VISION 2020「From Domestic No.1 to Global Top 10」

SiW Touch Driver
v 2.13

2017.08.15

R&D / Touch Team

History

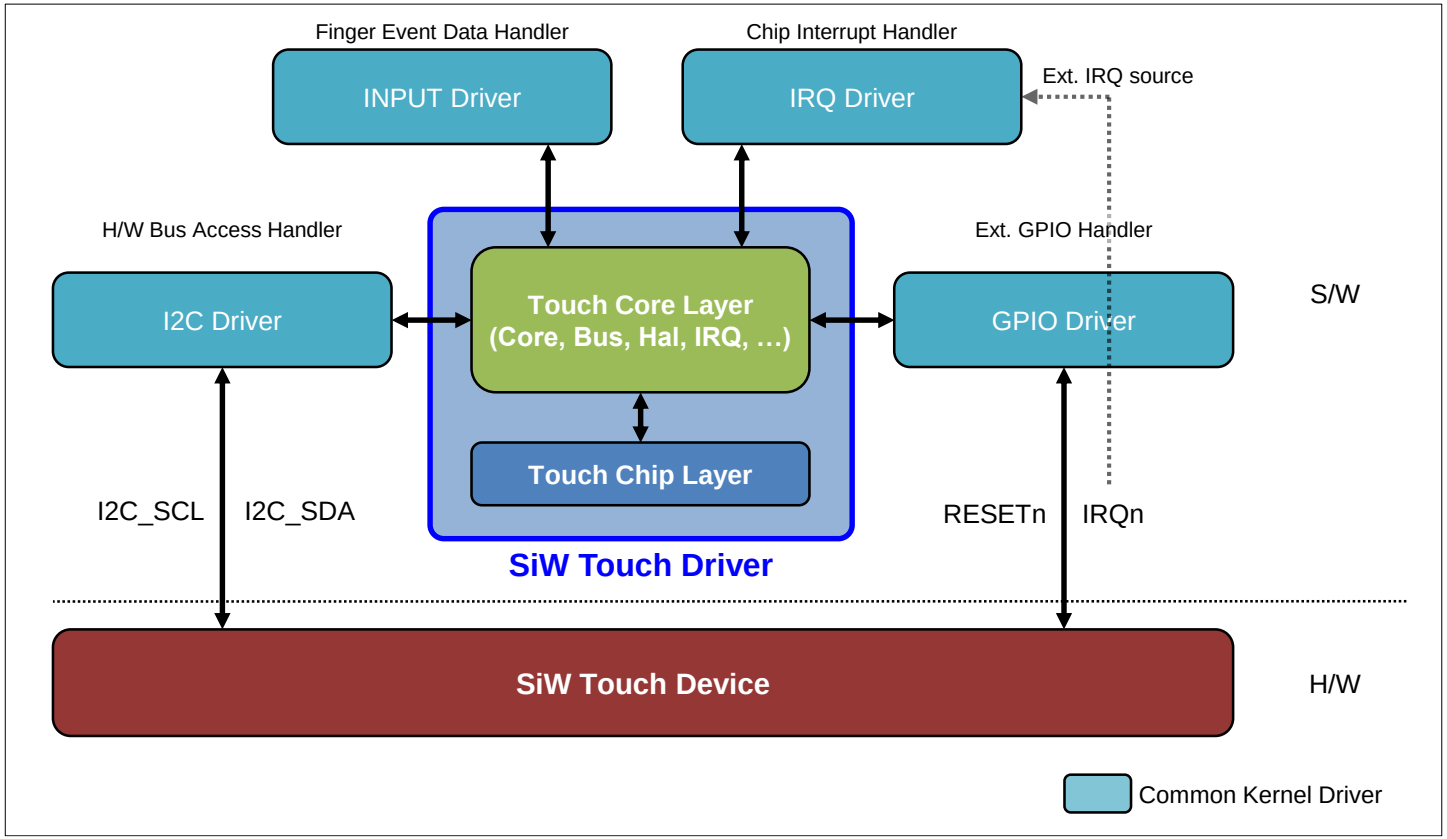
Version	Date	Description
1.0	2016.03.15	1st release
2.0	2016.04.15	Rebuild Driver Framework for HAL layer
2.07	2016.05.17	Add PRD, Watch
2.09	2016.05.30	3. Basic Register Setup Guide
2.11	2016.06.15	- 1.3 (2) IRQ Handler - I2C Protocol Example - 1.3 (3) FW Upgrade - 4. Flag
2.12	2016.07.29	- [Table. 1-1] Driver File List siw_touch_misc.c - 1.3 (3) FW Upgrade – Example - 1.3 (4) Version check
2.13	2017.08.15	- 1.1 Driver Architecture File name fixed : siw_touch_bus_event.c → siw_touch_event.c - 1.2 Initialization Flow Probe sequence is re-organized for TOUCH_USE_PROBE_INIT_LATE option - '3. Basic Register Setup Guide' and '4. Flag' are eliminated

Table Of Contents

1. Driver Operation

1.1 Driver Architecture

(1) Overview



[Fig. 1-1] Driver Relationship

1. Driver Operation

1.1 Driver Architecture

(2) SiW Touch Driver Files

Layer	Name	Description
Touch Core Layer	siw_touch.c	Touch Core
	siw_touch_bus.c	Touch Bus I/F main
	siw_touch_bus_i2c.c	Touch Bus I/F - I2C type
	siw_touch_bus_spi.c	Touch Bus I/F - SPI type
	siw_touch_event.c	Touch Input & Event control
	siw_touch_gpio.c	Touch GPIO control
	siw_touch_irq.c	Touch Interrupt control
	siw_touch_notify.c	Touch Notifier Chain
	siw_touch_of.c	Touch Device Tree analysis
	siw_touch_sysfs.c	Touch Sysfs control
	siw_touch_sys.c	Helper for Touch & System Inter-connection
	siw_touch_misc.c	Device node(/dev/{misc name}) for direct bus access
	siw_touch_hal.c	Touch HAL
	siw_touch_hal_abt.c	Touch HAL for ABT
	siw_touch_hal_prd.c	Touch HAL for PRD
	siw_touch_hal_sysfs.c	Touch HAL for Sysfs
	siw_touch_hal_watch.c	Touch HAL for WATCH
Touch Chip Layer	touch_xxxxx.c	Entry configuration for the chipset xxxxx
Build Files	Kconfig / Makefile	

