The Energy Complexity of Register Files *

V. Zyuban and P. Kogge
December 12, 1997

University of Notre Dame, CSE Department

Abstract

Register files represent a substantial portion of the energy budget in modern processors, and are growing rapidly with the trend towards larger Instruction Level Parallelism (ILP). The energy cost of a register file access depends greatly on the register file circuitry used. This paper compares various register file circuitry techniques for their energy efficiencies, as a function of the architectural parameters such as the number of registers and the number of ports. The Port Priority Selection technique combined with differential reads and low-swing writes was found to be the most energy efficient and provided significant energy savings compared to traditional approaches in the case of large register files. The dependence of register file access energy upon technology scaling is also studied. However, as this paper shows, it appears that none of these will be enough to prevent centralized register files from becoming the dominant power component of next-generation superscalar computers, and alternative methods for inter-instruction communication need to be developed.

Introduction

Current microprocessor design has a tendency towards wider issue and increasingly complex out-of-order execution. This leads to growth of the on-chip hardware, and, consequently, an increase in dissipated power. In [21] the authors have described and analyzed those portions of a microarchitecture where complexity grows with increasing instruction-level parallelism. Among them are: register rename logic, wakeup logic, selection logic, data bypass logic, register files, caches and instruction fetch logic.

These structures usually include multiported memory parts whose storage size (number of entries) and number of ports grows with increasing instruction-level parallelism. For example in [21] the number of read ports in the central CPU register file (RF) is shown to be the product of the number of read operands per instruction and the issue width. Also in [10] it was found that the performance of a four-issue machine with a 32-entry dispatch queue tends to saturate around 80 registers. For an eight-issue machine with a 64-entry dispatch queue performance does not saturate until 128 registers. Thus, we should expect that both the storage size and the number of ports of on-chip memories will grow in the future.

The silicon area of a multiported memory, built using conventional approaches, grows quadratically in the number of ports [25]. Therefore, taking into account growth both in storage size and the number of ports, we can expect that the power portion of multiported on-chip memories will grow rapidly in the future.

A lot of work has been done in estimating the minimum cycle time for on-chip memories. In this work we concentrate instead on the power issue. The existing SRAM power models cannot be applied to on-chip multiported memories because the large number of ports requires different approaches to virtually the entire memory design.

In this paper we will concentrate on the power dissipation of an integer register file. The developed approach can also be applied to other CPU on-chip multiported memories. We will develop an energy model for register files of future machines, with a huge number of read and write ports. The model will express the RF energy in terms of the read and write port number, N_{read} and N_{write}, the number of registers, N_{reg}, and several other relatively simple system parameters and technology parameters. Such a model is badly needed for architectural studies, where we are mostly interested in relative energy (power) estimates that would allow us to compare energy (power) complexity of different architectures. At the architectural level we do not need very accurate, absolute energy (power) estimates, therefore, we will try to keep the model simple.

Another aspect of this paper is that it does not only develop a model for some particular implementation of the RF, but, rather, tries to find the lower bound of the RF power that can be achieved (or approached) by different implementations. This makes this model particularly valuable for architectural studies.

The organization of the paper is as follows: Section 1 defines the terms of the RF access energy common to all techniques. Sections 2 through 6 give an energy analysis of various RF circuitry techniques and a comparison among them. Section 7 applies the developed RF energy model to the superscalar architecture and analyses the dependence of RF access energy and RF power upon the processor issue width and technology scaling. Section 8 summarizes the paper, with some directions for future work.

*This work was supported in part by the National Science Foundation under Grant No.MIP–95–03682.

1. Overall energy

In a superscalar processor several instructions are issued in every clock cycle, and each of them may read one or two operands from the RF, and/or write a result to the RF. There is an energy cost of every read and write access. We assume that the energy cost of every read and write accesses are independent of other accesses that are going on concurrently. Then we can derived the energy cost of every read and write access, and estimate the total energy dissipated in the RF by multiplying these access energy costs by the number of accesses per unit time.

The overall RF access energy (read or write) is the sum of energy dissipated in the decode logic, E_{decode}, memory array, E_{array}, sense amplifiers, E_{SA} (in case of read access), energy dissipated for driving signals that control the operation of the sense amplifiers, precharge circuitry and write drivers, $E_{control}$ (Fig. 1).

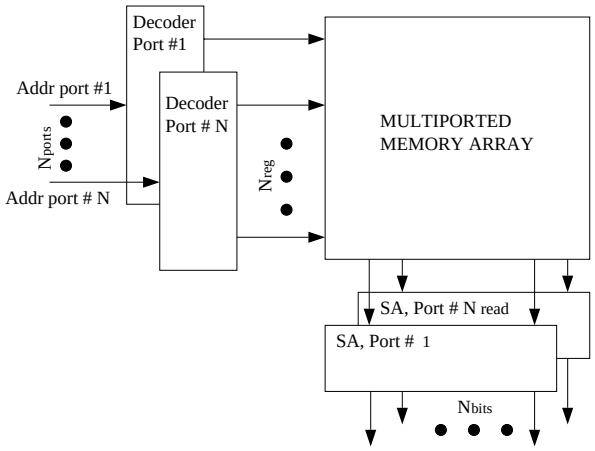

Figure 1. Generic register file memory cell with N_{read} read ports and N_{write} write ports.

To estimate the power dissipated by the decode logic we can use existing approaches, e.g. [8], [9], however, the power dissipated by the select circuitry is typically less than 10% of the total power. Moreover, it does not significantly depend on the register file organization. Therefore we will exclude this portion of the dissipation power from our model. The memory array portion, on the other hand, needs to be studied in detail, because it represents the major portion of the RF power dissipation, and because none of the existing energy/power models can be applied to these highly multiported memory configurations.

2. Conventional approach

We begin the RF access energy study by considering the conventional approach to the RF design. The conventional multiported memory cell for RF, Figure 2, typically uses two bit lines per write port and one bit line per read port, as well as one word line per every port to control the connection of the cell to the bit lines of the corresponding port [21], [25], [16], [19]. Thus, there are $N_{read} + N_{write}$ word lines for every row in the array, and $N_{read} + 2N_{write}$ bit lines. Multiple word lines can go high at the same time in case of simultaneous access through several ports to the same cell. Therefore the cell must be capable of driving significant current which is proportional to the number of read ports. To protect the data stored in the cell during such multiple read accesses, an additional buffer is typically inserted between the cell flip flop and the read pass transistor (Fig 2), further referred to as a decoupling buffer. Because of this decoupling buffer, the read bit line cannot serve as a write bit line, and thus the total of $N_{read} + 2N_{write}$ bit lines are needed.

Single-ended sensing requires higher voltage swing on bit lines than differential sensing, and, consequently, is slower, but it significantly reduces the array area. Therefore, single-ended sensing is usually a preferred approach in current RF design, particularly as the number of ports grows. Differential write bit lines allow fast write and robust noise margins [25], therefore double-ended writes is a typical choice in modern register files.

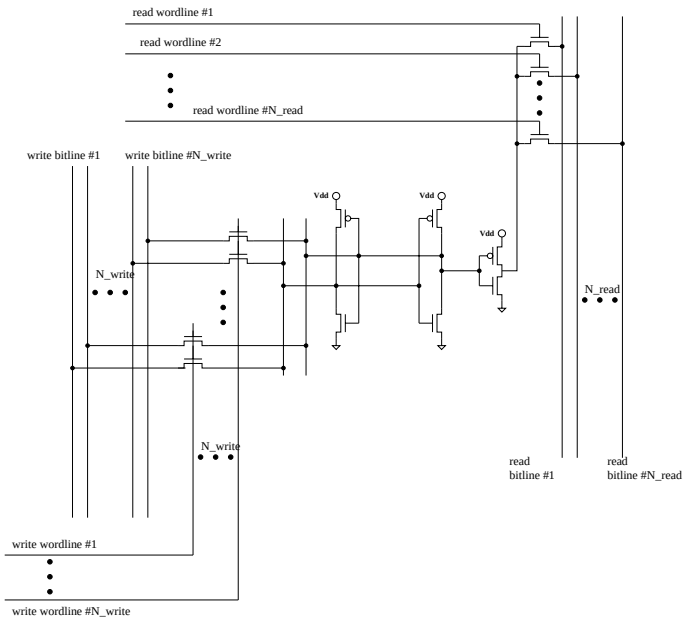

Figure 2. Generic register file memory cell with N_{read} read ports and N_{write} write ports.

2.1. Read access energy

We will consider four terms in the read access energy: the word line energy, $E_{wl,read}$, bit line energy, $E_{bl,read}$, sense amplifier energy, E_{SA}, and energy dissipated for driving control signals, $E_{control}$.

