Journal of Theoretical and Applied Information Technology
31st March 2015. Vol.73 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A FULLY ASSOCIATIVE CACHE ARCHITECTURE

S.SUBHA
SITE, Vellore Institute of Technology Vellore, India
E-mail: ssubha@rocketmail.com

ABSTRACT

Fully associative caches expend maximum energy among caches for any size. The blocks of fully associative cache are searched in parallel during address mapping. This paper proposes architecture to reduce the power consumption in fully associative cache. The cache blocks are assumed to operate in one of two modes - on or off. The model consists of introducing one bit per block which is enabled on block occupancy. This bit is checked before the address match is performed. This saves power consumption in address comparison. The proposed model is simulated with SPEC2K benchmark. An improvement in power consumption of 59% with no change in average memory access time is observed for the chosen parameters.

Keywords: *Average Memory Access Time, Cache Architecture, Cache Performance, Fully Associative Cache, Power Savings*

1. INTRODUCTION

A cache is denoted by (C,k,L) where C is the capacity, k the associativity and L the line size. Caches are of three kinds – direct mapped, set associative and fully associative [1,2]. A line is placed in fixed location in direct mapped cache. It can occupy any of k ways in k-way set associative cache. It can be place in any of n blocks in fully associative cache of n blocks. Fully associative caches have all blocks enabled during program execution. An address is matched with all the cache blocks. On match, the block is accessed. On cache miss, the least recently used block is used to place the address line. This architecture enabling all cache blocks consumes more power than enabling subset of cache blocks. Algorithms to reduce the energy consumption during address mapping achieved by restricting the address comparison to subset of cache blocks as a function of the address is proposed in [3, 4]. Algorithm to map incoming line to specific cache block is proposed in literature [6]. This method performs address translation by performing mathematical transformations on the address line. An address maps to specified block in this case. Algorithm to achieve improved average memory access time (AMAT) with energy saving is proposed in literature [7]. This method employs mathematical transformations like shifting on the address. An algorithm to separate the address part of the fully associative cache is proposed in [5].
This paper proposes architecture to reduce power consumption in fully associative cache. The proposed model is independent of address to be probed. The blocks are assumed to be in on or off position. On occupancy, the block is placed in on mode. The architecture consists of introducing one bit vector for the cache. This vector is called *enable vector*. All entries in enable vector are set to zero initially. On block occupancy the corresponding entry is set to one. Address mapping with the block address is performed only if the enable bit for the block is set to one. This reduces the number of block comparisons on an average. The traditional placement/replacement algorithm is used for address mapping. The time for address mapping is reduced unlike the models proposed in literature. The proposed model is simulated with SPEC2K benchmarks. An average improvement in power consumption of 59% with no change in AMAT is observed for the chosen parameters.
The rest of the paper is organized as follows. Section 2 gives the motivation, section 3 describes the proposed model, section 4 gives the mathematical analysis of proposed model, section 5 gives the simulation section 6 gives the conclusion followed by references.

2. MOTIVATION

Consider two level inclusive cache model. Consider fully associative level one cache with eight blocks. Let level two is 2-way set associative cache of eight sets. Consider address trace 100,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

200, 300, 400. Let line size be one byte. In traditional fully associative cache, the addresses are placed in blocks 0,1,2,3. All eight blocks are enabled during address mapping. The match logic of fully associative cache enables all the cache blocks. Assume it takes 10W power per cache block during operation. The total energy consumed in level one cache is 8*10*4W = 320W. Next consider the following algorithm. The cache operates in two modes – on and off. The cache is in off mode initially. A one bit vector indicating the power status of the block is present. Initially it is set to zero for all entries indicating that the blocks are unoccupied. On block occupancy the corresponding bit is set to one. According to this logic, the blocks 0-3 are enabled. During address mapping, for mapping 100, no match logic is implemented. For placing address 200, the block zero is accessed, for address 300 blocks zero and one are accessed, for address 400 blocks zero, one , two are accessed. The total power consumed is (0+3*10+2*10+10) W=60W. The first term is the energy consumed before placing block with address 100. Block with address 100 is enabled for next three addresses in the trace. This is the second term. Block with address 200 is active for two address accesses after placement. This is the third term. Block with address 300 is active for placing the last address. This is the fourth term. Assume the additional circuitry consumes 20J of energy. The total energy consumed is 80J. The level two cache in both cases is assumed to consume the same amount of power. For the proposed algorithm an improvement in power consumption of 75% is observed. This is the motivation of this paper. There are four misses in the proposed model and traditional model in level one and level two caches. There is no change in average memory access time.