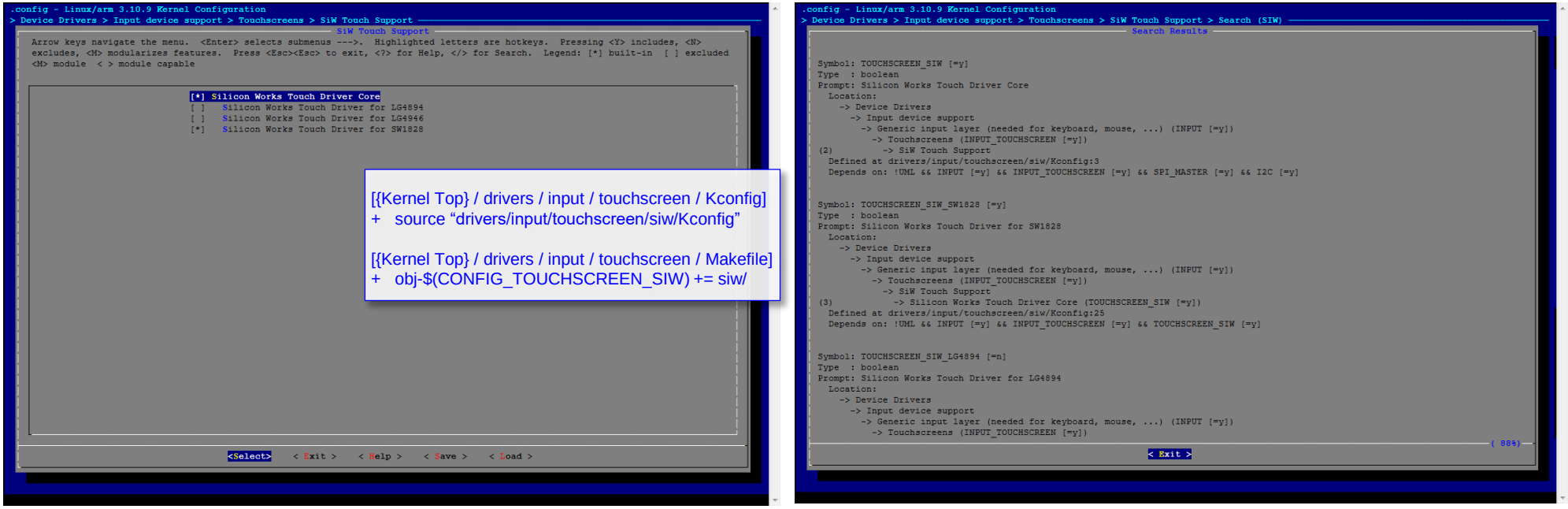
[Table. 1-1] Driver File List

* HAL : Hardware Abstraction Layer

1. Driver Operation

1.1 Driver Architecture

(2) SiW Touch Driver Files - Kconfig



[Fig. 1-2] Kconfig (example)

1. Driver Operation

1.1 Driver Architecture

(2) SiW Touch Driver Files - Test Environment

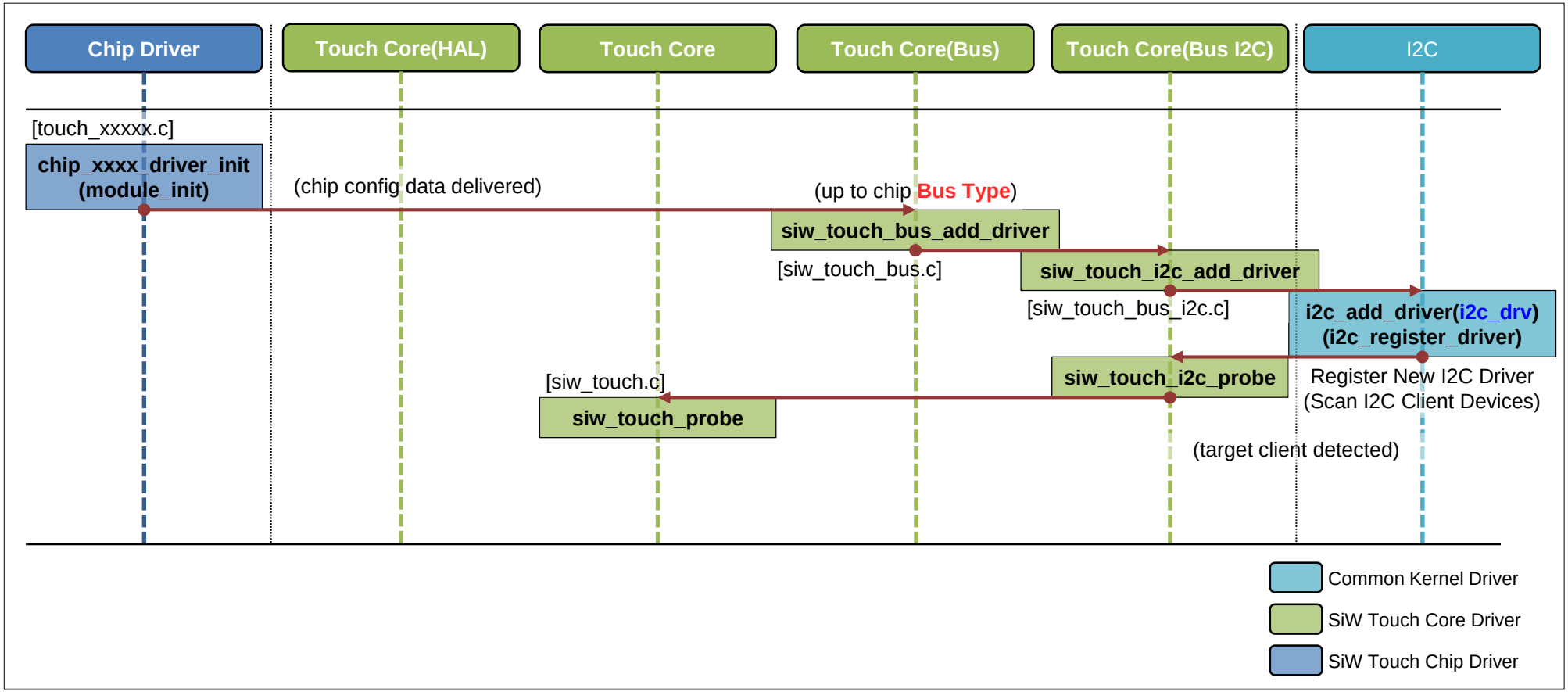
Test Environment		
H/W		Odroid-XU4(Exynos5422)
S/W	Platform Version	Android 4.4.4
		Kernel 3.10.9
	Driver Folder	{Kernel Top} / drivers / input / touchscreen / siw
		{Kernel Top} / include / linux / input : siw_touch_notify.h

[Table. 1-2] Test Environment

1. Driver Operation

1.2 Initialization Flow

(1) Probe Sequence - I2C (LG4894, LG4951, SW1828, SW49501, SW42101)

