

Reducing Translation Lookaside Buffer Active Power

Lawrence T. Clark
Intel Corp.

5000 W. Chandler Blvd.
Chandler, AZ 85226 USA

lawrence.clark@intel.com

Byungwoo Choi
Intel Corp.

5000 W. Chandler Blvd.
Chandler, AZ 85226 USA

byungwoo.choi@intel.com

Michael Wilkerson
Intel Corp.

5000 W. Chandler Blvd.
Chandler, AZ 85226 USA

michael.w.wilkerson@intel.com

ABSTRACT

Lowering active power dissipation is increasingly important for battery powered embedded microprocessors. Here, power reduction techniques applicable to fully associative translation lookaside buffers, as well as other associative structures and dynamic register files, are described. Powermill simulations of implementation in a microprocessor on 0.18 μ m process technology demonstrate 42% power savings. Circuit implementations, as well as architectural simulations demonstrating applicability to typical instruction mixes are shown.

Categories and General Descriptors

C.5.3 Microcomputers---Microprocessors

General Terms

Performance, Design.

Keywords

Low power, memory management units, dynamic circuits, translation lookaside buffers, register files.

1. INTRODUCTION

Memory management unit (MMU) power consumption comprises a significant portion of microprocessor cache memory power. Most of this dissipation is in the translation lookaside buffer (TLB) that provides physical address translation and page access permissions. These circuits are generally a critical timing path, so power savings must not be obtained at the expense of increased circuit delay. Virtually addressed caches do not require the physical address for cache comparison, but permissions must still be determined. This can allow late permission determination, e.g., the scheme used in the StrongARMTM [1] and XScaleTM microprocessors [2]. The most common TLB circuit implementation is the fully associative cache due to its high hit rates, ability to handle multiple page sizes, and high speed. Since the content addressable memory (CAM) and register file (RF) data portions are dynamic circuits, they are high power consumers. MMU power can also be avoided by including extra permission

bits in the data caches [3] at a performance penalty for each line's initial access. Micro-TLB's have also been proposed [4][5] to limit power by limiting the number of entries interrogated per access, at the cost of a lower hit rate and the added complexity of a secondary TLB.

TLB accesses exhibit temporal and spatial locality, key to all cache operations. Sequential memory accesses are often to the same page, particularly instruction fetches. The circuits described in this paper principally rely on this locality to limit power in the TLB by limiting overall access, or, when that is not possible, the register access and output bus delivery power. The basic scheme is to determine if the page accessed as a lookup in cycle N is the same page accessed in lookup cycle N-1, and to gate off subsequent actions if the page did not change. Intervening clock cycles that do not access the TLB are unimportant; hence, in this paper a sequential cycle will imply a sequential TLB lookup cycle. The amount of circuitry that can be gated depends upon how early a sequential access can be determined.

In section 2, analysis of benchmark programs demonstrates the expected locality and portends the expected power savings. Sections 3 and 4 introduce the basic scheme for limiting the RF and CAM power, respectively, as well as implementation in 0.18 μ m and 90nm microprocessor designs. Results and conclusions comprise the final sections.

2. REFERENCE LOCALITY

To determine the efficacy of the scheme, six benchmark programs were run on the architectural level simulator and their page accesses were logged to quantify sequential access behavior. Data accesses were used since it will be shown below that instruction accesses are easier to limit. The benchmark programs were Dhystone, Freeampi (an integer version of the Freeamp MP-3 application), IPtest2 and retix_atm (both networking IP benchmarks), and DCT, a discrete cosine transform application. For each program the number of sequential accesses, as defined by the access at cycle N being to the same page as the access at cycle N-1, was determined. The results are presented in Table 1.

Fig. 1 shows the cumulative percentage of sequential accesses, relative to the total number of accesses for access counts of lengths 1 through 10, between 11 and 20, and greater than 21 for three of the benchmarks. Each line represents the access pattern for a different page. For each program, most of the accesses are to a few of the pages, typically with long sequences. RetixATM is anomalous in that it has one page where single accesses comprise 33% of the total accesses. Nevertheless, the majority of accesses in the program are in sequences of two or more. For the DCT program, while not discernable from the figure, there are 22 pages, each comprising 1.54% of accesses in groups of two, that

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ISLPED '03, August 25–27, 2003, Seoul, Korea.

