Samsung Flip

Hardware and Technology Specifications

Sarthak Garg

Basic Physical Specifications

- Panel : 55” 60 Hz LED
- Body Dimension(mm) : 1302.6 X 805.5 X 52.0
- Weight of package: 34.9 kg
- Typical Power Consumption: 100 W/h
- Recommended Humidity : 10% - 80%

Basic Internal Specifications

- Processor: CA72 Quad(1.7 GHz)
- Clock Speed: 1.7 GHz CPU Quad
- Operating System: Samsung Proprietary OS (VDLinux)
- Storage: 8GB (2.65GB Occupied by O/S, 5.35GB Available)

Ways to connect

- HDMI
- Screen Mirroring
- Wifi-out connectivity
- NFC
- USB

Primary Sensors

- Hall Sensor (Wakes up the display when passive pen is removed from the holder)
- Accelerometer (Detects rotation of frame and adjusts UI accordingly)
- Proximity Sensor (Wakes up screen when senses a person approaching close)

Now let’s get into the details.

Display

Resolution : 3840 X 2160
Response Time : 8ms

Screen

- UHD Display
  - Better resolution which supports screen mirroring from small devices.
- Four Width object recognition
  - It can detect passive object of 4 different sizes 2 / 4 / 8 / 50 mm
- Multi- writing upto 4 people
  - It can differentiate between the 4 collaborators so they all can write with different colors at the same time

InGlass Technology

Previously p-cap and IR-based touch was used. But problems associated with these are loss of clarity in large sized display, huge bezels, no support for multi-touch and susceptible to dust.

Samsung Flip uses InGlass technology developed by FlatFrog.

	IR	PCAP	InGlass™
Touch Performance	X	√	√
Bezel Free Design	X	√	√
Curve Support	X	√	√
>1000 Pressure	X	X	√
Ease of Integration	√	X	√
Glove	√	X	√
Scalable	√	X	√
HD/UHD Clarity	√	X	√

Figure 18: Exemplary screen size vs feature technology options (F-Series).

Source: https://www.flatfrog.com/whitepapers

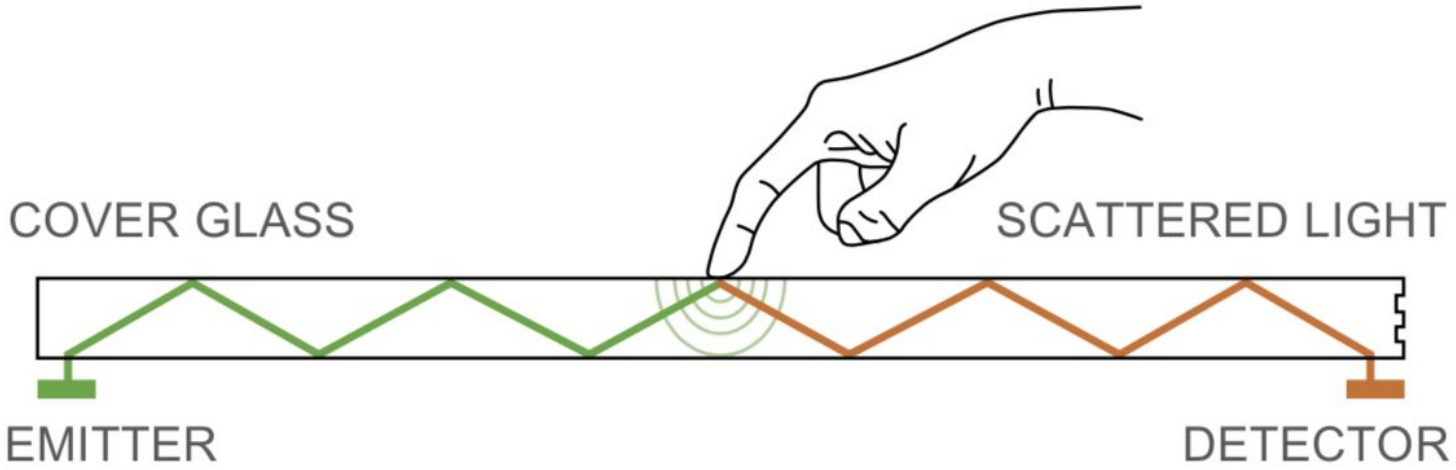
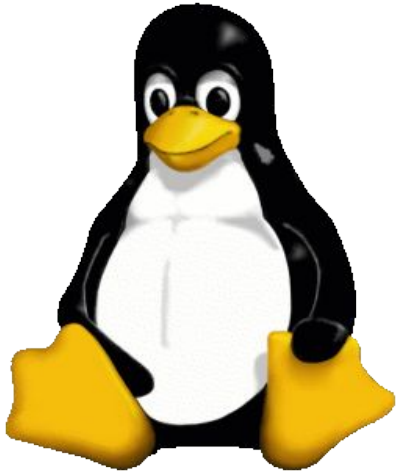


Figure 3: The basic FTIR shown for F-Series (How InGlass™ Touch works n.d.).

Source: https://www.flatfrog.com/whitepapers

Developer Specification

- Operating System : Tizen 3.0 (VDLinux) used by wide range of Samsung Devices.
- Programming language : Python



Obtaining logs

- Install pygame
- Install kivy module using pip install
- Run the following code.
- Connect the computer to HDMI and USB port.

```
from kivy.app import App
from kivy.uix.widget import Widget
from kivy.graphics import Line

class DrawInput(Widget):

    def on_touch_down(self, touch):
        print(touch)
        with self.canvas:
            touch.ud["line"] = Line(po

    def on_touch_move(self, touch):
        print(touch)
        touch.ud["line"].points += (to

    def on_touch_up(self, touch):
        print("RELEASED!",touch)


class SimpleKivy4(App):

    def build(self):
        return DrawInput()

if __name__ == "__main__":
    SimpleKivy4().run()
```

Source:
https://pythonprogramming.net/kivy-drawing-application-tutorial/

Logs obtained from the device using the Kivy Module on Python

	Time Stamp	x-coordinate	y-coordinate	
56	44937300	53	337	2
56	44970019	57	298	2
56	44986079	65	262	2
56	45000499	77	229	2
56	45010348	82	215	2
56	45020225	94	187	2

0- starting to write (pen down)
1- currently writing
2- pen up event

Problems/ Challenges

“ Houston, we have a problem! ”

Problem of Latency

Appreciable time delay between fast movement of passive pen and the response on the display

Problem of Unwanted erasing

Software isn’t able to differentiate between intended and unintended erasing.

Initial proposal to solve the problem

- We will try to make a logistic regression model that will be able to classify the palm touches as intended to erase or not.
- We thought of deep neural network but the computational time will introduce latency.
- The primary features that we will in the model will be :-
  - Distance from pen
  - Shape of patch
  - Direction of movement of palm
  - Speed of the palm with respect to the pen

Problem of Big Palm

Software detects the area of palm in contact larger that what is actually in contact