2.1.1. Word Line Energy. Since the voltage swing on word lines is equal to V_{dd}, the energy $E_{wl,read} = V_{dd}^2 C_{wl,read}$ is dissipated for driving every word line. $C_{wl,read}$ is the read

word line capacitance which is the sum of the line capacitance and the gate capacitance of pass transistors, $C_{wl,read} = C_{line} + C_{pass,r}$. In single ended sensing schemes the word line is connected to one pass transistor in every cell, therefore, $C_{pass,r} = C_{gate} W_{pass,r} N_{bits}$, where C_{gate} is gate capacitance per unit width, and $W_{pass,r}$ is the width of the cell read pass transistor. Here and in the following the value of $W_{pass,r}$ is chosen in such a way that the bit line signal sufficient for sensing, V_{sense} can be developed in at least $\frac{T_{period}}{4}$ time, where T_{period} is a typical CPU clocking period for a given technology level.

For the line capacitance we have: $C_{line} = W_{cell} C_{metal} N_{bits}$, where C_{metal} is the metal line capacitance per unit length, and W_{cell} is the cell width. If the number of ports is large enough, then the cell sizes are determined by the bit line and word line crossing area, therefore, $W_{cell} = (2N_{write} + N_{read}) * W_{pitch}$. Thus, we have for the word line energy of one read access:

$$E_{wl,read} = V_{dd}^2 N_{bits} \Big(C_{gate} W_{pass,r} + (2N_{write} + N_{read}) W_{pitch} C_{metal} \Big) \qquad (2.1)$$

2.1.2. Bit Line Energy. In estimating bit line energy we assume that the pulse word line activation technique [14] is employed. The word line is pulsed for the minimum time required for reading data from a cell. As soon as the bit line signal V_{bl} reaches the value sufficient for reliable sensing, V_{sense}, the data from the sense amplifier (SA) output is latched and the word line goes low, disconnecting the cells from the bit lines. There is, however, a limitation on how short the word line activation pulse can be, which we must take into account for short bit lines. We assume that for robustness reasons, the word line activation pulse cannot be made any shorter than $\frac{T_{period}}{4}$, where T_{period} is the CPU clocking period. If bit lines are so short that the signal V_{sense} can be developed in a shorter period, then weaker cell transistors should be used to avoid energy waste. Also, a word line swing control circuitry can be used, as in [15].

In addition, a safety margin must be provided to ensure that the word line pulse is long enough even under process corner conditions. We assume that a 30% margin is sufficient: $M_{margin} = 1.3$ which means that if the bit line signal $V_{bl} = V_{sense}$ is sufficient for reliable sensing under nominal conditions, then the word line should be activated for a longer interval, such that the bit line signal $V_{bl} = M_{margin} V_{sense}$ is developed under nominal conditions.

We also assume that bit line loads are entirely cut off during reading, so that the selected memory cell is the only current source that drives the bit line during reading phase. On the other hand, during the precharge phase, the precharge transistors are the only current source that drives bit lines. Under these assumptions, the energy dissipated for driving bit lines in a read access is $E_{bl,read} = V_{dd} M_{margin} V_{sense} C_{bl,read} N_{bits}$ per access by every port.

Energy can be reduced if we exclude the p-channel transistor from the cell decoupling buffer (Fig. 2), or if the bit line precharge voltage is $V_{dd} - V_{TH}$ or higher, where V_{TH} is the threshold of the n-channel read pass transistor. In this case the bit line is discharged only if zero is read from the cell. If zero is read on the average in P_{zero} percent of all read accesses, then the bit line energy is reduced to

$$E_{bl,read} = V_{dd} M_{margin} V_{sense} C_{bl,read} P_{zero} N_{bits} \qquad (2.2)$$

We have measured that on the Sparc-V8 architecture, in integer programs, $65 - 75\%$ of all bits read from the RF were zeros (bits read from the register $R0$ were not taken into account). This result means that if we store in the cells the compliment values of actual data, and precharge bit lines to an appropriate level, we will reduce the average energy of read accesses by a factor of three or even four. We will extrapolate this result to 64-bit architectures, and use $P_{zero} = 0.3$ in the following.

Half of this energy in (2.2) is dissipated in the cell transistors, and the other half in bit line precharge transistors. In real designs the energy will always be larger than (2.2), and (2.2) gives us the lower bound that can be achieved, if energy is the only concern, and speed is not an issue.

2.1.3. Estimation of the required Bit Line Signal. What bit line signal, V_{sense}, is sufficient for reliable single-ended reading? To reduce the bit line swing sufficient for sensing V_{sense}, the bit line precharge voltage, $V_{precharge}$, should be as close as possible to the threshold of the SA. However, $V_{precharge}$ should be higher than $V_{threshold}$ by a sufficient margin to provide reliable operation in process corners (especially PMOS/NMOS process skews) and in the presence of noise. To reduce the necessary margin, the $V_{precharge}$ and $V_{threshold}$ must change in the same way with temperature and V_{dd} changes, and technology deviations.

It is possible to use a feedback circuitry so that changes in the $V_{precharge}$ follow changes in the $V_{threshold}$. It has been reported in [17] that in a BiCMOS process with the use of operational amplifiers the correct biasing $V_{precharge} - V_{threshold}$ can be controlled within a range of $100mV$, making it possible to sense a $200mV$ signal. However, without the Automatic Bias Control the biasing voltage ranges as much as $1060mV$ in the same experiment.

We assume that with the use of a feedback circuitry built of CMOS transistors only, as in [2], the biasing $V_{precharge} - V_{threshold}$ can be controlled within a range of $300mV$. This value depends on the range of technology deviations, temperature and V_{dd} changes, temperature gradients. It is not likely to scale down with the feature size or with V_{dd}.

Another component to the V_{sense} is noise on the bit lines, caused by capacitive coupling. This component scales down with V_{dd} reduction, and we will use for estimates the $0.1V_{dd}$ value

for the sufficient noise margin. Thus, we will use for V_{sense} in single-ended sensing the estimate $V_{sense} = 2 * 300mV + 0.1V_{dd}$. For comparison, in the UltraSparc RF [25] $V_{dd} = 3.3V$, $V_{precharge} = 1.31V, V_{threshold} = 0.70V$ and $V_{sense} = 1.31V$. In the RF of iWarp [16] $V_{dd} = 5V$ and $V_{sense} = 1.5V$.

2.1.4. Bit Line Capacitance. The bit line capacitance in (2.2) is the sum of the metal line capacitance, C_{line} and the diffusion capacitance of pass transistors connected to the bit line, $C_{pass} = N_{reg}C_{drain}W_{pass,r}$, where N_{reg} is the number of registers and C_{drain} is the drain capacitance of the pass transistor (including the capacitance of a contact), $W_{pass,r}$ is the width of the cell pass transistor, determined as described earlier. For the line capacitance we have: $C_{line} = H_{cell}C_{metal}N_{reg}$, where N_{reg} is the number of registers, C_{metal} is the metal line capacitance per unit wire length, H_{cell} is the cell height. If the number of ports is large enough, then $H_{cell} = (N_{write} + N_{read} + 2) * W_{pitch}$, with the constant 2 standing for the power and ground lines. Thus,

$$C_{bl,read} = N_{reg}\Big(C_{metal}W_{pitch}(N_{write} + N_{read} + 2) + C_{drain}W_{pass,r}\Big)$$

Then, we have for the bit line energy of one read access:

$$E_{bl,read} = V_{dd}M_{margin}V_{sense}P_{zero}N_{bits}N_{reg}\Big(C_{metal}W_{pitch}(N_{write} + N_{read} + 2) + C_{drain}W_{pass,r}\Big) \quad (2.3)$$

We see that if both the number of ports $(N_{write}+N_{read})$ and the number of registers N_{reg} grow linearly with increasing the instruction level parallelism ILP, then we will observe quadratic growth in bit line energy per access. If we take into account that the number of accesses per cycle also grows linearly with ILP, then the energy dissipated per cycle grows as a cube. This will be discussed more fully later.

2.1.5. Energy of Sensing Circuitry. The energy dissipated by sensing circuitry greatly depends on the type of the SA used. The scheme that is normally used is basically an inverter with the input connected to the bit line. Since the bit line precharge voltage is close to the inverter threshold, there is a short-circuit current flowing through the inverter while the SA is activated. The short-circuit current value depends on the feature size and the V_{dd}, as well as on the the width of the inverter transistors. We assume that to save power the transistors are chosen to have the minimum gate width, the value of V_{dd} scales according to the "High-Speed Scenario", described in [12], and the drain saturation current per unit width is $\frac{I_{dsat}}{W} = 0.5mA$, independent of the feature size, which has indeed been observed in practise, [1]. Since the input voltage of the sensing inverter is close to $V_{dd}/2$, the short-circuit current can be estimated as $I_{SA} = \frac{1}{2}I_{dsat}$.