3. PROPOSED MODEL

Consider two level inclusive cache model. Let level one be fully associative cache of N blocks. Let level two is w-way set associative cache of S sets. The fully associative cache is assumed to be in either on or off mode. The fully associative cache is in off mode initially. A one bit vector indicating the power status of the block is present. It is initially set to zero indicating that the blocks are in off mode. On block occupancy the bit is set to one. An address a is mapped to level one as follows. The enabled blocks are matched with the address a. If found, it is cache hit. The line is accessed. If not found, it is miss. The least recently used cache block is chosen to place the line. If this block is not enabled, the block is enabled for placing the line. The enabled blocks are used for performing match logic. A block is enabled if it is already enabled or a line is placed in it. Else, the block is in off mode. The proposed model for fully associative cache is shown in figure 1

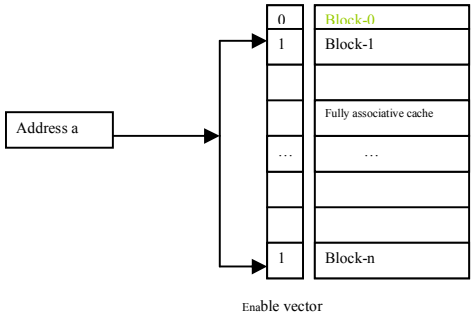

Figure 1 Architecture of Proposed System. An Enable vector indicates block occupancy. Address a is matched with enable vector entries that are equal to one.

4. MATHEMATICAL ANALYSIS OF PROPOSED MODEL

Consider the two level inclusive cache system mentioned in Section 3. Let N be number of cache blocks in level one. Denote this system as C_{prop}. Let the address trace be R references. Let $h_1, h_2, t_1, t_2, t_{12}, m$ be level one cache hits, level two cache hits, level one access time, level two access time, transfer time between level one and level two and miss penalty in the proposed system. The AMAT of the system is given by

$$AMAT(C_{prop}) = \frac{1}{R}\begin{pmatrix} h_1 t_1 + h_2(t_1 + t_2 + t_{12}) + \\ (R - h_1 - h_2)m \end{pmatrix} \quad (1)$$

Consider the traditional system i.e. with traditional fully associative cache in level one and set associative cache in level two. Denote this system by C_{trad}. Let $H_1, H_2, T_1, T_2, T_{12}, M$ be level one cache hits, level two cache hits, level one access time, level two access time, transfer time between level one and level two and miss penalty in the traditional cache system. The AMAT of this system is given by

Journal of Theoretical and Applied Information Technology
31st March 2015. Vol.73 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$AMAT(C_{trad}) = \frac{1}{R}\begin{pmatrix} H_1T_1 + H_2(T_1 + T_2 + T_{12}) + \\ (R - H_1 - H_2)M \end{pmatrix} \quad (2)$$

An improvement in AMAT is observed if

$$\frac{1}{R}(h_1t_1 + h_2(t_1 + t_2 + t_{12}) + (R - h_1 - h_2)m) <= \frac{1}{R}\begin{pmatrix} H_1T_1 + H_2(T_1 + T_2 + T_{12}) + \\ (R - H_1 - H_2)M \end{pmatrix} \quad (3)$$

Consider the energy consumption. In the traditional cache, the whole cache is in high energy mode during operation. Let E_{high} be the energy consumed per cache block in level one cache during operation. Let E_2 be the energy consumed in level two cache during cache operation. The total energy consumed by traditional cache model during operation is given by

$$E(C_{trad}) = RNE_{high} + E_2 \quad (4)$$

In the proposed system, let the total time of cache operation is T and energy consumed by additional circuitry be E_{seq}. Let n_1 blocks be enabled from time t_1, n_2 blocks be enabled from time t_2, and so on n_q blocks are enabled at time t_q assuming maximum of n_q blocks is enabled during cache operation. The total energy consumed in this system is given by

$$E(C_{prop}) = \begin{pmatrix} n_1(T - t_1) + (n_2 - n_1)(T - t_2) \\ + \ldots + (n_q - n_{q-1})(T - t_q) \end{pmatrix} E_{high} + E_2 + E_{seq} \quad (5)$$

An improvement in energy consumption is observed if

$$RNE_{high} + E_2 >= \begin{pmatrix} n_1(T - t_1) + (n_2 - n_1)(T - t_2) + \ldots \\ + (n_q - n_{q-1})(T - t_q) \end{pmatrix} E_{high} + E_2 + E_{seq} \quad (6)$$

5. SIMULATION

The proposed model is simulated with parameters shown in Table 1. Routines in C language were written to simulate the traditional and proposed models. The values of hits and misses were collected on simulating with SPEC2K benchmarks and Simplescalar Toolkit. The proposed model was validated with Verilog code run on Quartus II. The values for power consumption were collected from this run. An address of 32bits is assumed with block size of 32 bytes.

