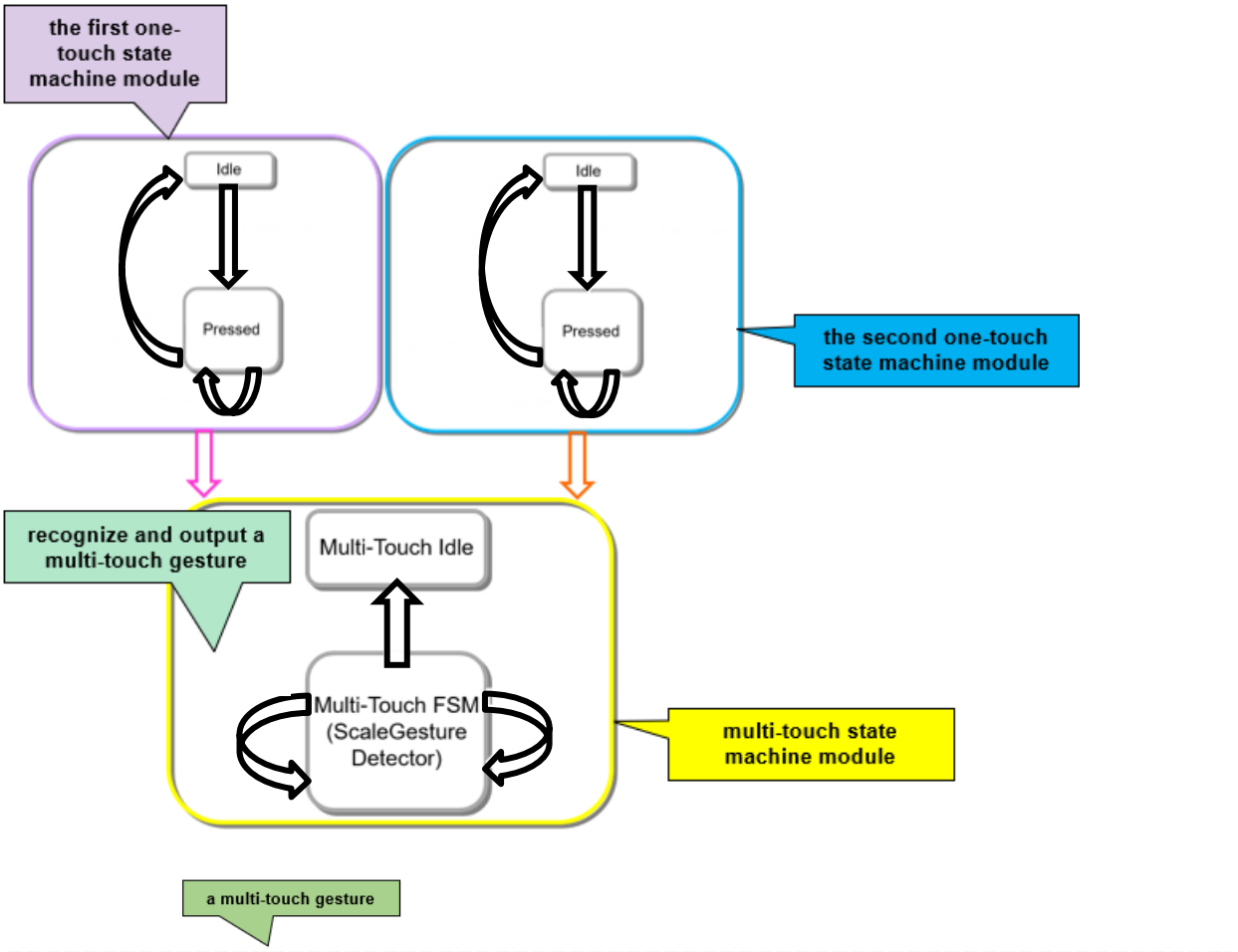
a multi-touch state-machine module operable to:
receive, directly from the first one-touch state-machine module, the first output;
receive, directly from the second one-touch state-machine module, the second output; and