In order to save energy, it is desirable to activate SA's for as short interval as possible, however, there are the same limitations as for the word line activation pulse. As for the word line, we assume that the shortest SA activation pulse we can afford without sacrificing robustness is $\frac{T_{period}}{4}$, where T_{period} is the CPU clocking period achievable for a given technology. Then the energy dissipated in one SA during one access can be estimated as $E_{SA,inv} = \frac{1}{8}V_{dd}T_{period}I_{dsat}$, where I_{dsat} is the drain saturation current per unit width. Given the huge number of SA's that need to be activated at the same time $(N_{read} \cdot N_{bits})$, this represents a significant portion of the total energy of the read access.

2.1.5. Control Signals. The last component to the read access energy is the energy dissipated for driving SA and precharge control signals. We estimate the energy of each of them similarly to the energy of the word line in (2.1):

$$E_{SA,ctrl} = V_{dd}^2N_{bits}\Big(C_{gate}W_{SA,ctrl} + (2N_{write} + N_{read})W_{pitch}C_{metal}\Big)$$

$$E_{precharge,ctrl} = V_{dd}^2N_{bits}\Big(\frac{C_{bl,read}}{40} + (2N_{write} + N_{read})W_{pitch}C_{metal}\Big) \quad (2.4)$$

It is assumed that the size of the precharge transistors is proportional to the bit line capacitance, $W_{precharge} = \frac{C_{bl,read}}{40C_{gate}}$, in order for the precharge time to be independent of the bit line capacitance.

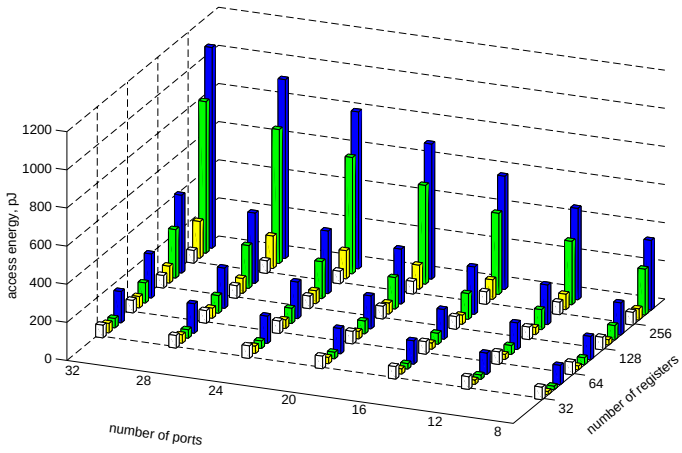

Figure 3. Components to the average read access energy for the 0.5um technology, $V_{dd} = 3.3V$. Bars represent 1.–SA energy; 2.–word line energy plus energy of control signals; 3.–bit line energy; 4.–total read access energy.

Figure 3 shows the main components to the read access energy versus the two main architectural parameters: the number of registers, and the number of ports. Here and in the following we assume $N_{bits} = 64$, $N_{read} = 2N_{write}$, a typical ratio for RISC CPU's. We see that for large register files the bit line energy is the dominant component to the energy of the read cycle. For small register files, on the other hand, the sense amplifier energy dominates.

2.2. Write access energy

The write operation energy in a register file has a higher importance than in memories because in RISC processors the ratio between the number of RF reads and writes is typically reported as about 2:1, compared to the typical 3:1 ratio between the number of loads and stores. Actually, we have measured the $\frac{\text{reads}}{\text{writes}}$ ratio for the RF in the SPARC-V8 architecture to be about 1.6 for most integer programs, meaning writes even more important.

For the write access energy we consider three terms: word line energy $E_{wl,write}$, bit line energy $E_{bl,write}$, and the energy dissipated for driving control signals $E_{ctrl,write}$. For the word line energy we basically have the same formula, as for the read access, except that for differential writes every word line is connected to the gates of two write pass transistors in every cell. These write pass transistors do not need to be any bigger than the minimum transistor size, $W_{pass,w} = 1$. Thus,

$$E_{wl,write} = V_{dd}^2 N_{bits} \Big(2C_{gate} W_{pass,w} + (2N_{write} + N_{read}) W_{pitch} C_{metal} \Big) \quad (2.5)$$

Writes to a cell are usually done by a full-swing signal. Differential write bit lines allow fast write operation and robust noise margins [25]. In this case, however, in every write, one of the bit lines goes all the way to V_{dd}, while the other bit line goes all the way to the ground, resulting in energy dissipation of $C_{bl,write} V_{dd}^2$. The write energy can be reduced if after every write operation we equalize the write bit lines through an equalizing transistor. In this case, in the limit, we save half of the energy stored in the bit line that was at V_{dd} during the write operation, so that

$$E_{bl,write} = \frac{1}{2} C_{bl,write} N_{bits} V_{dd}^2 \quad (2.6)$$

The write bit line capacitance is the same as the read bit line capacitance, except that the write pass transistors have the minimum size independent of the size of the register file,

$$C_{bl,write} = N_{reg} \Big(C_{metal} W_{pitch} (N_{write} + N_{read} + 2) + C_{drain} W_{pass,w} \Big)$$

The energy dissipated for driving the precharge control signals is estimated as for the read access:

$$E_{precharge,ctrl} = V_{dd}^2 N_{bits} \Big(\frac{C_{bl,write}}{40} + (2N_{write} + N_{read}) W_{pitch} C_{metal} \Big) \quad (2.7)$$

The energy dissipated for driving write driver control signal is estimated to be twice as much, $E_{driver,ctrl} = 2E_{precharge,ctrl}$, since there are two write drivers for each column.

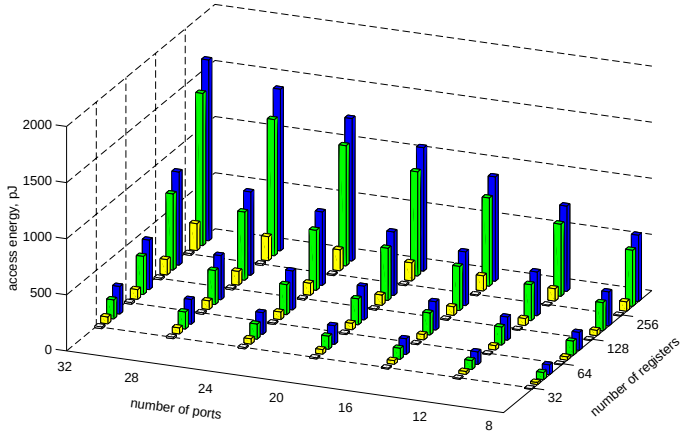

Figure 4. Components to the average write access energy for the 0.5um technology, $V_{dd} = 3.3V$. Bars represent 1.–word line energy; 2.–energy of write driver and precharge control signals; 3.–bit line energy; 4.–total write access energy.

Figure 4 shows the main components to the write access energy versus the number of registers, and the number of ports. We see that the bit line energy is the dominant component to the energy of the write cycle. Comparing Fig. 3 and Fig. 4 we see that the write access energy is substantially higher than the read access energy in the case of large register files.

2.3. Total access energy

Now we can combine all the terms and obtain the average access energy per instruction as a sum of the read access energy and the write access energy taken with the weights according to the measured read/write ratio of 1.6. We have measured on integer programs that for the Sparc - V8 architecture there are on the average 0.95 RF read accesses and 0.6 write accesses per instruction, therefore, $E_{average} = 0.95 E_{read} + 0.60 E_{write}$.

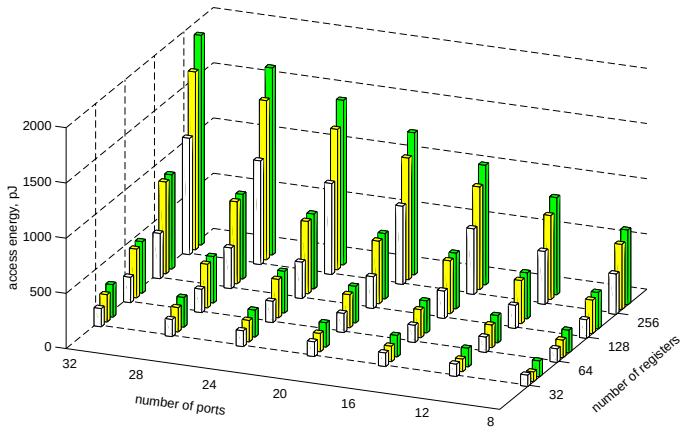

Figure 5. Average read access energy (1st bar), write access energy (2nd bar) and average access energy per instruction (3d bar) for the 0.5um technology, $V_{dd} = 3.3V$.

The results are shown in Figure 5. There is a significant energy penalty per instruction issued in supporting a large (ILP) with a centralized register file. Notice that Fig 5 shows the average energy per instruction. The "worst case" energy is much more then the average energy. To estimate "the worst" case access energy we assume that every instruction issued reads two operand from and writes a result to the RF, besides the values read from the RF are such that all the read bit lines are discharged every other read access. The worst-case access energy is shown in Fig 6.