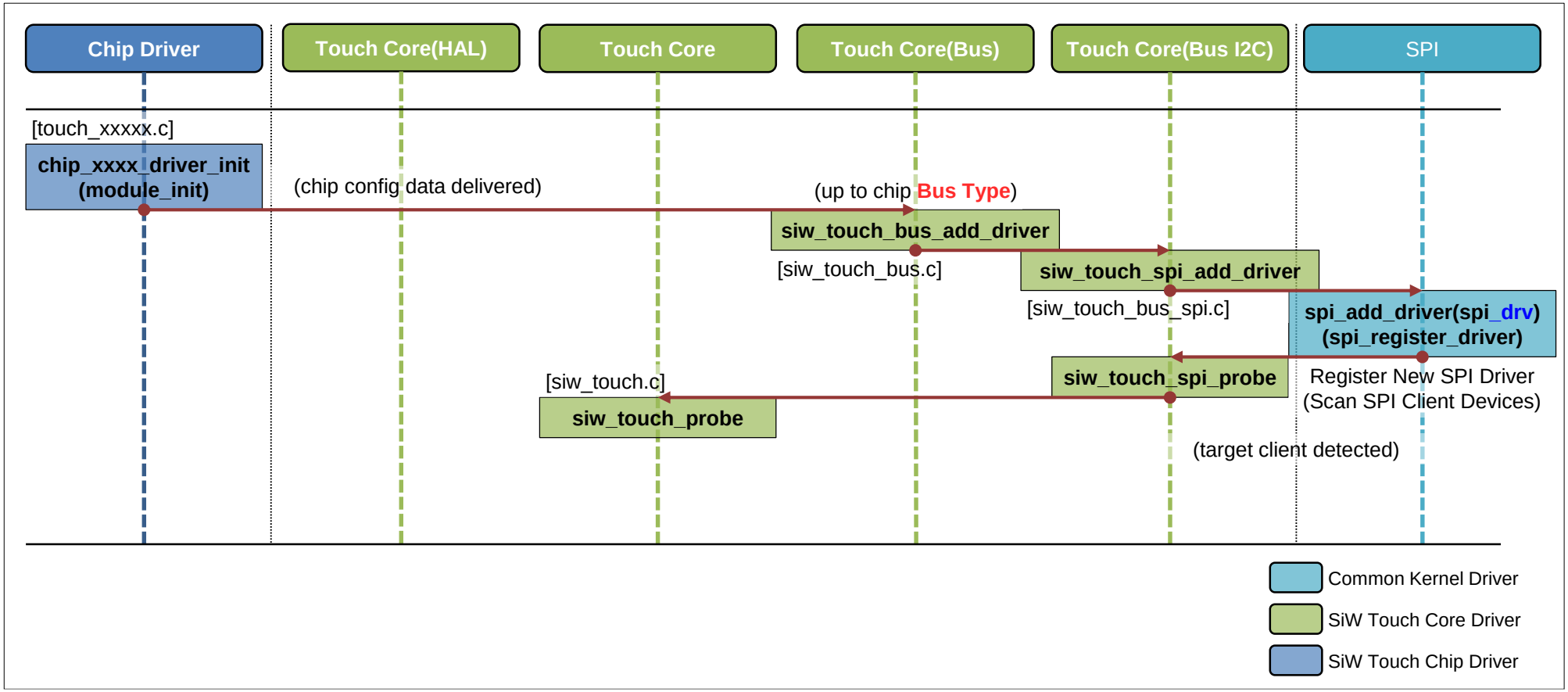


[Fig. 1-3] Initial Probe Sequence (I2C)

1. Driver Operation

1.2 Initialization Flow

(2) Probe Sequence - SPI (LG4895, LG4946, SW49407, SW49408)

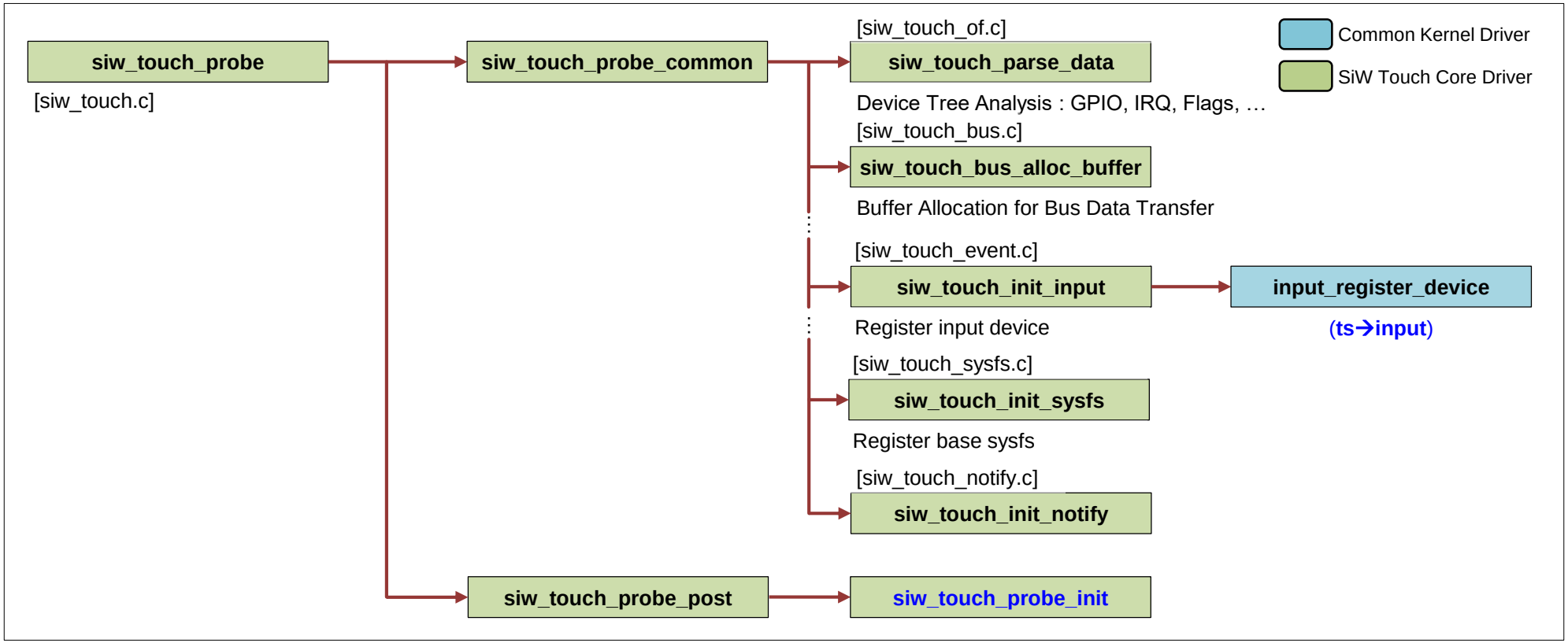


[Fig. 1-4] Initial Probe Sequence (SPI)

1. Driver Operation

1.2 Initialization Flow

(3) siw_touch_probe (1/2)

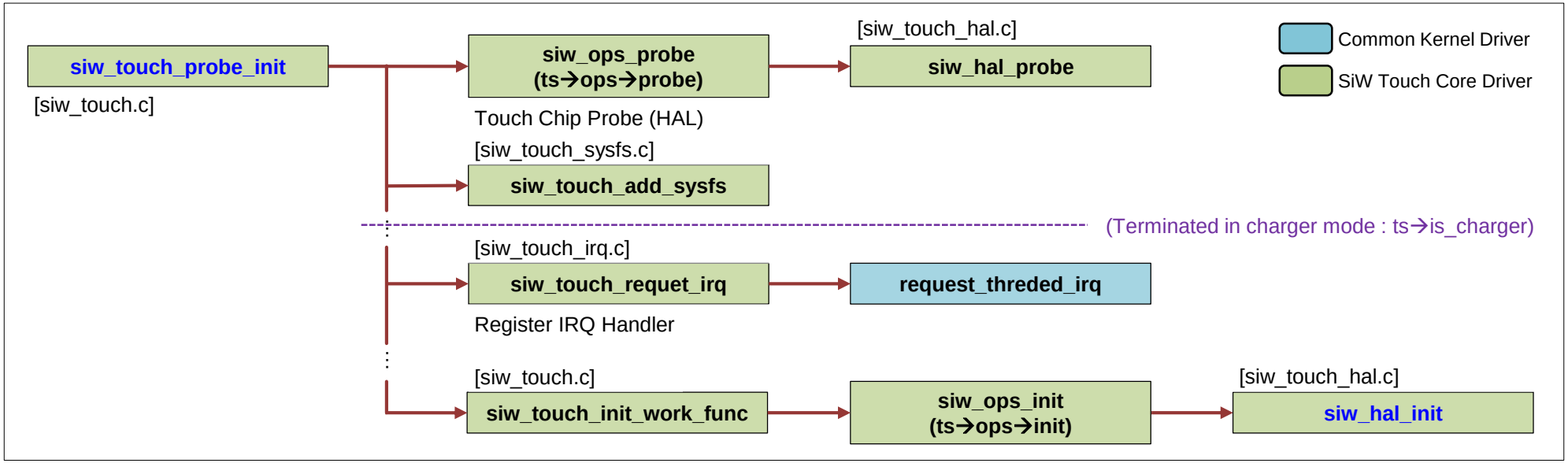


[Fig. 1-5-1] Inside operation of siw_touch_probe (1/2)

