Cache Design using CACTI

CACTI is an analytical tool used to model caches, memories, and off-chip I/O. It is widely used in research to study the trade-offs of PPA (Power, Performance, Area) for different cache configurations such as capacity, associativity, manufacturing technology node, etc.

In this assignment, you will use CACTI tool to observe the impact on design metrics (area, power, delay) for SRAM-based caches with different configurations.

This assignment is worth 10 points and the submission is due on 11:59 pm ET on Feb 10, 2025.

Your tasks

1) Please clone CACTI tool using the command below
   a. git clone https://github.com/HewlettPackard/cacti.git
2) Build CACTI
   a. make -j
   This will create an executable (cacti) and a directory of object files (obj_dbg/)
3) Run CACTI
   a. ./cacti -infile cache.cfg
   b. This will create cache.cfg.out (a csv file) with the modeled values. Example below:

```
1 Tech node (nm), Capacity (bytes), Number of banks, Associativity, Output width (bits), Access time (ns), Random cycle time (ns), Dynamic search energy (nJ), Dynamic read energy (nJ), Dynamic write energy (nJ), Standby leakage per ban
  k(mW), Area (mm2), Ndwl, Ndbl, Nspd, Ndcm, Ndsam_level_1, Ndsam_level_2, Data arrray area efficiency %, Ntwl, Ntbl, Ntspd, Ntcm, Ntsam_level_1, Ntsam_level_2, Tag arrray area efficiency %,
2 90, 131072, 1, 2, 512, 1.47098, 1.86851, N/A, 0.303592, 0.615022, 63.7023, 2.24949, 2, 2, 1, 2, 2, 4, 1, 8, 1, 77.9289,
3 90, 131072, 1, 2, 512, 1.47098, 1.86851, N/A, 0.303592, 0.615022, 63.7023, 2.24949, 2, 2, 1, 2, 2, 4, 1, 8, 1, 77.9289,
4 90, 131072, 1, 2, 512, 1.47098, 1.86851, N/A, 0.303592, 0.615022, 63.7023, 2.24949, 2, 2, 1, 2, 2, 4, 1, 8, 1, 77.9289,
```

Each new line corresponds to a ./cacti run.

You can create different .cfg files for the cache/memory configuration you want to model. **However, for this assignment, you will use *cache.cfg* (in the CACTI github repository) as the default configuration and modify it for the following cases. Outputs from all the runs will be appended to *cache.cfg.out* file which can then be analyzed.**

1) [2 points] sweep associativity: 2, 4, 8, 16
   a. Report the Access time (ns), Data area efficiency (%), and Area (mm2) in a table.
   b. Observe the trend for each of the above metrics with increasing associativity and mention your reasoning for the same.
   Note: all other parameters including cache capacity remain default.

2) [2 points] sweep manufacturing technology node (-technology): 22 nm, 32nm, 40 nm, 90 nm
   a. Report the Access time (ns), Area (mm2), Energy (nJ) in a table.

b. Observe the trend for each of the above metrics with increasing node and mention your reasoning for the same. Note: all other parameters remain default.

3) [1 point] sweep cache <u>size</u> (Bytes): 32768, 131072, 1048576, 2097152
   a. Report the Access time (ns), Area (mm2), Energy (nJ) in a table.
   b. Observe the trend for each of the above metrics with increasing cache capacity and mention your reasoning for the same. Note: all other parameters remain default.

4) [2 points] sweep the number of <u>read-write port</u>: 1, 2, 4
   a. Report the Access time (ns) in a table. [Note: all other parameters remain default.]
   b. Observe the trend of access latency with the increasing number of ports and mention your reasoning for the same.
   c. Why do you think a cache with more read-write ports be chosen for a system even though the access latency increases with the number of ports?

5) [1 point] modify *cache.cfg* with the parameters that have the least "Area" from each of the above four cases (#1, #2, #3, #4) and run CACTI to
   a. Report the associativity, technology node, cache size, and the number of read-write port along with the access latency, area, energy, data array efficiency, and tag array efficiency.

6) [2 points] What configuration (associativity, cache size, cache block size, etc.) would you choose for the following:
   a. L1
   b. LLC
   Explain your choices and mention if you need any additional information to decide.