EECS 452 Project Progress Report

Zhonghan Li, Nithin Krishnan

Our project is on finding links between user inputs and architectural events on smartphones. To do this, we must first find methods to log user input in synchronization with hardware performance. Our platform of choice is mobile; although neither of us have experience with Android development, we hope the wide array of user gestures(swipe, tap, multi-touch, etc) will better supply us with rich input information.

We quickly came to realization that we needed rooted phones for overlaying user input tracking, and thus were supplied with Google Nexus 5 phones running Android Marshmallow on the Snapdragon 800. We are looking to test our devices mainly three applications to measure the performance of the CPU. The applications will be a simple offline game, writing into a text document and browsing through cached web pages. To isolate the tests further from extraneous and unpredictable processing, we first hope to run it in airplane mode, so no network activity or stray packets will impact the fidelity of the tests.

Progress:

With the help of the application Repetitouch we are able to measure certain parameters of user inputs. These are logged by the application and can be easily retrieved and play-backed. We pulled the logs from the phone and were able to write a small Python script to parse the relevant data(time, gesture, coordinates) into a csv file, which we hope to do some analysis on using pandas to see if there is a correlation between the input and hardware events.

The Performance measuring Units (PMUs), which are available in ARMv7 processors can be used to measure architectural events like cycle count, memory accesses etc. There are several resources online to enable a user to enable the use of PMUs present in the target device and measure architectural events. We require root access to access the PMU and measure the architectural events. We enabled root access with the help of SuperSU and are currently working on implementing it before we run our tests.

Planned Work:

We plan to use monkeyrunner to set up our test environment by opening the relevant applications together. After this, we will each run the test and have two initial sets to compare to. If there is no immediate distinction, we may find volunteers to participate in the tests, which should last no longer than 5 minutes.

Devise an efficient method of logging the information from the PMUs. The raw information from the PMUs are however difficult to operate on. So, we aim to measure the PMU for a certain interval of time after a significant user input to obtain more tangible information. Finally, once we have all our time stamped data, we will see if there is a correlation between input events and architectural events on both a micro(CPU statistics per action) and macro(total architectural events for each user to achieve the tasks) scale.