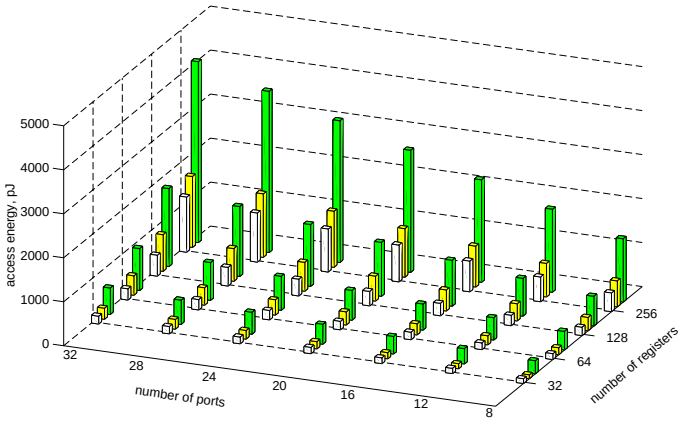

Figure 6. Worst case read access energy (1st bar), write access energy (2nd bar) and worst case access energy per instruction (3d bar) for the 0.5um technology, $V_{dd} = 3.3V$.

2.4. Possible improvements

There are at least two modifications to the conventional RF architecture that could improve the energy efficiency of the register file. The first modification reduces the average energy of the write access, taking advantage of the correlation between consecutive writes. The second modification is to use a synchronous latch-type SA that does not consume constant current.

If the speed of the write operation is not an issue, to save energy we may choose not to precharge write bit lines after writes. In this case the energy $C_{bl,write}V_{dd}^2$ is dissipated only when the value being written by a particular port is different from the previous value written through the same port. Then, neglecting the energy dissipated within the cell, we can write $E_{bl,write} = \alpha C_{bl,write}V_{dd}^2$, where α is the activity factor, that is the percentage of the writes to individual cells in which the value written by a port is different from the previous value written by the same port. We have measured that for the SPARC-V8 architecture α is in the range from 0.25 to 0.35 for most integer applications, and will use the value $\alpha = 0.3$. As noted earlier the write energy can be reduced if before driving the write bit lines we first equalize them through the equalizing transistor, charging the bit line that is to go high by the charge from the bit line that is to go low. In this case, in the limit, we can reduce the write energy twice, so that

$$E_{bl,write<improved} = \frac{1}{2}\alpha C_{bl,write}V_{dd}^2$$

Notice that the single-ended write scheme would result in the same energy, however, it is not as robust as differential write scheme. The described technique can only be applied if the column multiplexing is not used in the memory array, otherwise the data in the cells connected to the selected write word line, but to unselected bit lines, would be falsely overwritten. Therefore, the application of the above technique is limited to memories with hight width and relatively small number of entries (such as register files), where column multiplexing is not needed.

The use of a latch-type SA, such as in [23] or [6] eliminates the dc power component. For this kind of SA, when used in the single-ended read scheme, the reference voltage must be generated however. The power of the latch-type SA consists of the power due to the short-circuit current that flows through the SA while it is in a metastable state and the power due to capacitive loading. We will ignore the latter because it is the same for all sensing schemes. The time during which the SA stays in the metastable state, τ, depends on the technology level, V_{dd}, and the effective voltage difference between the sensing nodes at the beginning of the sensing. To give a simple estimate we will assume that τ is equal to a typical gate switching delay for a given technology level (that is $\tau = \frac{T_{period}}{15}$), and that the SA through current, while it is in the metastable state, is equal to the saturation current through the minimum-size transistor. These assumptions yield the estimate for the SA energy $E_{sa,latch} = \frac{1}{15}V_{dd}I_{dsat}T_{period}$ which is in satisfactory agreement with data we obtained by Spice simulations, as well as with data reported in [9].

The dc component of SA power can also be eliminated by using as a sensing element the circuit proposed in [20] for level conversion of low swing busses. It requires approximately the same swing at the input for reliable operation under V_{TH}, V_{dd} and V_{cc} variations, as the conventional single-ended SA, however the range of voltages at the input which causes the short circuit current is reduced more than 3 times.

Figure 7 shows the average access energy of the improved version of the RF and the original one. We observe almost 40% improvement in energy efficiency for small register files, and about 30% improvement for large RF's. For small RF's the improvement is due to the use of a more energy-efficient sensing scheme which, according to Fig 3, represents the major part of the energy in small RF's. For large RF's the improvement is due to energy savings during write operation.

However, there is a significant design complexity and speed penalty for using the described tricks. In the following sections we consider other, less typical approaches to the RF architecture.

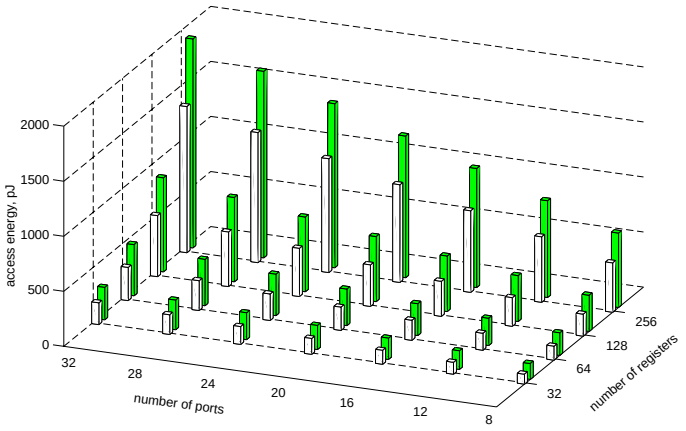

Figure 7. Average access energy per instruction of the improved RF architecture (1st bar), and the original architecture (2nd bar), $V_{dd} = 3.3V$.

3. Use of Current - Direction Sensing Technique for Reads

In this section we will look at the current sensing technique [15] [3] [4] as a way to reduce the bit line energy in the case of heavily loaded bit lines. In [3] the authors justify the use of the current sensing scheme to increase the speed of the sensing scheme. The same idea of having very low bit line swing during sensing is also very attractive from the energy prospective.

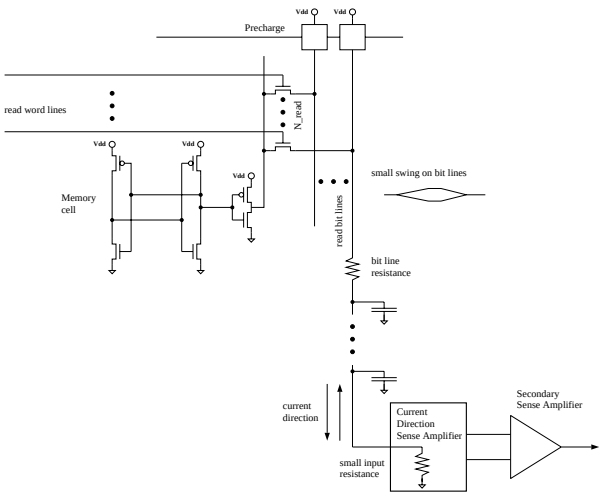

Figure 8. Average access energy per instruction of the improved RF architecture (1st bar), and the original architecture (2nd bar), $V_{dd} = 3.3V$.

The idea of the current-mode SA is based on the observation that the large bit line capacitance will limit energy reduction beyond some point of any sensing scheme that requires a voltage difference be developed on bit lines to initiate sense amplifier operation. This observation leads one to consider current-mode sensing in which the sense amplifier has a very low input impedance and responds to current signals rather than voltage signals [3]. Due to the small impedance at the sensing mode, the signal current from the memory cell can be injected into the sense amplifier without the need for charging or discharging the bit line capacitance, Fig 8. As a consequence, the voltage change on the bit lines during cell access is extremely low, eliminating the source of most (not all, see Section 5) voltage noise coupling problems, and yielding low power dissipation during the sensing operation.

The use of the current-direction sensing scheme affects the read access energy only, and does not to a first approximation affect the write access energy.

For the word line energy, $E_{wl,read}$, and the control signal energy, $E_{control}$ we have the same formulas as before. The only difference is that the cell pass transistors do not need to be as large as in voltage-sensitive sensing scheme, leading to minor energy savings. As for the bit line energy, on the other hand, we would expect more significant energy savings, especially in the case of large register files.

The bit line energy with the current-direction sensing approach can be estimated as $E_{bl,read} = I_{cell}V_{dd}T_{wl}$, where I_{cell} is the cell current during the read access, T_{wl} is the word line activation pulse length.

To minimize the bit line energy we need to minimize both I_{cell} and T_{wl}. The minimum value of the SA input current sufficient for reliable sensing, I_{sense} is determined by the sensitivity of the SA which, in turn, depends on mismatches in transistor parameters in the SA. We have determined through simulation that $I_{sense} = 70uA$ is sufficient for the SA proposed in [15] with 1% mismatches in neighboring transistors. The authors of [15] have implemented the cell with a current of $100uA$ in their design. Therefore we will use the value of $I_{sense} = 100uA$ in our estimates, and assume that this value does not change with technology scaling.