1. Driver Operation

1.2 Initialization Flow

(3) siw_touch_probe (2/2) - Actual HW access for touch device



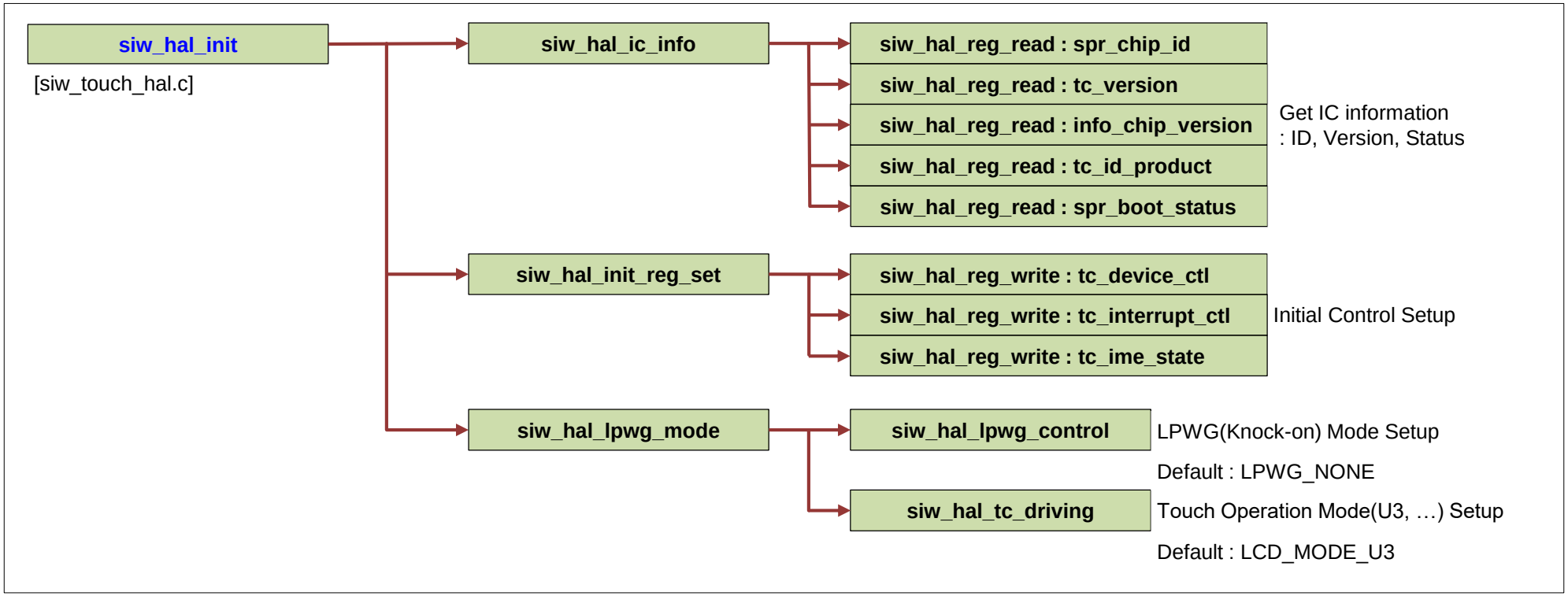
[Fig. 1-5-2] Inside operation of siw_touch_probe (2/2)

- The **siw_touch_probe_init** can be postponed by TOUCH_USE_PROBE_INIT_LATE option and the post processing(ts->init_late) can be triggered by either of the following
  (1) via notifier - int value = 0x1234; //any value for identification except the below values
  //0x0000 : not permitted
  //0x55AA : for sysfs
  //0xAA55 : feedback indicating init_late done
  siw_touch_atomic_notifier_call(LCD_EVENT_TOUCH_INIT_LATE, (void *)&signal);
  (2) via sysfs - echo 0x55AA > /sys/devices/virtual/input/siw_touch_input/init_late

