Influence of Caching and Encoding on Power Dissipation of System–Level Buses for Embedded Systems

William Fornaciari (1), Donatella Sciuto (1), Cristina Silvano (2)
(1) Politecnico di Milano, Dip. di Elettronica e Informazione, P.zza L. Da Vinci 32, 20133 Milano, Italy.
(2) CEFRIEL, via Fucini 2, 20133 Milano (MI), Italy.

Abstract

This paper proposes a methodology to evaluate the effects of encodings on the power consumption of system-level buses in the presence of multi-level cache memories. The proposed model can consider any cache configuration in terms of size, associativity and block. It includes also the most widely adopted power oriented encoding techniques for data and address buses. Experimental results show how the proposed model can be effectively adopted to configure the memory hierarchy and the system bus architecture from the power point of view.

1. System-level power model

The proposed model is composed of three main sub-models: the memory hierarchy, the bus encoder and the address and data stream generator, which have been integrated in an object-oriented sw tool written in *C++*. The models can be used as basic blocks of different types of system architecture, ranging from dedicated system to general-purpose computer systems.

The *memory hierarchy model* consists of a multi-level storage hierarchy of on-processor and off-processor caches. The generic level of the hierarchy can be organized as single *unified* cache or *split* between two different caches for instructions and data. The cache model considers several configurations in terms of cache size, block size, degree of associativity, write strategy and replacement policy. More in detail, the model offers the capability to vary: the size of the block, the cache size and the degree of associativity. The write strategy can be *write-through* or *write-back*. In the case of a write miss both the options *write-allocate* and *no-write-allocate* can be used. For set or fully associative caches, the block replacement policy can be *random* or *LRU*.

To evaluate the bus encoding effects on power consumption, the *bus encoder model* can be inserted either on the interface from the processor to the first level of the memory hierarchy or between any adjacent levels of the memory hierarchy. The model implements the most common power-oriented bus encoding techniques, such as Gray, Bus-Invert, *T0*, *T0_BI*, *Dual_T0* and *Dual_T0_BI*. The encoding schemes can be applied to both the data and address buses.

The *address and data stream generator* aims at analyzing the system-level bus behavior by using address and data streams derived either by tracing a real microprocessor or by using a stream generator to simulate the execution of a generic program on a microprocessor. More specifically, the address generator models the processor-to-memory communication taking into account the spatial and temporal locality of memory references. The current version of the generator includes a generic load/store *RISC* architecture, although to derive the experimental results we refer to the instruction set of an existing processor, the *32*-bit *ARM7TDMI*. In our model, we assume that the memory address spaces for data and instructions are separated. The address sequence in memory is generated by assigning the percentage of instructions of different classes, considering that we can specify: the format and the execution frequency for each instruction class; the addressing modes for each instruction and the related execution frequency and the execution rate of a conditional branch.

2. The simulation methodology

In this section, we describe the simulation methodology used to profile the power consumption of an embedded system consisting of the *32*-bit low-power processor *ARM7TDMI* and a multi-level memory hierarchy providing a *32*-bit address bus and a *32*-bit data bus. The reference architecture is composed of the *33 MHz ARM7TDMI* processor and its main memory interfacing through *66 MHz* and *60 pF* buses. Starting from this reference architecture, *four* different system configurations have been analyzed to evaluate:

- the bus encoding effects without cache *(CASE1)*;
- the cache effects without bus encoding *(CASE2)*;
- the combined effects of on-processor bus encoder and off-processor cache *(CASE3)*;
- the combined effects of off-processor cache followed by off-processor bus encoder *(CASE4)*;

In the simulation, we used a *100 000* generated instructions stream and a memory hierarchy constituted by a first level off-processor cache adopting write through, no write allocate and random block substitution policies. Results concerning *CASE1* has been presented in [2]. *CASE2* aims at studying the effects of the off-chip first level cache, whose parameters vary from *4KB* to *32KB* for the cache size, *32*-bit to *256*-bit for the block size and associativity of *1-2-4-8* ways. We analyzed the miss rate vs cache size for four different block sizes and degrees of associativity. As expected, the miss rate decreases when these three parameters increase. Power behaves similarly because, by increasing the number of memory requests directly satisfied by the cache, the number of references to the main memory decreases. Consequently, a considerable reduction of the traffic occurs on the cache-to-memory bus, which has to switch larger capacitance *(60 pF)* than the processor-to-cache bus *(10 pF)*. The corresponding power figures show a similar trend. Due to space limitation, only

the diagrams of 2-ways set-associative caches are reported in the following (fig. 1 and fig. 2).

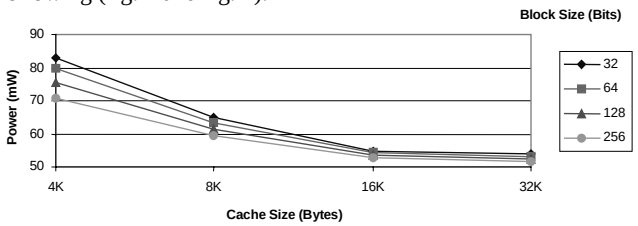

Figure 1. Power for address bus vs cache size for a 2-ways set associative cache and four different block sizes.

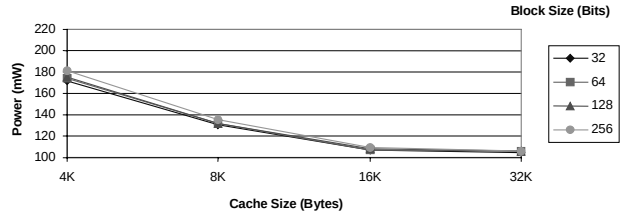

Figure 2. Power for data bus vs cache size for a 2-ways set associative cache and four different block sizes.