As before, we require the sensing to be done in at least $\frac{T_{period}}{4}$ time, therefore we need that the cell current be at least

$$I_{cell} = \frac{I_{sense}}{1 - exp(-\frac{T_{period}}{4(r_{bl}+r_{SA})C_{bl}})} M_{margin} \qquad (3.1)$$

where r_{bl} is the bit line resistance, and r_{SA} is the sense amplifier input resistance. The denominator in the formula is typically close to unity. If necessary, wider transistors should be used in the cell. If, on the other hand, I_{cell} is smaller than the saturation current through the minimum-size transistor, then the word line swing control circuitry can be used, as in [15] to precisely adjust the cell current, instead of implementing pass transistors with longer gates. Yet, we assume that, as before, the 30% safety margin should be provided to ensure that cell supplies sufficient current even in the process corner conditions.

As to the word line activation pulse length, we assume, as before, that for robustness reasons it should not be shorter than

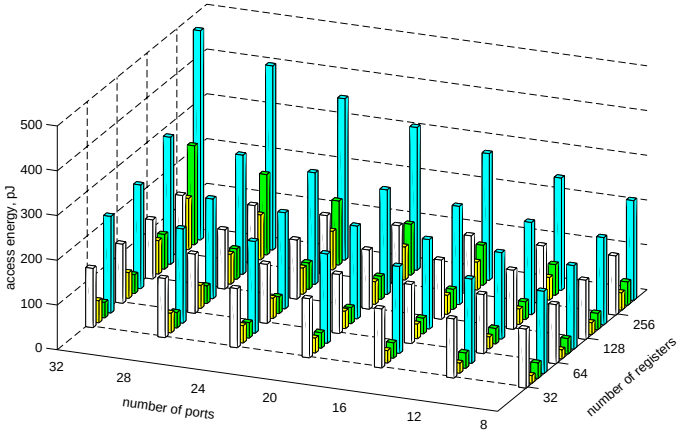

Figure 9. Components to the average read access energy for the current-direction sensing technique, 0.5um technology, $V_{dd} = 3.3V$. Bars represent 1.–SA energy; 2.–word line energy plus energy of SA and precharge control signals; 3.–bit line energy; 4.–total read access energy.

$\frac{T_{period}}{4}$. The restriction on the minimum cell current in (3.1) makes sure that sufficient current flows through the SA by the end of the word line activation period. Thus, we have for the bit line energy of the read access:

$$E_{bl,read} = I_{cell} V_{dd} \frac{T_{Period}}{4} N_{bits}$$

We see that if the ratio $\beta = \frac{T_{period}}{(r_{bl}+r_{SA})C_{bl}}$ is not too small (so that $exp(-/beta) \ll 1$), then the bit line capacitance does not affect the read access energy as significantly as in the voltage sensitive sensing scheme. The bit line resistance is $r_{bl} = R_{line} N_{reg} W_{pitch} (N_{write} + N_{read} + 2)$. The bit line resistance per unit length, R_{line} grows linearly with technology scaling as $R_{line} = \frac{40\Omega/mm}{\lambda}$, where λ is the feature size [1]. The SA input resistance, r_{SA} depends on the SA implementation, and on the current that we allow to flow through the SA (the larger the SA current - the smaller the resistance). If we allow the SA current to be $\frac{I_{dsat}}{2}$ (which is a good estimate for the circuits described in [15] and [3]), then the SA input resistance is approximately 400Ω, almost independent of the technology feature size.

To estimate the SA energy, we assume, as before, that the SA is activated for the period of $\frac{T_{period}}{4}$. Since the SA current is assumed to be $\frac{I_{dsat}}{2}$, then $E_{SA} = \frac{1}{8} V_{dd} T_{period} I_{dsat}$. According to [15], a secondary voltage-sensitive SA is needed to amplify the voltage difference at the output of the current-direction SA to the full swing. We assume that the secondary SA dissipates the same amount of energy, so that, the total energy dissipated by the sensing circuitry is $E_{SA} = \frac{1}{4} V_{dd} T_{period} I_{dsat}$.

Figure 9 shows the components to the read access energy for the RF with the current-direction sensing scheme. For small register files the SA energy is the dominant component. For big RF's the bit line energy dominates.

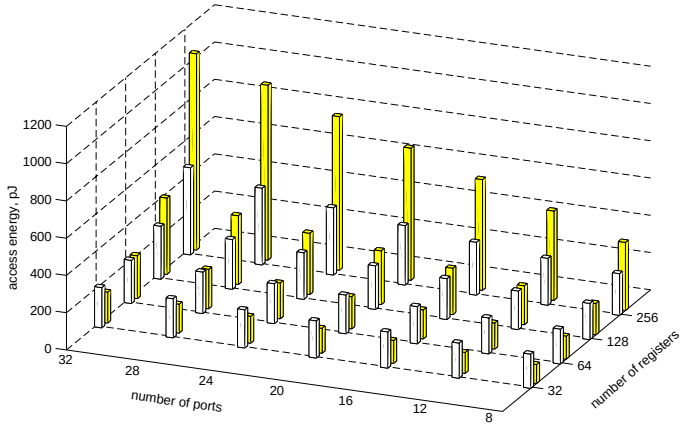

Figure 10. Average read access energy of the RF with the current-direction sensing (1st bar), and the original architecture, with voltage sensitive SA (2nd bar), $V_{dd} = 3.3V$.

Figure 10 compares the average read access energy of the RF with the current direction sensing scheme and that of the conventional RF. For small register files, where the SA energy is the dominant component (Fig. 3) the use of the current-direction sensing technique results in increased read access energy. For large RF, where the bit line energy dominates, the use of the current-direction sensing significantly reduces the energy of read accesses.

4. Use of Differential Sensing Scheme

Differential sensing is a common technique used in RF's [5] [7]. The differential sensing scheme has the advantages of high common-mode rejection and the ability to sense small bit line swings, resulting in lower energy, and a higher speed. The main disadvantage of double-ended sensing in multiported RF's is a significant area penalty.

The derivation of the formulas for read and write access energy of double-ended sensing scheme follows the one for the single-ended sensing scheme in Section 2. The cell width in case of double-ended sensing is larger, however, $W_{cell} = (2N_{write} + 2N_{read}) * W_{pitch}$, resulting in a higher energy for all signals running across the RF, such as $E_{wl,read}$, $E_{wl,write}$, $E_{control}$. Also, additional pass transistors represent an extra load on read word lines, as compared to the single-ended read scheme (though they do not need to be as large as in the single-sensing scheme), so that

$$E_{wl,read} = V_{dd}^2 N_{bits} \Big(2C_{gate} W_{pass,r} + (2N_{write} + 2N_{read}) W_{pitch} C_{metal} \Big) \quad (4.1)$$

In estimating the energy dissipated in bit lines we make the same assumptions as for the single-ended scheme in Subsection 2.1, so that the basic formula for the bit line energy during read access is $E_{bl,read} = V_{dd} M_{margin} V_{sense} C_{bl,read} N_{bits}$ per access by every port. If we equalize the read bit lines through an equalizing transistor, passing the extra charge from the bit line that was high to the bit line that was low, then the bit line energy can be reduced, in the limit, by a factor of 2 (in the following we assume that this thick is not used).

The bit line signal sufficient for reliable differential sensing, V_{sense} is much less than that in the case of single-ended sensing. There are three components to the V_{sense}: offset voltage due to transistor parameter mismatches, capacitive asymmetry and the coupling noise [11]. To be specific, we assume that the cross-coupled latch-type SA is used, because it is both energy efficient and fast [24].

For a typical CPU process the offset voltage is a random variable whose standard deviation is approximately $\sigma = 10mV$, for a careful design of the SA [23]. Given the large number of the SA's, at least 5σ margin is necessary, resulting in the $50mV$ value, which is again not likely to scale down with the feature size. Bit line capacitive asymmetry does not affect the offset voltage as significantly as in [11], because we assume that the SA in a register file is not directly connected to bit lines. We also add to V_{sense} $10mV$ for the differential component of the coupling noise [11] (assume that this component scales down with V_{dd}), and $40mV$ to make up for the bit line precharge error (bit line signal remained on bit lines from the previous accesses) and the difference in the signal on bit lines and the one on sensing nodes of the SA. This difference occurs because as the bit line differential signal grows, the signal at the SA sensing nodes lags in time by $\tau = (r_{SAinput} + r_{bitline}) C_{SA}$ (assume that this component does not scale down with V_{dd}). We also add $50mV$ necessary for fast switching of the SA (assume that this component scales down with V_{dd}), for the total of $V_{sense} = 150mV$, of which the $60mV$ component is assumed to scale down with V_{dd}, and the $90mV$ is assumed not to scale down.

We should notice that with the use of offset-compensating techniques or/and careful centroid layout [13] used for SRAM memories, the offset voltage can be significantly reduced ($4mV$ value was reported in [13]). However, taking into account the huge number of SA's in register files, and area and/or power penalties of these techniques, we assume that they will not be used for RF's in the nearest future.