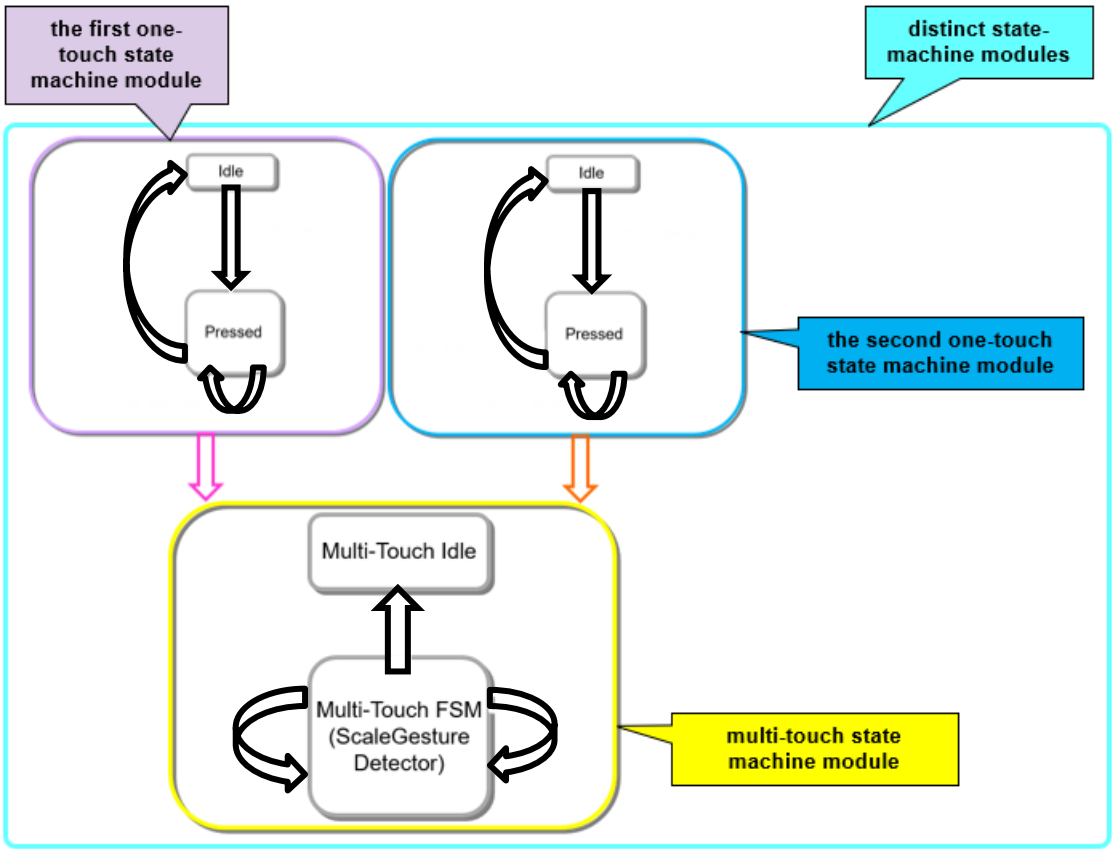
[1i] "recognize, based on at least the first and second outputs, at least one multi-touch gesture, the first one-touch state-machine module, the second one-touch state-machine module, and the multi-touch state-machine module being distinct state-machine modules; and output the recognized multi-touch gesture."