Copyright 2003 ACM 1-58113-682-X/03/0008...\$5.00.

make up over 1/3 of all accesses. One page has over 60% of all accesses, primarily in groups of 43. Thus, power can be conserved by detecting these cases and limiting the power dissipation of consecutive accesses to the same page.

Table 1: Sequential accesses by benchmark.

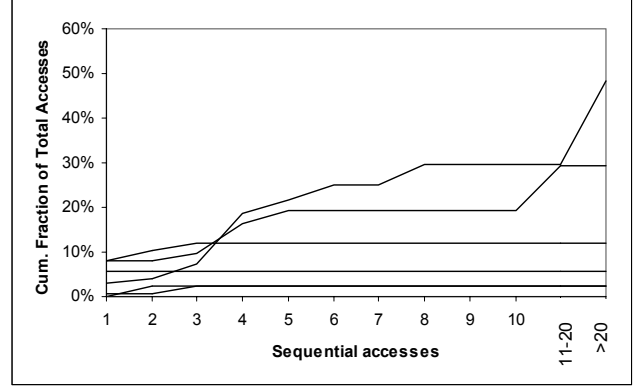
Benchmark	Active pages	Sequential accesses	Non-sequential accesses
Dhrystone	17	65%	35%
Freeampi	51	83%	17%
IPtest2	18	75%	25%
Retix_atm	15	57%	43%
DCT	131	70%	30%

3. LIMITING RF AND OUTPUT POWER

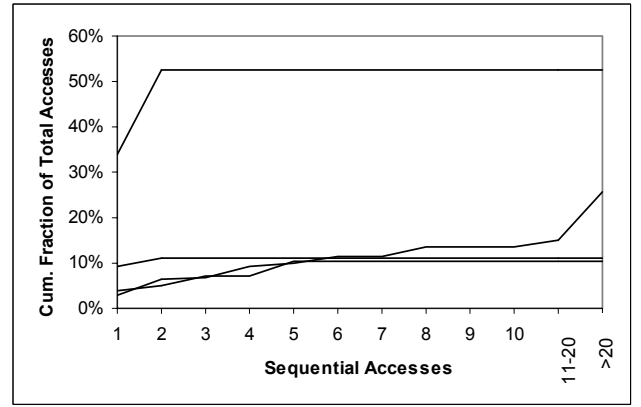
If the address to the MMU is available early in the pipeline, the virtual page address can be compared to that previously accessed. This in turn allows clock gating of the TLB access (see Fig. 2) to save power. Ideally, this gating is as high in the clock tree as possible, a goal that can be difficult to achieve due to the earlier setup times required. A dynamic compare [6] greatly accelerates the high fanin “or” gate. Where the address generation for loads and stores, i.e., the data memory path, is a critical timing path, clock gating is not feasible. This is the case in a typical RISC pipeline. Pages with a high percentage of single accesses tend to have a low total number of accesses and hence minimally affect the overall program energy. The small number of active pages is also shown in the table. Since the methods described here save power when sequential accesses are to the same page, the use of fewer, larger pages is suggested.

Lacking time to pre-compare the page addresses, the CAM portion of the TLB operation is unavoidable. However, power can still be saved by determining sequential accesses to the same page after the CAM and gating operation of the RF holding the physical addresses and permissions as shown in Fig. 3. No RF operation is required, since in this case, the next read will output the same physical page and permissions. The approach is to flop each match line and logically “and” sequential match line states to gate RF access. Gating cannot be done at the word-line (WL) enable clock driver as this would require logically ‘ORing’ the line by line match data which simply cannot be done in time. Instead, the circuit deselects the WL chosen by signal *match* when the previous access was to the same entry. The flops must all be reset to indicate no prior match when the processor is reset or when writing a new entry into the TLB. This reset, as well as the write circuits, are not shown in the figure.