1. Driver Operation

1.2 Initialization Flow

(4) siw_hal_init

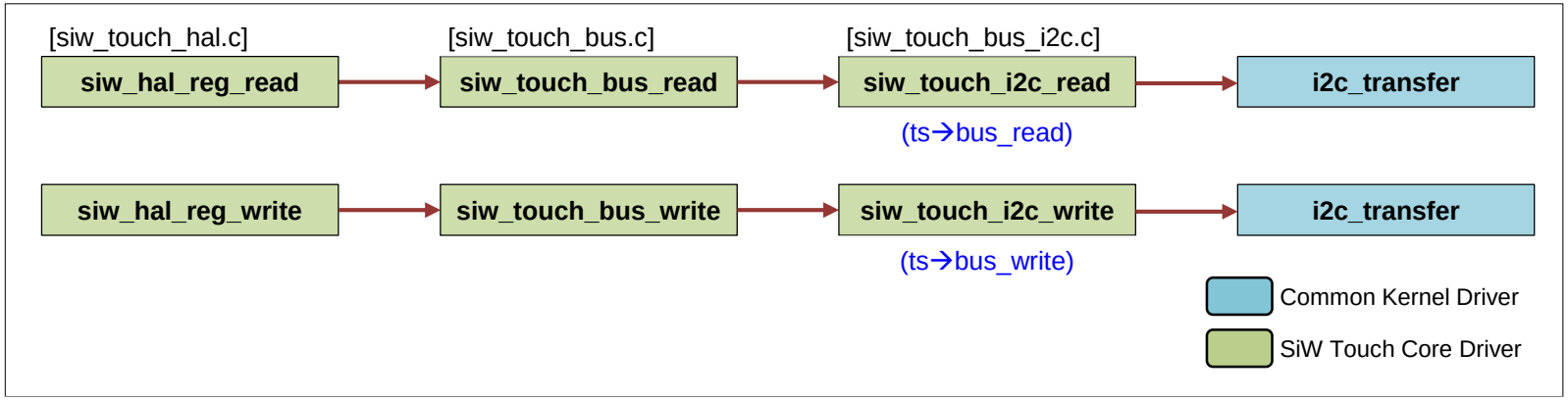


[Fig. 1-6] Inside operation of siw_hal_init

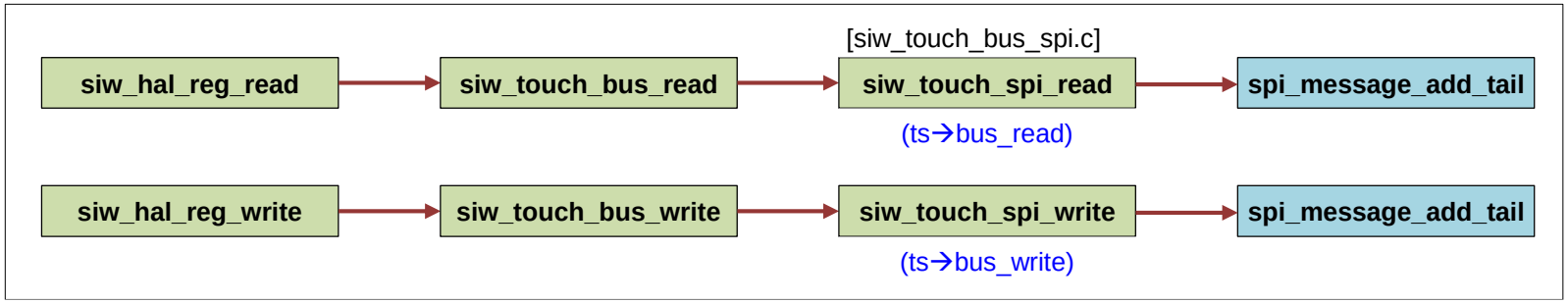
1. Driver Operation

1.3 Operation

(1) Bus Access



[Fig. 1-7] Bus Access Flow for I2C type

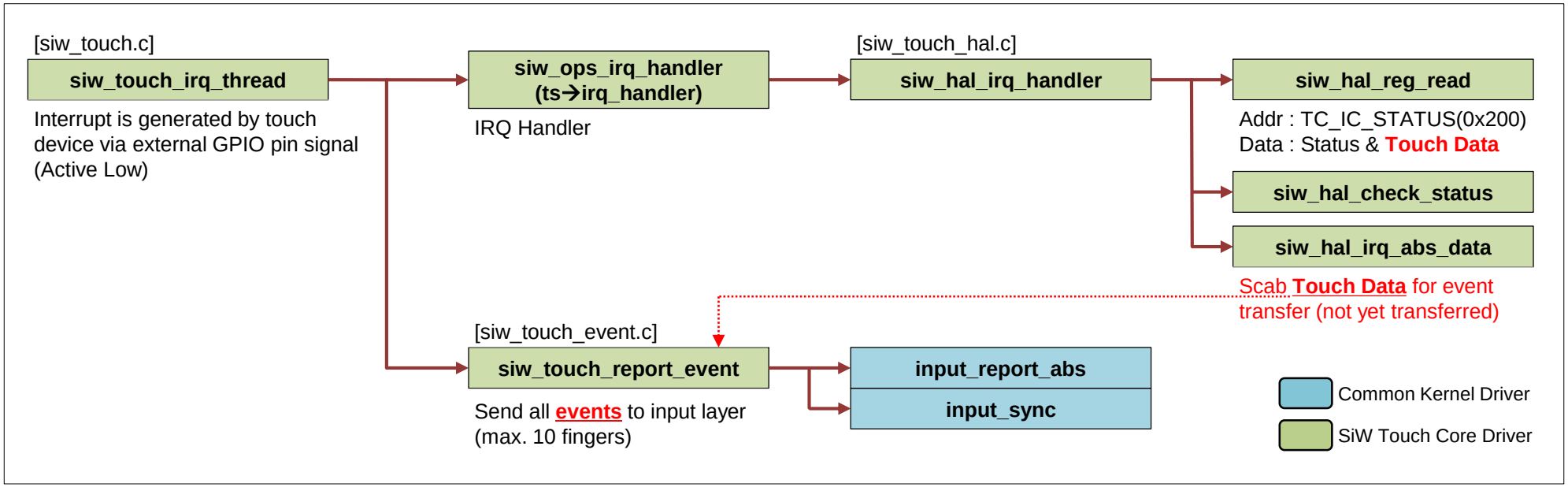


[Fig. 1-8] Bus Access Flow for SPI type

1. Driver Operation

1.3 Operation

(2) IRQ Handler (when touch event detected)



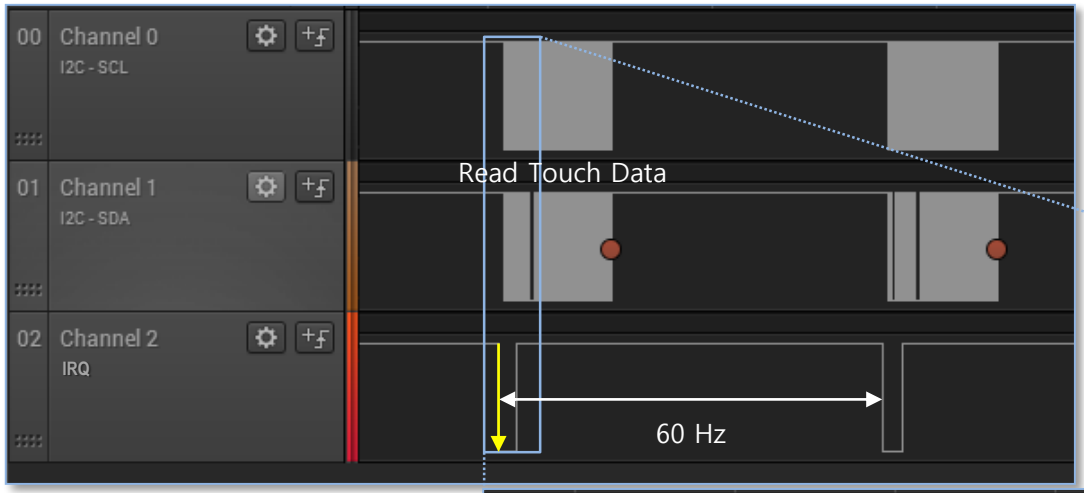
[Fig. 1-9] Interrupt Handling

- An appropriate external interrupt connection shall be guaranteed for the accuracy of this IRQ operation
- IRQ Flags
  Recommended flag setup value is **0x2002**((**IRQF_TRIGGER_FALLING**(0x02) | **IRQF_ONESHOT**(0x2000)), however, some problematic chipset may call handler routine twice at both edge, falling and rising.
  In this case, use **0x2008**((**IRQF_TRIGGER_LOW**(0x08) | **IRQF_ONESHOT**(0x2000)) instead of **0x2002**

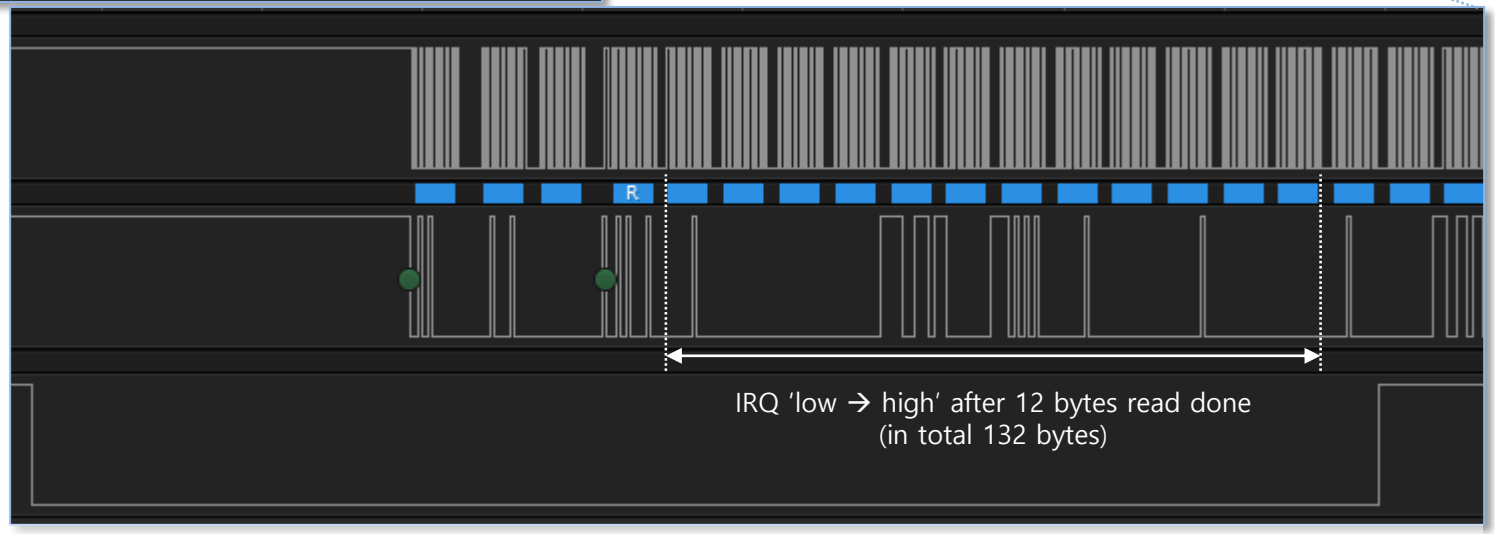
1. Driver Operation

1.3 Operation

(2) IRQ Handler - I2C protocol example



- The regular period of touch IRQ is always 60Hz.
- Reading time for 132 bytes data shall be terminated in 60Hz period or the IRQ sync distortion will happen.
- Reading data twice in single IRQ section is not permitted because 'invalid IRQ state' may be detected in 2nd reading