45. The Accused Products recognize, based on at least the first and second outputs, at least one multi-touch gesture, the first one-touch state-machine module, the second one-touch state-machine module, and the multi-touch state-machine module being distinct state-machine modules; and output the recognized multi-touch gesture**:**





https://r2.community.samsung.com/t5/Tech-Talk/Guide-to-Touchscreen-Gestures/td-p/4095444



46. Defendants also knowingly and intentionally induce and contribute to infringement of the '767 patent in violation of 35 U.S.C. §§ 271(b) and 271(c). Through the filing and service of this Complaint, Defendants have had knowledge of the '767 patent and the infringing nature of the Accused Products. Defendant SEC also has had knowledge of the '767 patent through the issuance of U.S. Patent No. 9,207,792 on December. 8, 2015 and assignment to SEC, which cites on its face the '767 patent. Despite this knowledge of the '767 patent, Defendants continue to actively encourage and instruct its customers to use and integrate the accused products in ways that directly infringe the '767 patent. Defendants do so knowing and intending that their customers will commit these infringing acts. Defendants also continue to make, use, offer for sale, sell, and/or import the Accused Products, despite their knowledge of the '767 patent, thereby specifically intending for and inducing its customers to infringe the '767

patent through the customers' normal and customary use of the Accused Products.

47. Defendants have infringed multiple claims of the '767 patent, including independent claim 1. By way of example only, the accused Samsung Galaxy S9 phone infringes an exemplary claim of the '767 patent, as in the description set forth above, which Solas provides without the benefit of information about the Accused Products obtained through discovery.

48. Defendants have known how the Accused Products are made and have known, or have been willfully blind to the fact, that making, using, offering to sell, and selling the Accused Products to their customers, would constitute willful infringement of the '767 patent. Those products imported into and sold within the United States include, without limitation, Samsung laptop computers, Galaxy tablets and phones.

49. Defendants have induced, and continue to induce, infringement of the '767 patent by actively encouraging others (including its customers) to use, offer to sell, sell, and import the Accused Products. On information and belief, these acts include providing information and instructions on the use of the Accused Products; providing information, education and instructions to its customers; providing the Accused Products to customers; and indemnifying patent infringement within the United States.

50. Solas has been damaged by Defendant's infringement of the '767 patent and is entitled to damages as provided for in 35 U.S.C. § 284, including reasonable royalty damages.

Jury demand.

51. Solas demands trial by jury of all issues.

Relief requested.

Solas prays for the following relief:

A. A judgment in favor of Solas that Defendants have infringed the ’144 patent and the ’767 patent, and that the ’144 patent and the ’767 patent are valid, enforceable, and patent-eligible;

B. A judgment and order requiring Defendants to pay Solas compensatory damages, costs, expenses, and pre- and post-judgment interest for its infringement of the asserted patents, as provided under 35 U.S.C. § 284;

C. A permanent injunction prohibiting Defendants from further acts of infringement of the ’144 patent and the ’767 patent;

D. A judgment and order requiring Defendants to provide an accounting and to pay supplemental damages to Solas, including, without limitation, pre-judgment and post-judgment interest;

E. A finding that this case is exceptional under 35 U.S.C. § 285, and an award of Solas’ reasonable attorney’s fees and costs; and

F. Any and all other relief to which Solas may be entitled.

Dated: March 22, 2021

Respectfully submitted,

/s/ Reza Mirzaie

Reza Mirzaie
CA State Bar No. 246953
Email: rmirzaie@raklaw.com
Marc Fenster
CA State Bar No. 181067
Email: mfenster@raklaw.com
Neil A. Rubin
CA State Bar No. 250761
Email: nrubin@raklaw.com
James S. Tsuei
CA State Bar No. 285530
Email: jtsuei@raklaw.com
RUSS AUGUST & KABAT
12424 Wilshire Blvd. 12th Floor
Los Angeles, CA 90025
Telephone: 310-826-7474

T. John Ward, Jr.
Texas State Bar No. 00794818
Email: jw@wsfirm.com
Claire Abernathy Henry
Texas State Bar No. 24053063
Email: claire@wsfirm.com
Andrea L. Fair
Texas State Bar No. 24078488
Email: andrea@wsfirm.com
WARD, SMITH & HILL, PLLC
PO Box 1231
Longview, Texas 75606-1231
(903) 757-6400 (telephone)
(903) 757-2323 (facsimile)

ATTORNEYS FOR PLAINTIFF, SOLAS OLED, LTD.