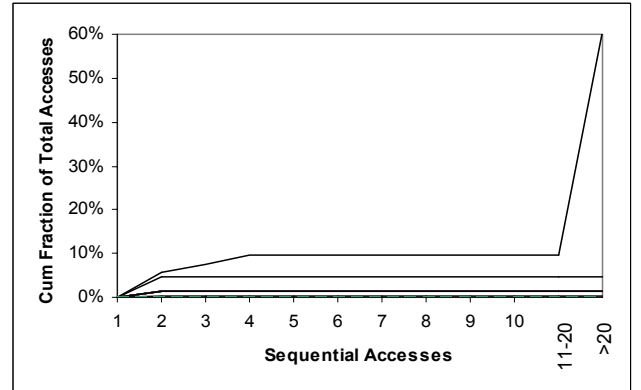
Referring to Fig. 3, if the CAM generates a match for a particular entry, the match signal is compared to the result of the previous CAM operation for each entry. This is accomplished by the RF WL driver ‘and’ gate, by including the *matN1#* signal in the select. Signal *matN1#* is the match line state in the previous cycle. The read WL is asserted only if the previous CAM page was different, i.e., a miss to this entry.



(a)



(b)



(c)

Fig. 1. Cumulative accesses/page for (a) Dhrystone, (b) RetixATM, and (c) DCT.

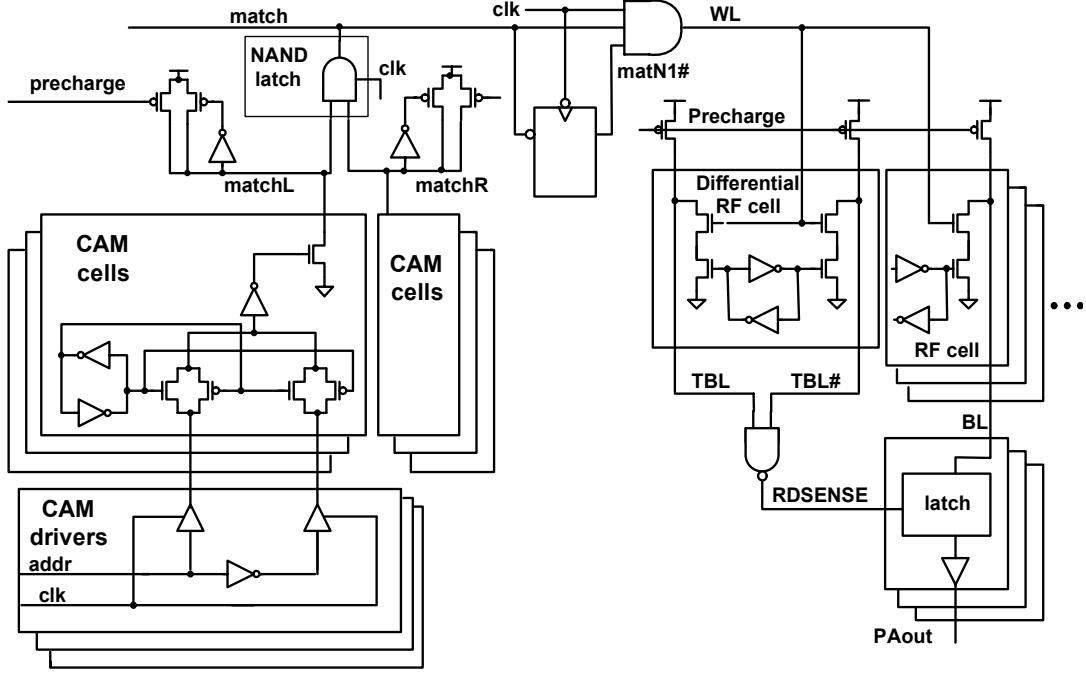


Figure 3. TLB with gated register file word line and output glitch suppression.