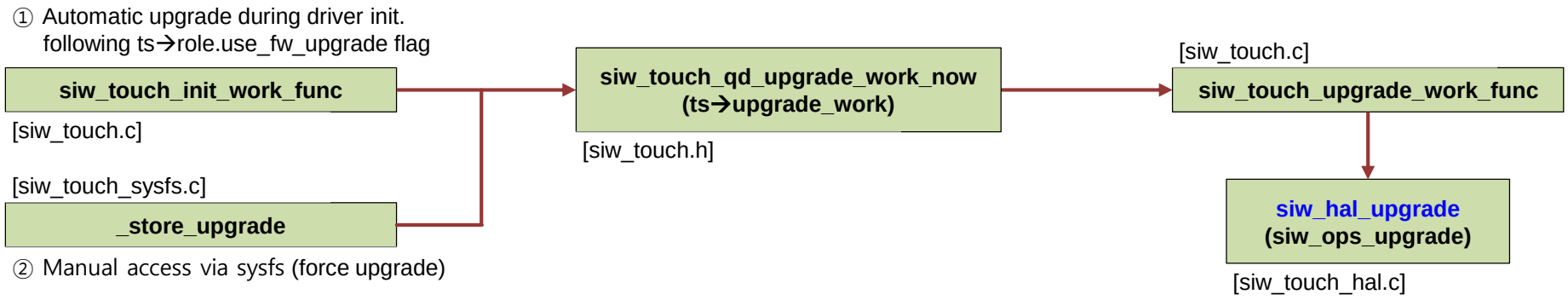


[Fig. 1-10] IRQ & I2C protocol

1. Driver Operation

1.3 Operation

(3) FW Upgrade (core layer)



1. If TOUCH_USE_FW_BINARY flag used
1-1 Default upgrade (through version comparison)
do upgarde using **binary header** link
1-2 echo {bin} > fw_upgrade
do force-upgrade using **binary header** link (same as 1-1)
1-3 echo /.../fw_img > fw_upgrade
do force-upgrade using **request_firmware** (relative path)
1-4 echo {root}/.../fw_img > fw_upgrade
do force-upgrade using **normal file open** control (absolute path)

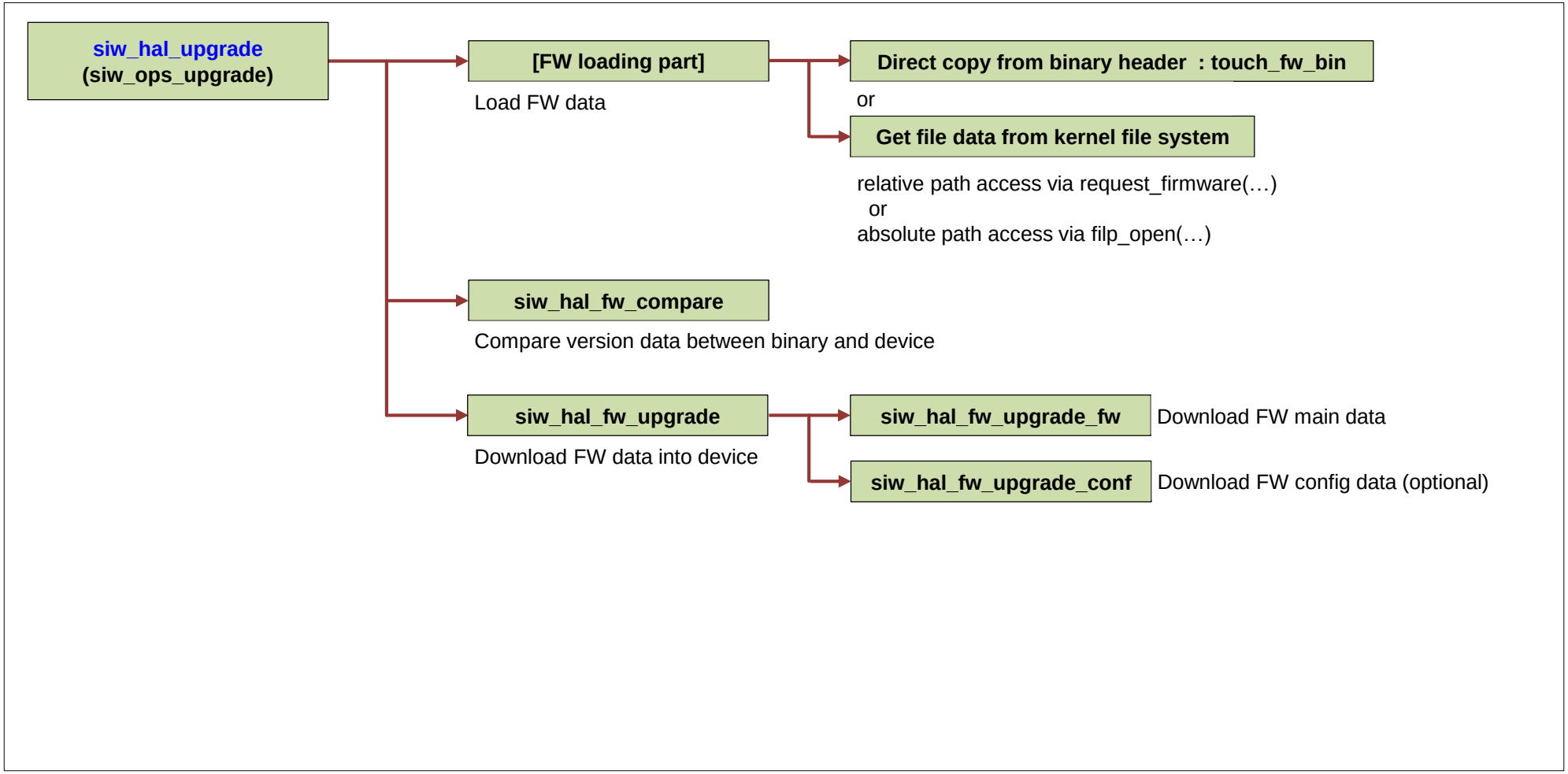
2. Else
2-1 Default upgrade (through version comparison)
do upgarde using request_firmware (relative path)
2-2 echo /.../fw_img > fw_upgrade
do force-upgrade using **request_firmware** (relative path)
2-3 echo {root}/.../fw_img > fw_upgrade
do force-upgrade using **normal file open** control (absolute path)

[Fig. 1-11] FW Upgrade (part 1)

1. Driver Operation

1.3 Operation

(3) FW Upgrade (hal layer)



[Fig. 1-12] FW upgrade (part 2)

1. Driver Operation

1.3 Operation

(3) FW Upgrade – Example (1/2)

- Command

$ echo {firmware image} > {siw sysfs folder path}/**fw_upgrade**

```
[Actual usage example]
▪ Relative path
$ echo /siw/sw49501/AURORA_10_01.img > /sys/device/virtual/input/siw_touch_input/fw_upgrade

▪ Absolute path : {root} is keyword to access absolute path
$ echo {root}/sdcard/firmware/siw/sw49501/AURORAO_10_01.img > /sys/device/virtual/input/siw_touch_input/fw_upgrade
```

- Parameters
  - ▪ Firmware image

    Firmware image = ‘Kernel firmware path / target image’ for relative path

    ‘General firmware path / target image’ for absolute path

    - Kernel firmware path = see ‘const fw_path[ ]’ in [drivers/base/firmware_class.c]

[SiW Test Platform case]

▪ Make external link in ‘/lib/firmware’ which is one of the kernel firmware path, to avoid read-only restriction

```
root@odroidxu3:/lib/firmware # ll
-rw-r--r-- root     root         8192 2016-02-11 08:19 rt2870.bin
drwxr-xr-x root     root              2016-02-11 08:19 rtlwifi
lrw-r--r-- root     root              2016-04-20 15:11 sdcard_firmware -> /sdcard/firmware
```

▪ Copy firmware image into ‘/sdcard/firmware’ which is write-permitted folder

```
root@odroidxu3:/sdcard/firmware/siw/sw49501 # ll
-rwxrwx--- root     sdcard_r    99840 2017-07-28 16:20 AURORA58_10_01.img
```

- ▪ siw sysfs folder path

  Driver default : /sys/device/virtual/input/siw_touch_input

  When it’s not default folder, you can find the ‘siw_touch_input’ under ‘i2c master device’ or ‘/sys/class/input/’ folder

1. Driver Operation

1.3 Operation

(3) FW Upgrade – Example (2/2)