A power reduction occurs for both address and data buses for larger cache sizes. A power reduction for larger block sizes can be observed only for the address bus, whilst for the data bus the power is almost invariant for any block size. As a matter of fact, for larger block sizes, the number of consecutive addresses loaded in caches increases and thus the average number of transitions (i.e., the power) of the address bus decreases for larger block sizes. The data bus behavior is quite different, since the data value of consecutive memory locations are distributed randomly. Hence, the power is almost the same for larger block sizes. A comparison with the bus power dissipated by the reference architecture (*197.64 mW* and *330.53 mW* for address and data bus respectively) has shown how the memory hierarchy implies performance advantages but also power savings. The reduction increases for larger cache sizes. These results do not consider the internal power dissipation of the cache array, thus the effective reduction could be traded-off by the cache internal power.

For *CASE3*, the bus encoder is implemented on-processor, whilst an off-processor *L1* cache is provided, whose cache size varies from *4KB* to *32KB*, the degree of associativity is *1-2-4-8* ways and the block size is *64*-bit. The power versus the cache size for 2-ways set associative cache and several bus encodings is reported in fig. 3 and fig. 4 for address and data bus, respectively.

Concerning the address bus, the power dissipation is considerably reduced by adopting the Gray, *Dual_T0* and *Dual_T0_BI* schemes. The average percentage of power saved by several encodings with respect to the reference architecture are reported in table 1 for four cache sizes. For each encoding technique, larger savings can be obtained for larger caches, since the number of accesses to the main memory decreases.

Finally, table 2 reports the average percentage of power saved by the adopted encodings with respect to the binary encoding (same as *CASE2*) for four cache sizes. For the data bus, the power saving with respect to the binary code is very limited (approximately the 4% for *BI*), while the power is almost invariant for the other encodings. These results eliminate practical applications of the studied encoding methods on the data bus.

The analysis of *CASE4* is similar to those carried out for the *CASE3*. Details and diagrams for all the presented cases can be found in [4].

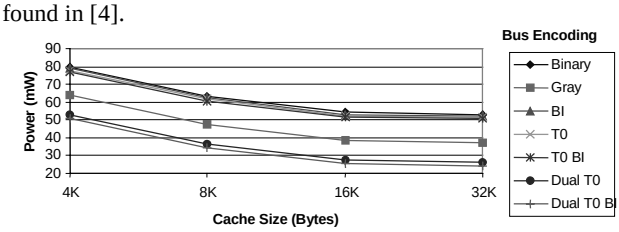

Figure 3. Power for address bus vs cache size for 2-ways set associative cache and several bus encodings.

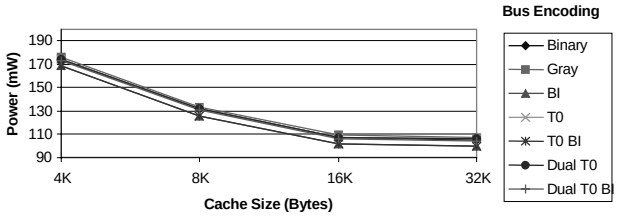

Figure 4. Power for data bus vs cache size for 2-ways set associative cache and several bus encodings.

% Saved (Average) vs Reference							
Cache Size	**Binary**	**Gray**	**BI**	**T0**	**T0 BI**	**Dual T0**	**Dual T0 BI**
4KBytes	58.78	66.74	59.28	59.70	60.12	72.42	73.41
8KBytes	67.46	75.42	67.96	68.38	68.80	81.10	82.09
16KBytes	72.17	80.13	72.67	73.08	73.50	85.80	86.80
32KBytes	72.76	80.72	73.26	73.68	74.10	86.40	87.39

Table 1. Average power saving on address bus for different bus encodings vs. the reference architecture for 4 cache sizes.

% Saved (Average) vs Binary Encoding						
Cache Size	**Gray**	**BI**	**T0**	**T0 BI**	**Dual T0**	**Dual T0 BI**
4KBytes	19.32	1.22	2.22	3.25	33.09	35.50
8KBytes	24.48	1.54	2.82	4.11	41.92	44.98
16KBytes	28.65	1.80	3.30	4.81	49.06	52.63
32KBytes	29.25	1.84	3.36	4.91	50.08	53.74

Table 2. Average power saving on address bus for different bus encodings vs. the binary encoding for 4 cache sizes.

3. Conclusions and future work

Aim of this work has been to evaluate the effects on power of bus encoding schemes in the presence of multi-level cache memories. Current effort is devoted to analyze high-end general purpose systems targeted for PowerPC architecture, where the presence of Virtual Memory as well as a finer grain model of the memory arrays contribution have to be taken into account.

4. References

[1] M. R. Stan and W.P. Burleson, "Low-Power Encodings for Global Communication in CMOS VLSI", IEEE Trans. on Very Large Scale Integration (VLSI) Systems, Vol. 5, No. 4, December 1997, pp. 444- 455.

[2] L. Benini, G. De Micheli, E. Macii, D. Sciuto, C. Silvano, "Address Bus Encoding Techniques for System-Level Power Optimization", DATE'98, IEEE Design Automation and Test in Europe, Paris, 1998.

[3] C.L. Su and A.M. Despain, "Cache Design Trade-offs for Power and Performance Optimization: A Case Study" ISLPED, Int. Symposium on Low Power, 1995.

[4] Cristina Silvano, "Power Esitimation and Optimization Methodologies for digital circuits and Systems", PhD Thesis, University of Brescia, DEA, Italy, 1998.