



FreeBSD and Touchscreens: Playing with your fingers.

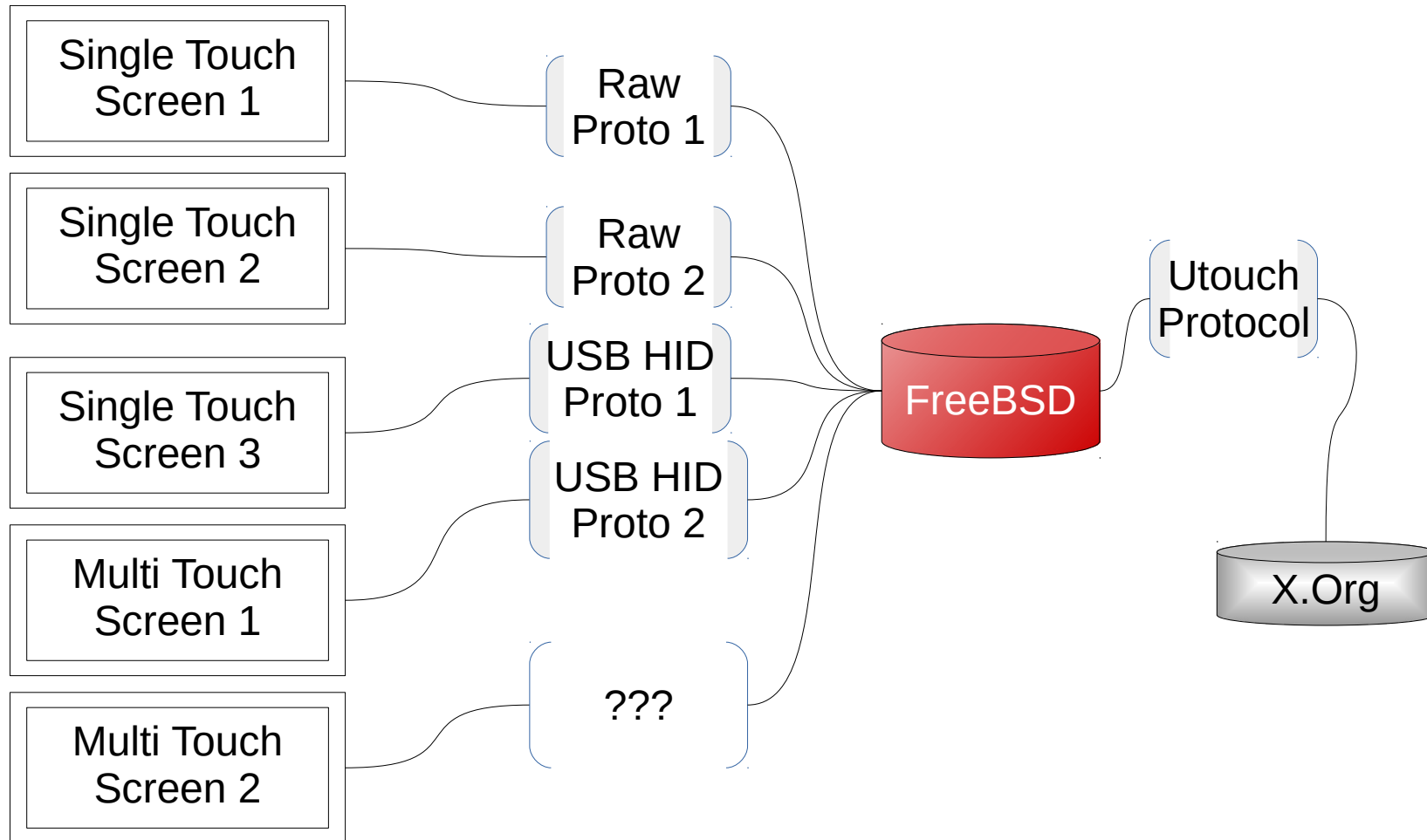
Roberto Fernández Cueto
System Developer

Bally Wulff Games & Entertainment GmbH

Introduction

- What does Bally Wulff produces?
 - ▶ Slot Machines
- Why do we need a custom driver?
 - ▶ There was no driver for FreeBSD in 2008.
- Are all the touch screen the same?
 - ▶ No, they implement different protocols and features.

Design





Kernel module

- Output protocol:

```
struct utouch_coord {
    uint16_t x;
    uint16_t y;
    uint16_t pressure;
    uint16_t button;
};
struct utouch_proto {
    char sig[2];
    uint8_t version;
    uint8_t ssize;
    uint8_t mode;
    struct utouch_coord coord;
    uint8_t buttons;
    uint8_t res[1];
};
```



Kernel module

- Decoder function:

```
struct utouch_hid {  
    struct hid_location    loc;  
    uint32_t    flags;  
    uint8_t    id;  
};  
  
typedef uint8_t (touchdecode_t)(  
    char    buf[],  
    struct utouch_coord * coord,  
    struct utouch_hid ** locations  
);
```



Kernel module

- Status structure.

```
struct utouch_status {  
    struct utouch_coord    poscurr;  
    struct utouch_coord    poslast;  
    struct utouch_coord    posmin;  
    struct utouch_coord    posmax;  
    uint8_t                buttons;  
    uint16_t               filtered;  
};
```



