**Introduction:**

In the world of PBs and TBs data, if power is not consumed in an appropriate way, it’s harmful in all aspects like performance, energy loss, cost of storage and contributes highly towards environment pollution. It is always assumed that with increase in data, power consumption is more, but in actual it depends on what type of operation is performed on the data you consist of and what type of memory accessed to perform the same.

In a typical “pipeline” process, data move is from processor to processor where different operations are performed. The assumption is that there is an efficiency gain by allowing the processors (or cores) specialize – think automobile assembly line. However, there may be a flaw in this thinking. As each core is general purpose, the efficiency gains, if any, from specialization, may be small. Given a large enough data set, it may be more efficient to move the operations, instead of the data, from core to core. Our project hence here tests this idea using raspberry pi which helps to avoid the noise while the actual code is running on OS.

**Goal:**

This project intends to write the benchmark codes which states in what situation power consumed is high with respect to data movement. And Performance is calculated with respect to the number of instructions executed and operation performed with respect to time.

**Hypothesis:**

It is assumed that as the number of context switches increase, amount of power consumed also increases and similarly if the computation fits within cache, the power consumed should be very less compared to the consumption when the execution has the transfer of information from cache to main memory.

Hence, given a large enough data set, relative to the size of the local memory (local being relative itself), the power consumption will decrease if the operation is moved from between cores as compared to moving the data between cores.