The use of the differential sensing technique affects the write access energy only through the increased capacitance of all lines running across the RF, and we will omit formulas for the write access energy.

Figure 11 compares the read access energy of the RF with the double-ended sensing scheme and that of the conventional RF. The double-ended sensing yields improvement in energy efficiency for all RF sizes. The improvements is due to the reduced bit line swing during read accesses. The write access energy is slightly more than in the conventional RF architecture because of the increase in capacitance of all lines running across the RF. The primary disadvantage of the double sensing read approach is the area penalty which is approximately 50%. The increase in area results in higher energy dissipation of other components to the total CPU energy, not taken into account in the present model, such as clock distribution energy and energy of driving data busses.

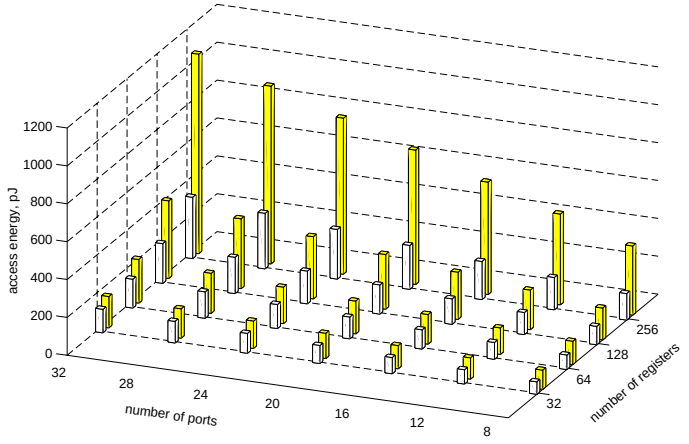

Figure 11. Average read access energy of the RF with the double-ended sensing (1st bar), and the original architecture with single-ended voltage sensitive SA (2nd bar), $V_{dd} = 3.3V$.

5. Low Swing Write Technique

According to the two previous sections, the energy of the read access can be reduced by decreasing the swing on read bit lines, or using current direction sensing techniques. Then, once the read access energy has been reduced, the write cycle energy begins to determine the average access energy per instruction. We saw in Section 2, Fig. 3, that in the conventional RF architecture the energy during the write operation is dissipated primarily in write bit lines, because they need to be charged to a full swing. Consequently, swing reduction during write operation is essential in reducing the write access energy.

One way to reduce the write swing is to use the Driving Source Line (DSL) cell architecture, [18]. The basic idea is to connect the sources of the n-channel transistors in the cell to the source line, controlled by a source line driver, rather than to the ground as in conventional memory cell (as shown in Fig.13 and Fig.14). During reads the source line is driven low, and the cell operates as a conventional cell. During write operation the source line is either left floating as in [18], where the the bit lines are precharged to $V_{dd}/2$, or the source line can be driven high, if the bit lines are precharged to V_{dd} (in the latter case p-channel write pass transistors are needed to pass the small swing from bit lines inside the cell). In both cases the result is that a small swing on the bit line is sufficient to change the potential at the nodes inside the cell. At the end of the write cycle the source line is driven

low, and the cell work as a well known latch-type SA, latching the new data.

Besides lower write access energy, this approach has other advantages. The source of the coupling noise due to full swing during writes is eliminated, allowing a more robust design, and lower bit line swing during read operation. Writing with a low swing is also faster than writing with a full swing, moreover, it allows faster bit line recovery after writes. We expect that low-swing writes will become a common technique in multiported register files. The main disadvantage of low swing writes is that an additional control signal is needed to control the source lines, increasing the design complexity.

The read access energy of this approach depends on the read sensing technique used. A natural combination would be to use the differential sensing technique with the small swing writes. In this case, if read and write operations are separated in time, the same bit line pairs could be used both for reads and writes, reducing the register file area significantly. We will consider this idea in the next section.

The write access energy in the case of using the low swing write technique, consists of the same components, as before, plus the energy for driving the source lines, E_{source} which is estimated similarly to the word line energy.

To estimate the bit line energy it is necessary to know the bit line swing sufficient for reliable writes to a memory cell, V_{write}. It is estimated similarly to V_{sense} for a latch-type SA in the previous section. Some of the terms to the V_{write} for a cell are larger, however, than the corresponding terms to the V_{sense} for a SA. The careful layout used for the SA (including increased channel lengths to reduce variance in transistor drivability) may not be available for the cell because of tighter area constraints, so we assume that the offset for a cell is a random variable with $\sigma = 15mV$, rather than $10mV$ for the SA. Also, since the number of cells in the RF is much larger than the number of SA's, we assume that a 6σ margin is necessary for the cell. Then, since the capacitance at the internal nodes of the cell is higher than that in the SA, we add additional $60mV$ to make up the difference between the signal on bit lines and the inside the cell. As a result, we estimate the value V_{write} for the RF cells to be $250mV$, of which the $100mV$ component is assumed to scale down with V_{dd}, while the other $150mV$ component is assumed not to scale. For comparison, $100mV$ bit line write swing was reported in [18] for a 256 × 32b memory that had just a single read-write port.

Having estimated the minimum bit line swing sufficient for reliable writes, we can estimate the energy dissipated for driving bit lines as $E_{bl,write} = V_{dd} M_{margin} V_{write} C_{bl,write} N_{bits}$ per access by every port. Again some part of this energy can be saved (half in the limit), if we equalize the read bit lines through an equalizing transistor, passing the extra charge from the bit line that was high to the bit line that was low. However, if the write bit lines are precharged to V_{dd} (and this is what we assume in the following), then the use of the equalizing transistor does not save anything.

During the low-swing write operation the cell works similarly to the latch type SA. Therefore, we estimate the energy dissipated in the cell during low-swing write operation to be the same as the energy dissipated in the latch-type SA during differential read operation.

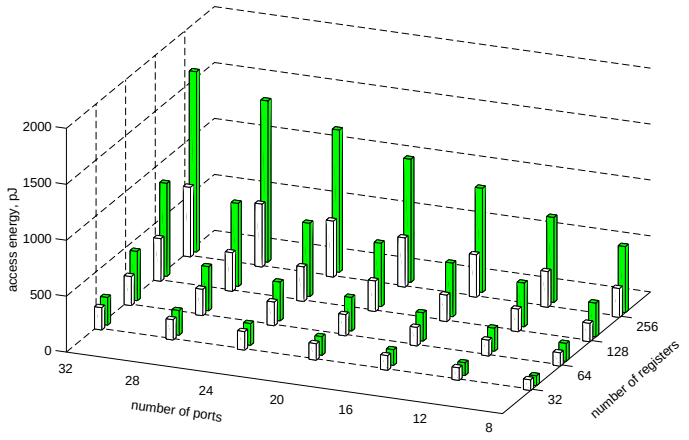

Figure 12. Average write access energy in the RF with the small-swing write technique (1st bar), and the original architecture, with full-swing writes (2nd bar), $V_{dd} = 3.3V$.

Figure 12 compares the write access energy in the RF with the small-swing write technique and that of the conventional RF. The small-swing write technique yields improvement in energy efficiency of the write access for all RF sizes. The improvement is due to the reduced bit line energy dissipation. Energy savings are especially significant for large RF's, where the energy dissipated for driving write bit lines is the dominant component. The use of the small-swing write technique can be combined with various read sensing techniques, described in the preceding sections.

6. Use of Port Priority Selection technique

The Port Priority Selection (PPS) technique, patented by Sun Microsystems (U.S. Pat. No. 5,657,291,), allows us to significantly reduce the register file area and, consequently power, by reducing the number of word lines and bit lines. The number of bit lines can be reduced from $2N_{read} + 2N_{write}$ in the conventional cell with double-ended reading scheme to $2N_{read}$ (assuming $N_{read} \geq N_{write}$). The number of word lines can be reduced from $N_{read} + N_{write}$ in the conventional cell to up to $\log_2 N_{read}$, breaking the well known law of square dependence of cell area upon the number of ports [25].

The idea of the PPS technique is based on the following observations, [22]. First, is it really necessary to provide the possibility of simultaneous read access to the same cell by several read ports? When several ports need data from the same location, we do not actually need to read the multiple copies of data from the cell. One copy would be sufficient. We only need to be able to distribute data read from the cell among all the ports

that need this data. Based on this observation, we can prohibit the simultaneous read access of more than one port to the same cell. If more than one port tries to read data from the same cell, a special priority mechanism chooses among these ports the one with the highest priority, and allows this port to access the cell. All other ports that need the data from the same cell will get the data from the bit lines of the port with the highest priority. For efficient realization of the priority circuitry and the data steering mechanism we refer the reader to [22].

Once we prohibit the cell from being connected to more than one pair of bit lines at a time, the cell circuitry can be simplified. The cell no longer needs to be capable of driving more than one pair of bit lines at a time. Therefore, the decoupling buffer between the cell flip flop and the read pass transistor which serves to protect the data stored in the cell in the conventional design (Fig. 2) is no longer needed. Then, once this decoupling buffer has been removed, the same bit lines can be used both for read and write operations, assuming that read and write operations are separated in time.