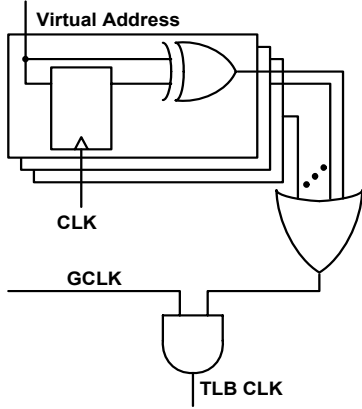


Fig. 2. Clock gating by detecting sequential page accesses.

Register files are usually implemented as domino circuits where the word-line acts as a D1 stage and the cell output as D2. The higher dynamic circuit activity factor propagates downstream, raising the activity factor of the output bus and all driven circuitry. Conventionally, the latch driving the physical address (PA) out opens on the same clock edge that activates the domino RF read. The latch opening propagates the precharge and later, the domino discharge (in the case where the output is a logic “0”). Limiting RF accesses can lower the activity factor of this high capacitance *PAout* output bus. The circuit to accomplish this is also shown in Fig. 3, and works in two ways. First, opening the latch dynamic to static converter [9] cannot occur if the RF will not fire, as it would propagate the precharge state with no subsequent data read to replace it. This eliminates activity on the

output bus as well as in the RF on a sequential hit to the same page. Secondly, it lowers the downstream activity factor to that of a static circuit. Here, by coinciding the latch opening with the typical RF read timing, the precharge signal is nominally unpropagated, even when a read occurs. The output precharge to signal glitch is suppressed. Losing the apparent race between the latch open and the output signal is not a functional issue. It (minimally) affects only power or performance when the silicon is off the nominal PVT corner by allowing a small output glitch. This technique is applicable to all dynamic register file designs.

Specifically, the word-line detection uses a differential readout cell and detecting a falling bit-line as shown in the figure. In our application, this does not require additional circuitry, since the differential readout cell was required for functionality. In the case the *WL* signal is not asserted, neither are signals *TBL* or *TBL#*. Hence, *RDSENSE* stays low. When the *WL* is asserted, *RDSENSE* opens the latch at the appropriate time, as described.

4. LIMITING CAM POWER

The primary CAM power consumers are the match line discharge and driving the differential CAM cell inputs. The CAM cell compares incoming data with the stored value, discharging a domino NOR pull-down *match* signal for a miss-matching address bit. Power is saved and performance gained by segmenting the *match* nodes into *matchL* and *matchR* and combining them via the static gate required to terminate any domino node as shown. The ‘and’ function is part of the dynamic to static converting latch.

The valid bit in each entry of the CAM is usually compared to a known value that forces a miss-match on invalid entries. The circuit shown in Fig. 4 can reduce the *match* line power by de-asserting the precharge signal for invalid entries. The *match* line is

statically held low, avoiding both precharge and discharge power when *VALID* is not asserted. This technique is also applicable to other CAM based circuits, e.g., write and fill buffers, which are often empty.

The circuit includes a self-terminating precharge signal that eliminates the power race between the precharge de-assertion and the dynamic pull-down circuit. This is also shown in Fig. 4. It also limits the clock power by detecting when the domino has not discharged on the last cycle and not asserting the precharge signal in that case. By gating the precharge early in the path, clock load transitions are also spared. CAM writes may occur on the low phase of the clock, while compare operations take place in the high phase. This requires that the *match* lines be properly precharged when writing a value to a previously invalid entry. This is done by asserting *precharge#* during a write, via the write word-line *WWL* signal.

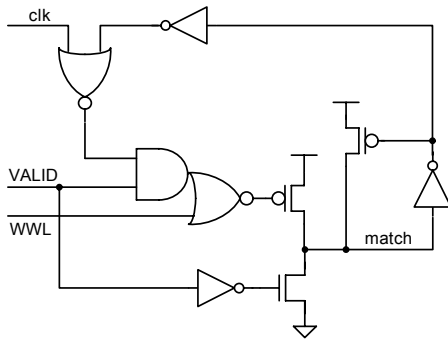


Fig. 4. Self-terminating precharge and valid bit gating of MATCH node.