Table 1 Simulation Parameters

Parameter	Value
L1 Size	32768 blocks
Line Size	32 bytes
L2 Size	4096 sets
L2 associativity	16
L1 access time	6 cycles
L2 access time	20 cycles
L1 to L2 transfer time	12 cycles
Miss penalty	65 cycles
Power for 27bits address in traditional model	456.9075mW
Power for 27bits address + enable bit in proposed model	457.0575mW

The average memory access time (AMAT) is calculated based on the model derived in section 4 of this paper. The AMAT is shown in figure 2. As seen from the figure 2 there is no change in AMAT.

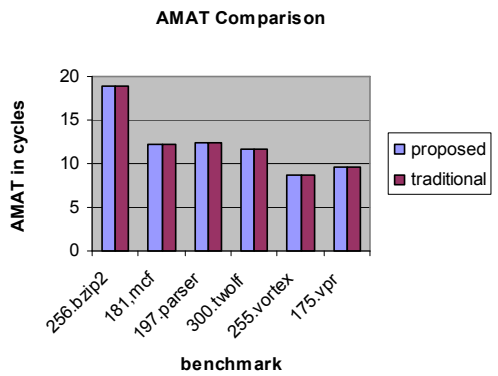

Figure 2 AMAT Comparison

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The power consumption is shown in figure 3. As seen from the figure 3 there is 59% improvement in power consumption. As observed from figure 3 if all the cache blocks are enabled, there is increase in power consumption due to additional hardware. This is seen in the case of 256.bzip2 and 197.parser.

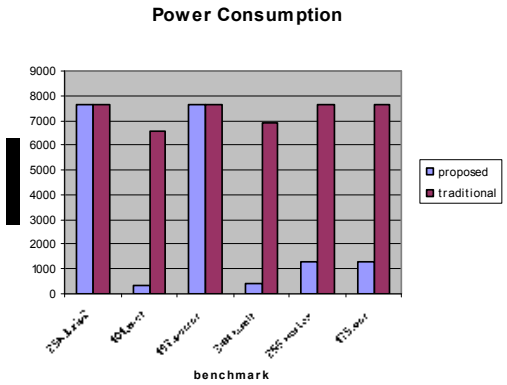

Figure 3 Power Consumption

6. CONCLUSION

A fully associative cache with power saving and no change in AMAT is proposed in this paper. Each cache block in fully associative cache is assumed to be either in on or off position based on occupancy. The model introduces one bit vector of size of number of cache blocks. This vector indicates the occupancy of the block. The address match logic is performed after inspecting this vector bit for each block. The occupied blocks are only used for address mapping. The proposed model is simulated with SPEC2K benchmarks. Power saving of 59% with no change in AMAT is observed for the chosen parameters.

ACKNOWLEDGMENT

The author expresses thanks to Santa Clara University, Santa Clara, CA, USA for providing Simplescalar Toolkit and SPEC2k benchmarks.

REFERENCES

[1] Alan Jay Smith, "Cache Memories", Computing Surveys, Vol. 14, No. 3, September 1982, PP.473-530

[2] David A Patterson, John L Hennessey, Computer Architecture: A Quantitative Approach, 3rd Edition, Morgan Kaufmann Inc. 2003

[3] S.Subha, "An Algorithm for Fully Associative Cache Memory", Proceedings of Second Bihar Science Conference, 2009

[4] S.Subha, "Performance Analysis of Variable Number of Sets in Fully Associative Cache", Proceedings of International Conference on Computing, Engineering and Information, 2009, pp. 295-298

[5] S.Subha, "An Energy Saving Model for Fully Associative Cache", Proceedings of TISC, 2011, pp.141-143

[6] S.Subha, "Energy Efficient Fully Associative Cache Model", International Journal of Computer Applications, Vol. 47, No. 6, June 2012, pp. 16-18

[7] S.Subha, "A Fully Associative Cache Algorithm with Improved Performance", JATIT, Vol. 60, No. 3, 2014