In this case the combination of differential reads (Section 4) and low swing writes (Section 5) appears particularly natural, Fig. 13. The only difference between the read and write operations is that the source line is at the ground during read operation, and it goes high during the write operation (assuming V_{dd} precharge of the bit lines). P-channel pass transistors are used to allow low-swing writes with the V_{dd} precharge of the bit lines. If the speed of the read operation is critical, then n-channel pass transistors could be inserted in parallel to the p-channel pass transistors. The use of the same bit line pairs both for reads and writes allows us to reduce the number of bit lines from $2N_{read}+2N_{write}$ to $2N_{read}$, which is the lowest limit that we can get for the differential sensing scheme. If $N_{read} = 2N_{write}$ (a typical ratio in RISC CPU's), then the number of bit lines becomes the same as in the conventional approach with single-ended reads and differential writes (Section 2), meanwhile, as shown in Sections 4 and 5, the energy is significantly reduced.

With the use of the PPS technique, the only information that really needs to be passed to the cell through word lines during read or write operation is the number (ID) of the one of the N_{read} bit line pairs to which the cell needs to be connected. To pass this information we do not need to implement all $N_{read} + N_{write}$ word lines. In the limit, $\log_2 N_{read}$ word lines would be sufficient. In this case, however, this information would need to be fully decoded within the cell, which might not be area efficient and would cause extra delay in the access time. As a compromise, partial port number decoding within the cell allows us to reduce the number of bit lines, without increasing the cell area. Also, with the correct choice of the degree of port number encoding, the total access time is actually reduced, because the speed up in bit line delay which is due to the reduction in the cell height is more significant than the extra delay of the partial decoding of the port number.

In [22] the $\left(\frac{N_{read}}{2}+1\right) \rightarrow N_{read}$ partial decoding scheme

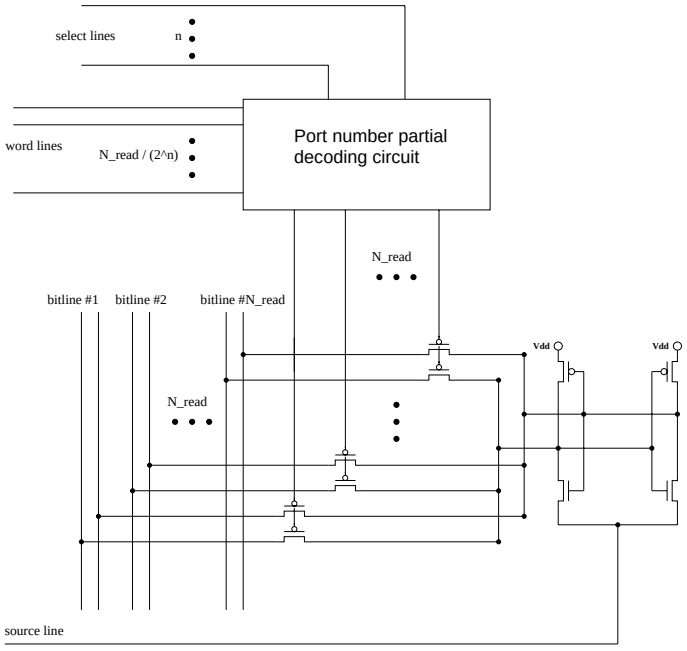

Figure 13. Register file memory cell using the PPS technique, differential reads and low-swing writes. There are N_{read} read ports and N_{write} write ports $(N_{read} \geq N_{write})$.

is used, such that all bit line pairs are divided into two halves, and the signal on an additional select line (Fig.14) indicates to which half of the bit lines the cell is to be connected (or to which half of the bit lines the one-hot signal on word lines refers). In this case a total of $\left(\frac{N_{read}}{2}+1\right)$ lines are needed.

Any of the partial port decoding schemes $\left(\frac{N_{read}}{2^n}+n\right) \rightarrow N_{read}, n = 0, 1, \ldots, \log_2 N_{read}$ could be used, (or $\left(\frac{N_{read}}{2^n}+2^n\right) \rightarrow N_{read}$), where $\left(\frac{N_{read}}{2^n}\right)$ one-hot word lines and n select lines (or 2^n one-hot select lines) are needed, for a total of $\left(\frac{N_{read}}{2^n}+n\right)$, or $\left(\frac{N_{read}}{2^n}+2^n\right)$ lines. Among them the $\left(\frac{N_{read}}{2}+1\right) \rightarrow N_{read}$ scheme is the easiest to implement [22], Fig. 14. This scheme appears to be optimal for a wide range of the number of ports. Therefore, we will assume this scheme in the following energy estimations.

The energy estimation for the RF using the described PPS technique is, basically, the combination of the formulas from Sections 4 and 5 for the differential read and small-swing write schemes. The cell width and height are significantly reduced compared to Sections 4 and 5. Now $W_{cell} = 2N_{read}W_{pitch}$, and $H_{cell} = \left(\frac{N_{read}}{2}+1+2+1\right)W_{pitch}$, where the term $\frac{N_{read}}{2}$ stands for the number of one-hot word lines, term 1 stands for the select line, term 2 stands for power and ground lines, and additional term 1 stand for the source line. Also, the energy dissipated for driving the select lines, term not present in the previous sections, should be taken into account. This term is estimated similarly to the word line energy.

The port priority and data steering circuits consume addi-

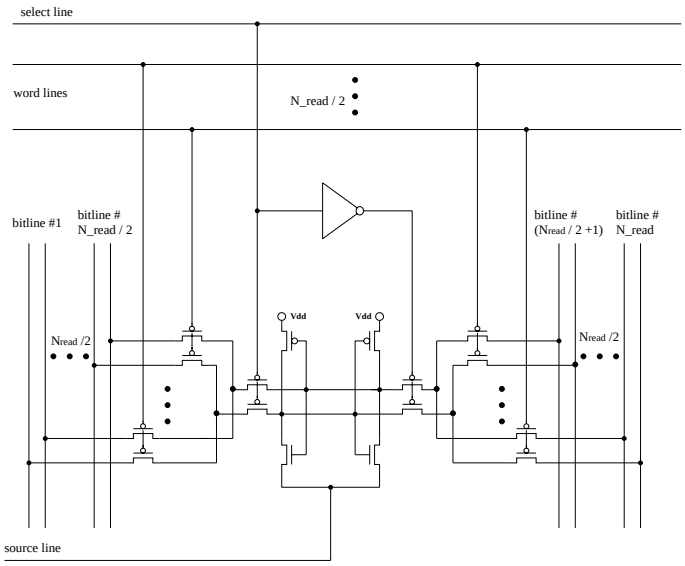

Figure 14. Register file memory cell using the PPS technique with the $\left(\frac{N_{read}}{2}+1\right) \rightarrow N_{read}$ port number decoding scheme. As before, differential reads and low-swing writes are used, and there are N_{read} read ports and N_{write} write ports $(N_{read} \geq N_{write})$.

tional energy (component not present in other RF architectures). However, the additional energy dissipated by these units is estimated to be of the order of 10% of the RF array energy for the small number of ports. Moreover, the asymptotic growth of the energy of these units as the number of ports and registers in the RF grows, is less than that of the RF array energy. Therefore, we do not take into account this energy component.

The results for the read access energy, write access energy and the average access energy per instruction are shown in Fig. 15. For comparison, the 4th bar shows the average energy per instruction of the conventional RF architecture (Section 2). The read access energy and the write access energy of the PPS RF architecture are well balanced. By comparison with the results for the conventional architecture we see that the use of the PPS register file architecture, combined with differential reads and low-swing writes results in significant improvements in energy efficiency of large register files. Besides, for large number of ports, the cell area is reduced almost three times, resulting in reduction in energy of other components, not taken into account in this model, such as clock distribution energy and data routing energy. The conclusion is that the PPS register file architecture combined with differential reads and small-swing writes is a very worthy candidate for future low power register files.