A goal of the 90nm implementation was to increase the circuit speed considerably beyond that provided by process scaling. While the basic CAM to RF TLB access cannot be substantially changed (i.e., the clock edges are required approximately where they are) a number of changes were implemented to improve the circuit speed. Foremost among these was reformatting some of the RF storage. Specifically, some bits that had been subsequently decoded were changed to fully decoded values in storage to lessen logic depth in the cycle. Additionally, other logic downstream of the TLB was converted to D2 domino for speed. The latter necessitates generating monotonically rising signals to fire the logic, even when the RF is not accessed. This adversely affects the power, but not on the heavily loaded PA bus.

5. RESULTS

The low power features were added to the TLB on a 0.18 μ m version of a microprocessor currently in volume production. Power dissipation with and without these features was measured using Powermill™ [10], running a DSP metric (different from the one discussed above in that it is heavily optimized for DSP throughput) and Dhrystone. Static timing analysis indicated that no new critical timing paths were created. The average MMU power savings were 42% when operating at 600MHz at 1.3V V_{DD} vs. the original design. This evaluation did not include the match line valid logic described above, since it is not expected to provide benefit with all entries valid. Exploitation of this feature will require that software be aware and limit the number of

entries. Powermill measurements indicate 11.3 μ W/MHz TLB power dissipation at 1.3V.

6. SUMMARY

Methods for lowering active power dissipation in TLB's have been described. The techniques have low overhead as measured by size and power cost, and they are applicable to both instruction and data memory pipelines. Implementation in a 0.18 μ m process generation microprocessor results in total MMU power savings of nearly 1/2, depending upon the program. The techniques maintain high circuit performance and are applicable to numerous integrated circuit structures using CAM based access as well as general dynamic register file design. Despite increasing the number and complexity of domino output paths for speed, the TLB achieves 11.3 μ W/MHz active power on a 90nm process technology.

One power saving approach suggested by these results is to consolidate small pages into single larger ones. This also implies that pages may be periodically invalidated or "aged" as suggested in [7] and [8] to limit power by limiting the number of valid but unused entries.

7. ACKNOWLEDGMENTS

The authors gratefully acknowledge the contributions of M. Morrow, S. Strazdus, F. Ricci, S. Demmons, S. Graham and the rest of the XScale Microarchitecture™ design team.

8. REFERENCES

- [1] J. Montanaro, et al., "A 160MHz, 32b, 0.5W RISC Microprocessor, *IEEE JSSC*, 31, pp. 1703-1714, 1996.
- [2] L. T. Clark, et al., "An Embedded 32b Microprocessor Core for Low-power and High-performance Applications," *IEEE JSSC*, 36, pp.1599-1608, 2001.
- [3] M. Ekman, et al., "TLB and Snoop Energy Reduction using Virtual Caches in Low-power Chip Multiprocessors," *Proc. ISLPED*, pp. 243-246, 2002.
- [4] T. Juan, et al., "Reducing TLB Power Requirements," *Proc. ISLPED*, pp. 196-201, 1997.
- [5] J. B. Chen, et al., "A Simulation Based Study of TLB Performance," *Proc. 19th Symp. Comp. Arch.*, pp. 114-123, 1992.
- [6] L. T. Clark and G. F. Taylor, "High Fan-in Circuit Design," *IEEE JSSC*, 31, pp.91-96, 1996.
- [7] Z. Hu, et al., "Timekeeping Techniques for Predicting and Optimizing Memory Behavior," *Proc. ISSCC*, pp. 166-167, 2003.
- [8] Z. Hu, et al., "Managing Leakage for Transient Data: Decay and Quasi-Static 4T Memory Cells," *Proc. ISLPED*, 2002, pp. 52-55.
- [9] D. Harris, *Skew-Tolerant Circuit Design*, Morgan Kaufmann, 2001, p. 112.
- [10] Powermill User Guide, Synopsis Inc., 2001.