- Command & Log example : Absolute path case

```
[14971.194803] [c3] siw_touch 5-0028: FW upgrade work func
…
[14971.208926] [c3] siw_touch 5-0028: fw type: FW_TYPE_1
[14971.214005] [c3] siw_touch 5-0028: getting fw from file
[14971.219165] [c3] siw_touch 5-0028: target fw: /sdcard/firmware/siw/sw49501/AURORA58_10_01.img (abs)
[14971.236021] [c3] siw_touch 5-0028: fw size: 99840
[14971.240508] [c3] siw_touch 5-0028: FW compare: bin-ver: 10.01 (AURORA58)(0)
[14971.247424] [c3] siw_touch 5-0028: FW compare: dev-ver: 1.01 (AURORA58)
[14971.254111] [c3] siw_touch 5-0028: FW compare: up 01, fup 08
[14971.464490] [c3] siw_touch 5-0028: ===== FW upgrade: start (0) =====
[14971.469479] [c3] siw_touch 5-0028: FW upgrade: idx 316h, dn 31Dh, code 3Fh
[14971.476263] [c3] siw_touch 5-0028: FW upgrade: include conf data
[14975.939869] [c0] siw_touch 5-0028: FW upgrade: boot check done
[14975.944787] [c0] siw_touch 5-0028: FW upgrade: conf_index: 1
[14977.689924] [c0] siw_touch 5-0028: FW upgrade: code check done
[14977.864917] [c0] siw_touch 5-0028: FW upgrade: conf check done
[14977.869779] [c0] siw_touch 5-0028: ===== FW upgrade: done (0) =====
[14977.875592] [c0] siw_touch 5-0028: SW49501 reset control(1)
…
[14978.106815] [c1] siw_touch 5-0028: [T] chip id 9501, version v10.01 (0x040E0A01, 0x00)
[14978.113360] [c1] siw_touch 5-0028: [T] product id AURORA58, flash boot idle(done), crc ok (0x00001322)
[14978.123579] [c0] siw_touch 5-0028: [SW49501] IC info is good: 14, 4
…
[14978.145635] [c0] siw_touch 5-0028: current driving mode is U3
[14978.152970] [c0] siw_touch 5-0028: DDI Display Mode[001Dh] = 0x00000003
[14978.159064] [c0] siw_touch 5-0028: TC Driving[0C03h] wr 0x00000185
[14978.205526] [c0] siw_touch 5-0028: command done: mode 3, running_sts 04h
…
[14978.241688] [c2] siw_touch 5-0028: SW49501 init done
…
```

1. Driver Operation

1.3 Operation

(4) Version Check

```
root@odroidxu3:/sys/devices/virtual/input/siw_touch_input # cat version
chip : SW49501
version : v10.01
revision : 0
product id : AURORA58

root@odroidxu3:/sys/devices/virtual/input/siw_touch_input # cat testmode_ver
v10.01

root@odroidxu3:/sys/devices/virtual/input/siw_touch_input # cat driver_data
=== Driver Data ===
Version : v2.19b
```

1. Driver Operation

1.4 Kernel Log (example)

(1) Probe message (1/2)

▪ 5-0028 : I2C adaptor(0x12C70000) is registered as I2C-5 and the slave address of the client(SW1828) is 0x28.

```
[14278.924451] [c1] siw_touch: SW49501 driver init - v2.19b
[14278.928485] [c1] siw_touch 5-0028: dev bus probe : 12c70000.i2c/i2c-5/5-0028
[14278.935405] [c1] siw_touch 5-0028: max finger : 10
[14278.940121] [c1] siw_touch 5-0028: chip type  : 0x0501
[14278.945246] [c1] siw_touch 5-0028: chip id    : 9501
[14278.950180] [c1] siw_touch 5-0028: chip name  : SW49501
[14278.955414] [c1] siw_touch 5-0028: drv name   : siw_touch
[14278.960789] [c1] siw_touch 5-0028: idrv name  : siw_touch_input
[14278.966670] [c1] siw_touch 5-0028: SiW Touch Probe
[14278.971437] [c1] siw_touch 5-0028: SW49501 quirks = 0x10050001
[14278.977266] [c1] siw_touch 5-0028: SW49501 ops is NULL : default ops selected
[14278.984398] [c1] siw_touch 5-0028: SW49501 reg quirks: [0] 0041h -> 007Eh
…
[14279.093170] [c7] siw_touch 5-0028: SW49501 reg quirks: ...
[14279.098639] [c7] siw_touch 5-0028: SW49501 reg quirks: t 50, m 0
[14279.104629] [c7] siw_touch 5-0028: flags(of) = 0x10000000 (0x10000000, 0x00000000)
[14279.112186] [c7] siw_touch 5-0028: of gpio  : reset-gpio(0x1), 23
[14279.118253] [c7] siw_touch 5-0028: of gpio  : irq-gpio, 22
[14279.123722] [c7] siw_touch 5-0028: irqflags(of) = 0x00002002 (0x00002002, 0x00002002)
…
[14279.218117] [c7] siw_touch 5-0028: caps max_x           = 2160
[14279.223932] [c7] siw_touch 5-0028: caps max_y           = 3840
[14279.230529] [c7] input: siw_touch_input as /devices/virtual/input/input6
[14279.237003] [c7] input input6: input device[i2c-5/5-0028 - siw_touch_input] registered (2160, 3840, 255, 15, 15, 1, 10)
[14279.247953] [c7] siw_touch 5-0028: [opt summary]
[14279.251816] [c7] siw_touch 5-0028:  f_info_more     : 0
…
[14279.340285] [c7] siw_touch 5-0028:  t_chk_fault     : 0
[14279.346560] [c7] siw_touch 5-0028: trigger gpio reset
[14279.350526] [c7] siw_touch 5-0028: tci info[TCI_1] tap_count 2, min_intertap 0, max_intertap 50
…
[14279.410393] [c7] siw_touch 5-0028: swipe info[SWIPE_R] distance 5, ratio_thres 100, ratio_distance 2
…
```

1. Driver Operation

1.4 Kernel Log (example)

(1) Probe message (2/2)

▪ 5-0028 : I2C adaptor(0x12C70000) is registered as I2C-5 and the slave address of the client(SW1828) is 0x28.

```
…
[14279.689974] [c0] siw_touch 5-0028: prd: sd_test_flag 7000511h, lpwg_sd_test_flag 0h
[14279.696228] [c0] siw_touch 5-0028: prd: param: row 32, col 18
…
[14279.776025] [c5] siw_touch 5-0028: prd: buffer allocted: dd3d6000(5920)
[14279.782861] [c5] siw_touch 5-0028: threaded irq request done(512, siw_touch, 0x2002)
[14279.790323] [c5] siw_touch 5-0028: irq(512) disabled
[14279.795276] [c5] siw_touch 5-0028: probe(normal) done
//ts->init_work
[14280.006810] [c0] siw_touch 5-0028: report type  : 1
[14280.010273] [c0] siw_touch 5-0028: status type  : 2
[14280.015408] [c0] siw_touch 5-0028: status mask  : 8050A6E0h
…
[14280.059852] [c0] siw_touch 5-0028: [T] chip id 9501, version v10.01 (0x040E0A01, 0x00)
[14280.067473] [c2] siw_touch 5-0028: [T] product id AURORA58, flash boot idle(done), crc ok (0x00001322)
[14280.077890] [c7] siw_touch 5-0028: [SW49501] IC info is good: 14, 4
[14280.084337] [c7] siw_touch 5-0028: lpwg resume: mode 0, screen 1
[14280.088955] [c7] siw_touch 5-0028: lpwg resume: screen(3)
[14280.094793] [c7] siw_touch 5-0028: lpwg_control mode = 0
[14280.099614] [c7] siw_touch 5-0028: current driving mode is U3
[14280.105861] [c7] siw_touch 5-0028: DDI Display Mode[001Dh] = 0x00000003
[14280.112336] [c7] siw_touch 5-0028: TC Driving[0C03h] wr 0x00000185
[14280.140908] [c7] siw_touch 5-0028: [0] ic debug: s 0F5480E1h / m 4h, l Ch, t 1Eh (0C00001Eh)
[14280.147887] [c7] siw_touch 5-0028: [0] ic debug: log 08000040h 00000000h 00000001h
[14280.155434] [c7] siw_touch 5-0028: command done: mode 3, running_sts 01h
[14280.162588] [c7] siw_touch 5-0028: glove_en[0C62h]: OFF(00000000h)
[14280.168694] [c7] siw_touch 5-0028: grab_en[0C64h]: OFF(00000000h)
[14280.183939] [c7] siw_touch 5-0028: lpwg resume(1, 0): lcd_mode 3, driving_mode 3
[14280.191274] [c7] siw_touch 5-0028: SW49501 init done
[14280.196215] [c7] siw_touch 5-0028: irq(512) enabled
[14280.201090] [c7] siw_touch 5-0028: irq(512) wake enabled
```

