Experiment Design

I. Final objective: what is the aim of the study and what literature has been consulted.

Generate a relation trend between the size of L1(d-cache)/L2 cache vs power consumption vs performance (Hit/Miss rates).
For determining ARM system properties, we are taking reference from the following website [1]:
System properties:

- The ARM processor incorporates data cache and instruction cache which are four way set associative, formed from synchronous SRAM and have similar architectures. The cache length line is 32 bytes.
- Each cache segment consists of data RAM for storing data or instructions and TAG RAM for storing cache line address.
- During cache access, all TAG RAMs are accessed for the first nonsequential access, and the TAG address compared with the access address. If a hit occurs, the information from the fragment is chosen to be returned. On the off chance of a miss, the external memory must be accessed. If access is a buffered write, at that point, the write buffer is utilized.
- If read access from a cachable memory locale misses, new information is stacked into one of the four fragments. This is a dispense on the read-miss replacement strategy. The choice of the portion is performed by a segment counter that can be checked in a pseudo-random way.

According to Stamenkovic et al., there is a sweet spot for cache size that leads to an optimal amount of power consumption. “The Power Index model implies the existence of a tradeoff relation between the cache miss rate and cache size. Namely, the power consumption has a sweet spot as cache size changes: for a too small cache, program thrashes burning power on external memory accesses; for a too large cache, cache itself burns too much power. As miss rate decreases with increase of the cache size, their product must have a local minimum for given application. In other words, there must be an optimal point between them.” [2]

II. The parameters of the CPU and/or memory models you plan change.

Keep the memory model fixed to “In-Order” but will vary the size of L1/L2 caches.

III. The range of each parameter, e.g., will change the L1 cache from128 K to 1,024 K in increments of 128 K.

Cache size settings:

- 256 b L1
- 1 kB L1
- 4 kB L1
- 16 kB L1

- 64 kB L1
- 64 kB L1; 64 kB L2
- 64 kB L1; 128 kB L2

IV. The number of runs for each experimental setup.

10 runs for each cache setting per dataset input.

V. Estimated run time per run.

Single run with Freqmine small size data set takes 47 minutes to complete.
Base config:

- CPU = ARM v8 (FS mode)
- Kernel = Linux
- CPU Type = HPI
- CPU Frequency = 4GHz
- Total Cores = 2
- Memory Type = DDR3_1600_8x8
- Number of Memory Channels = 1
- Memory Size = 2 GB
- Cache line size = 64 KB

VI. If you plan to construct confidence intervals, the total number of runs.

10 runs for each cache setting per dataset input.

VII. Discussion of research hypotheses, the expected results.

Hypothesis: Based on intuition from Stamenkovic et al., we expect there to be a sweet spot for cache size that leads to the lowest power consumption. We believe that cache size of 32 kB L1 will be too large and the caches will then generate too much power; thus we believe cache size 4 kB will have the lowest power consumption. In addition, we expect the performance will hit a peak around cache size 16 kB L1 and will not increase much more when increasing the L1 cache from 16 kB to 32 kB.

References:

[1] http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ddi0201d/I21752.html

[2] Stamenkovic, Zoran & Vater, F. & Dyka, Zoya. (2003). A Framework for Selection of Cache Configurations for Low Power. https://www.design-reuse.com/articles/7319/a-framework-for-selection-of-cache-configurations-for-low-power.html