Kernel module

- Softc structure.

```
struct utouch_softc {  
    struct mtx          mtx;  
    struct usb_device *  udev;  
    struct usb_callout  callout;  
    struct usb_xfer *   xfer[2];  
    struct usb_fifo_sc  fifo;  
    uint32_t            flags;  
    struct utouch_status touch;  
    touchdecode_t *     decoder;  
    struct utouch_hid ** loc_array;  
    uint16_t            loc_array_size;  
};
```



X.Org module

- Main problem:
 - ▶ Coordinates not adjusted to the resolution.
 - ▶ A calibration will be needed.
 - ▶ The calibration needs to know the resolution of the display.
- Additional problems:
 - ▶ All the changes on the touch screen will be received.
 - ▶ There is always changes on the screen
 - ▶ The input must be filtered.
 - ▶ The `xf86PostMotionEvent`, `xf86PostButtonEvent` are not enough for a multitouch screen.
 - = `Xf86PostTouchEvent` – `TouchBegin`, `TouchUpdate` & `TouchEnd`



X.Org module

- Solutions:
 - ▶ Linear Transformation

$$\begin{pmatrix} X_s \\ Y_s \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \end{pmatrix} \cdot \begin{pmatrix} X_i \\ Y_i \\ K \end{pmatrix} \begin{cases} X_s = \text{Screen position } X \\ Y_s = \text{Screen position } Y \\ X_i = \text{Input position } X \\ Y_i = \text{Input position } Y \end{cases}$$

- Three points for calibration = 6 equations

$$(X_{s_0}, Y_{s_0}) = (12.5\%, 12.5\%)$$

$$(X_{s_1}, Y_{s_1}) = (50\%, 87.5\%)$$

$$(X_{s_2}, Y_{s_2}) = (87.5\%, 50\%)$$



X.Org module

- We do not filter during calibration.
 - ▶ But we cannot take only a point for the calibration.
 - ▶ We take the average of points.

$$X_{i_n} = \frac{1}{N} \cdot \sum_{j=0}^N X_{j i_n}$$

$$Y_{i_n} = \frac{1}{N} \cdot \sum_{j=0}^N Y_{j i_n}$$



X.Org module

$$K = \left((X_{i_0} - X_{i_2}) \cdot (Y_{i_1} - Y_{i_2}) \right) - \left((X_{i_1} - X_{i_2}) \cdot (Y_{i_0} - Y_{i_2}) \right)$$

If $K == 0$ Then $a = e = 1$ && $b = d = 0$

$$a = \frac{(X_{s_0} - X_{s_2}) \cdot (Y_{i_1} - Y_{i_2}) - (X_{s_1} - X_{s_2}) \cdot (Y_{i_0} - Y_{i_2})}{K}$$

$$b = \frac{(X_{i_0} - X_{i_2}) \cdot (X_{s_1} - X_{s_2}) - (X_{i_1} - X_{i_2}) \cdot (X_{s_0} - X_{s_2})}{K}$$

$$c = \frac{Y_{i_0} \cdot (X_{i_2} \cdot X_{s_1} - X_{i_1} \cdot X_{s_2}) + Y_{i_1} \cdot (X_{i_0} \cdot X_{s_2} - X_{i_2} \cdot X_{s_2}) + Y_{i_2} \cdot (X_{i_1} \cdot X_{s_0} - X_{i_0} - X_{s_1})}{K}$$

$$d = \frac{(Y_{s_0} - Y_{s_2}) \cdot (Y_{i_1} - Y_{i_2}) - (Y_{s_1} - Y_{s_2}) \cdot (Y_{i_0} - Y_{i_2})}{K}$$

$$e = \frac{(X_{i_0} - X_{i_2}) \cdot (Y_{s_1} - Y_{s_2}) - (X_{i_1} - X_{i_2}) \cdot (Y_{s_0} - Y_{s_2})}{K}$$

$$f = \frac{Y_{i_0} \cdot (X_{i_2} \cdot Y_{s_1} - X_{i_1} \cdot Y_{s_2}) + Y_{i_1} \cdot (X_{i_0} \cdot Y_{s_2} - X_{i_2} \cdot Y_{s_2}) + Y_{i_2} \cdot (X_{i_1} \cdot Y_{s_0} - X_{i_0} - Y_{s_1})}{K}$$



X.Org module

- How do we tell the driver that we want to calibrate?
 - ▶ Signals:
 - ▷ $SIGCAL0 = SIGRTMIN + 1$
 - ◆ Exit and save calibration.
 - ◆ Unpredictable behavior expected when the calibration was not completed.
 - ▷ $SIGCAL1 = SIGRTMIN + 2$
 - ◆ Enter calibration mode.
 - ◆ Go to the next calibration state ignoring normal flow.
 - ◆ Usefull to go to the beginning state of the calibration when the driver is in a wrong status.
 - ▷ $SIGRESET = SIGRTMIN + 3$
 - ◆ Sends the reset signal to the kernel driver
 - ◆ Some devices needs a reset.

X.Org module

- UtouchDevice:

```
struct Point {  
    int x, y;  
};  
  
struct UtouchDevice {  
    ...  
    struct UCalLinear      utcLinear;  
    struct Point           fakeCalibrationButton;  
    unsigned int           buttons;  
    struct coord *         button_states;  
    int                    filter_limit;  
};
```



Questions?