1. Driver Operation

1.4 Kernel Log (example)

(2) System Information

```
root@odroidxu3:/sys/bus/i2c/devices/5-0028 # ll
lrwxrwxrwx root     root              2016-04-12 08:48 driver -> ../../../../bus/i2c/drivers/siw_touch
-r--r--r-- root     root         4096 2016-04-12 08:40 modalias          // = i2c:sw1828
-r--r--r-- root     root         4096 2016-04-12 08:40 name              // = sw1828
drwxr-xr-x root     root              2016-04-12 08:40 power
lrwxrwxrwx root     root              2016-04-12 08:40 subsystem -> ../../../../bus/i2c
-rw-r--r-- root     root         4096 2016-04-12 08:40 uevent

root@odroidxu3:/sys/device/virtual/input # ll
drwxr-xr-x root     root              2016-04-12 08:40 input2
drwxr-xr-x root     root              2016-04-12 08:41 input4
drwxr-xr-x root     root              2016-04-12 08:40 mice
drwxr-xr-x root     root              2016-04-12 08:47 siw_touch_input


root@odroidxu3:/proc/bus/input # cat devices
…
I: Bus=0018 Vendor=abcd Product=9876 Version=1234
N: Name="siw_touch_input"
P: Phys=i2c-5/5-0028 - siw_touch_input
S: Sysfs=/devices/virtual/input/input7
U: Uniq=
H: Handlers=event4
B: PROP=2
B: EV=9
B: ABS=67c8000 0
```

2. Device Tree (example for I2C)

- Definition of I2C client device for SW49501 (refer to DTS example files for more information)

```
&i2c_1 {                                    // indicates parent device : I2C_1 adapter block
   sw49501@28 {                             // define new client device(sw49501) and slave addr. is 0x28
      status = "okay";
      compatible = "siw,sw49501";           // compatible name (see touch_xxxxxx.c)
      reg = <0x28>;                         // slave addr. : 0x28
      interrupt-parent = <&gpx1>;           // interrupt source : GPIO group gpx1
      interrupts = <6 0x02>;                // index 6(0~7) in gpx1 external interrupts
      irqflags = <0x2002>;                  // IRQF_ONESHOT(0x2000) | IRQF_TRIGGER_FALLING(0x2)
      chip_flags = <0>;
      reset-gpio = <&gpx1 7 GPIO_ACTIVE_LOW>;          // index 7 in gpx1
      irq-gpio = <&gpx1 6 GPIO_ACTIVE_LOW>;            // index 6 in gpx1

      /* Caps */
      max_x = <2160>;
      max_y = <3840>;
      max_pressure = <0xff>;
      max_width = <15>;
      max_orientation = <1>;
      max_id = <10>;
      /* role */
      hw_reset_delay = <210>;
      sw_reset_delay = <90>;
      use_lpwg = <0>;
      use_lpwg_test = <0>;
      /* firmware */
      use_firmware = <1>;          // enable firmware control
      use_fw_upgrade = <1>;        // auto-update during driver initialization
      fw_image = "siw/sw49501/AURORA58_1_01.img";          // in android -> /lib/firmware/siw/..
      //absoulte path
      prd_in_file = "/sdcard/siw/sw1828_test_spec_V0.1.txt";
      prd_in_file_m = "/sdcard/siw/sw1828_test_spec_mfts_V0.1.txt";
      prd_out_file = "/sdcard/siw/touch_self_test.txt";
   };
};
```

(mandatory)

[gpx1 definition in exynos5422 pinctrl device tree]

```
...
   pinctrl@13400000 {
   …
      gpx1: gpx1 {
         …
                  interrupt-controller;
                  interrupt-parent = <&combiner>;
                  #interrupt-cells = <2>;
                  interrupts = <28 0>, <28 1>, <29 0>, <29 1>,
                               <30 0>, <30 1>, <31 0>, <31 1>;
      };
      …
   };
…
```

- This example has been established based on odroidx-xu4(exynos5422) platform
- The detail configuration shall be modified up to main chipset.

2. Device Tree (example for SPI)

- Definition of SPI client device for LG4895 (refer to DTS example files for more information) (1/2)

```
&spi_1 {                                    // indicates parent device : SPI_1 block
    status = “okay”;
    samsung,spi-src-clk = <0>;
    num-cs = <1>;

    lg4895@0 {                              // define new spi device(lg4895)
        status = "okay";
        compatible = "siw,lg4895";          // compatible name (see touch_xxxxxx.c)
        reg = <0>;
        interrupt-parent = <&gpx1>;         // interrupt source : GPIO group gpx1
        interrupts = <6 0x02>;              // index 6(0~7) in gpx1 external interrupts
        irqflags = <0x2002>;                // IRQF_ONESHOT(0x2000) | IRQF_TRIGGER_FALLING(0x2)
        chip_flags = <0>;
        reset-gpio = <&gpx1 7 GPIO_ACTIVE_LOW>;          // index 7 in gpx1
        irq-gpio = <&gpx1 6 GPIO_ACTIVE_LOW>;            // index 6 in gpx1

        /* Caps */
        max_x = <800>;
        max_y = <480>;
        max_pressure = <0xff>;
        max_width = <15>;
        max_orientation = <1>;
        max_id = <10>;
        /* role */
        hw_reset_delay = <210>;
        sw_reset_delay = <90>;
        use_lpwg = <0>;
        use_lpwg_test = <0>;
        /* firmware */
        use_firmware = <1>;          // enable firmware control
        use_fw_upgrade = <1>;        // auto-update during driver initialization
        fw_image = "siw/lg4895/L0W49P1_1_13.img.img";          // in android -> /lib/firmware/siw/..
        …
```

(mandatory)

- This example has been established based on odroidx-xu4(exynos5422) platform
- The detail configuration shall be modified up to main chipset.

2. Device Tree (example for SPI)

- Definition of SPI client device for LG4895 (refer to DTS example files for more information) (2/2)

```
&spi_1 {
    …
    lg4895@0 {
        …
        (after /* firmware */)

        //absoulte path
        prd_in_file = "/sdcard/siw/lg4895_test_spec_V0.1.txt";
        prd_in_file_m = "/sdcard/siw/lg4895_test_spec_mfts_V0.1.txt";
        prd_out_file = "/sdcard/siw/touch_self_test.txt";

        controller-data {
            cs-gpio = <&gpa2 5 GPIO_ACTIVE_LOW>;
            samsung,spi-feedback-delay = <0>;
            samsung,spi-chip-select-mode = <0>;      //MANUAL_CS_MODE = 0, AUTO_CS_MODE = 1,
        };
};
```

- This example has been established based on odroidx-xu4(exynos5422) platform
- The detail configuration shall be modified up to main chipset.