7. Application to Superscalar Processor Architecture

In modern superscalar computer architectures, multiple instructions are fetched, and their execution stared ("issued") at each

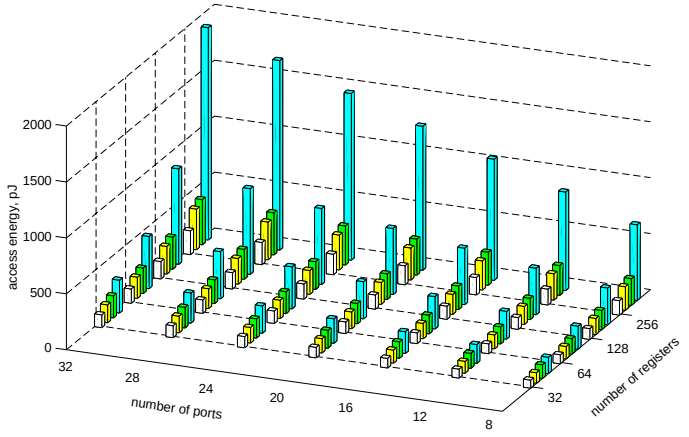

Figure 15. Average access energy for the RF using the PPS technique, double-ended reads and low-swing writes. Read access energy (1st bar), write access energy (2nd bar) and average access energy per instruction (3d bar), 4th bar shows the average access energy per instruction of the conventional RF architecture.

machine cycle. In this section we apply our energy model to such architectures to analyze the dependence of the register file energy efficiency upon the issue width of the processor. To perform this analysis we need to express the number of ports and the number of registers in the RF upon the issue width (IW) of the CPU (or, in other words, draw the line parameterized by IW in the coordinate plane of Fig. 15 and analyze the energy dependence along this line).

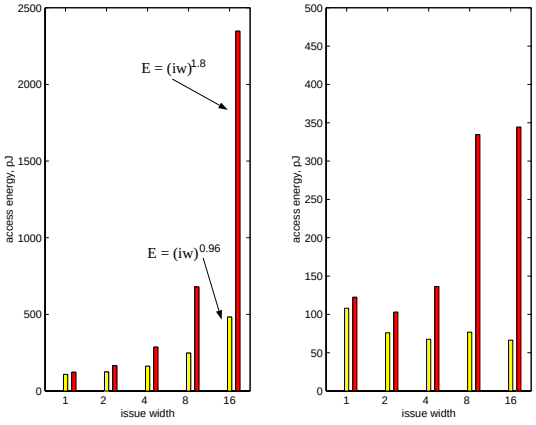

Figure 16. Average access energy per instruction for the conventional RF (2nd bar) and the RF using the PPS technique, described in Section 6 (1st bar) versus the issue width of a superscalar microprocessor. Left chart: 0.5μ feature size, $V_{dd} = 3.3V$, for all points. Right chart: technology feature size scales down with increasing issue width.

For the number of read and write ports needed to provide sufficient bandwidth to the RF, we assume: $N_{read} = 2IW$, and $N_{write} = IW$. To estimate the number of registers needed for CPU to achieve peak performance we use the results reported

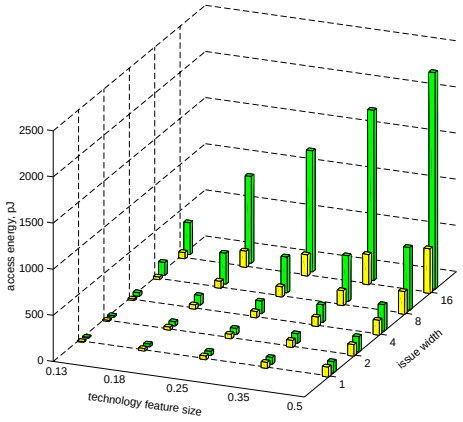

Figure 17. Average access energy per instruction versus issue width and technology feature size, for the RF using the PPS technique, described in Section 6 (1st bar), and the RF using the conventional architecture (2nd bar).

in [10], where the authors simulated machine performance depending on the number of physical registers in the RF. It was found that if the precise interrupt model needs to be supported, then for a four-issue machine with a 32-entry dispatch queue and aggressive assumptions about functional units and caches, 80 registers are needed to achieve the performance that is close (within a few percent) to that of a machine with an unlimited number of registers. For an eight-issue machine with a 64-entry dispatch queue the performance saturates around 128 registers, although at times the processor could use many hundreds of registers. Based on this data, and assuming that 40 registers is sufficient for a simple scalar machine, we extrapolate the dependence linearly to two-issue and 16-issue machines.

Under the above assumptions about the dependence of the number of registers and the number of ports on the issue width, the average RF access energy per instruction versus the microprocessor issue width is plotted in Fig. 16, left chart, both for the conventional RF and the RF using the PPS technique along with double-ended reads and low-swing writes, as described in the previous section. On this chart all register files are assumed to be built using the same process with 0.5μ feature size and $V_{dd} = 3.3V$. We see that there is a significant energy penalty per instruction in supporting large instruction level parallelism with a centralized register file. When we compute the overall RF energy (i.e. the energy per instruction multiplied by the issue width), the number grows even more rapidly.

The left chart in Fig. 16 does not take into account technology scaling, whereas machines with higher issue width are usually built using more advanced processes than machines with lower issue width. To take into account technology scaling we plotted in Fig. 17 the average access energy as a function of issue width and technology feature size. High-speed scenario [12] scaling was assumed, and all assumptions as to the scaling of the bit line swings during reads and writes are as stated in the preceding sections. Fig. 17 shows that if every new machine with higher issue width is built using a more advanced process or, in other words, if we are moving along the diagonal going through point ($\lambda = 0.5\mu, iw = 1$) to ($\lambda = 0.13, iw = 16$) in the coordinate plain, then the average access energy per instruction can be moderated for the RF using the PPS technique along with differential read and low-swing writes. The access energy along this diagonal is extracted and plotted as a one-dimensional chart in Fig. 16, right chart.

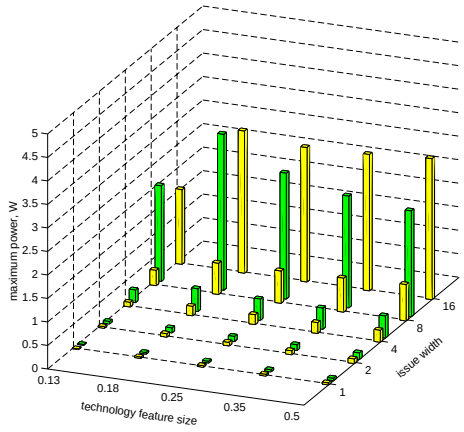

Figure 18. Maximum sustained power of the RF versus issue width and technology feature size, for the RF using the PPS technique, double-ended reads and low-swing writes (1st bar), and the RF using the conventional architecture (2nd bar). Note that this 2nd bar is not shown for $IW = 16$ because it goes significantly off scale.

If we are interested in dissipated power, however, we need to take into account the increase in the clocking rate for smaller technology feature sizes. Also, we must be able to estimate the maximum sustained power of the RF, dissipated if the maximum number of instructions, determined by the issue width of the processor, are issued every clock cycle. Assuming linear increase in the clocking rate with technology scaling, we plotted in Fig. 18 the maximum sustainable power versus issue width and technology feature size. We see that the maximum power grows rapidly with increasing issue width, even for the RF with the PPS architecture built on more and more advanced technology processes. The power for the conventional RF architecture is not shown for the issue width of 16 because it approaches $20W$. Also, for the issue width of 16 the use of the conventional RF architecture makes the desired clocking rate hard to achieve, because the time needed for the development of the sufficient signal on bit lines does not fit into the desired $\frac{T_{period}}{4}$ interval.

It must also be stressed that all the analysis discussed above assumed the very aggressive energy management techniques described in this paper, including pulse word line activation technique for reducing the bit line swing to the minimum, pulse activation of the sensing circuitry, fully cutting off precharge during reading and writing, taking advantage of the statistics of the data stored in RF memory cells, minimum transistor sizing wherever

is possible, use of equalizing transistors to save bit line energy during precharge, also the short-circuit currents were assumed to be negligible. In real wold CPUs the desire for speed will often not permit some of these techniques, meaning that real register file powers are even higher. The conclusion is that none of the known circuit techniques solves the problem of rapid RF power growth for the machines with increasing ILP. Neither does aggressive technology scaling solve the problem for high-speed microprocessors. This result should motivate the development of inter-instruction communication mechanisms alternative to the centralized register file.

8. Conclusions

In this paper we have developed energy models for multiported registers files with a variety of architectural variations. Model parameters included the number of registers and the number of access ports. These models were then used to compute energy efficiencies for individual read and write accesses, and the results were weighted to obtain a ”per instruction” energy measure.

The port priority selection technique combined with differential reads and low-swing writes was found to be the most energy efficient, and seems to provide significant energy savings in comparison to traditional approaches, especially for large register files.

Given the critical role played by centralized register files in modern superscalar computer architectures, we then expressed the number of registers and number of register file ports as a function of Instruction Level Parallelism (ILP), and applied those parameters to our models. Even assuming aggressive technology scaling to track the growth in register file requirements, the resulting power growth is huge, and may begin to swamp the power budgets for future microprocessors. This leads to the inescapable conclusion that the use of a centralized register file as an inter-instruction communication mechanism is going to become prohibitively expensive. Alternative techniques involving more than just circuit tricks are going to be necessary, and are a target of some of our future work.

References

[1] S. Asai and Y. Wada, “Technology Challenges for Integration Near and Below 0.1um.” *Proceedings of the IEEE*, Vol.85, No.4, April 1997.

[2] C. Asato, “A 14-Port 3.8-ns 116-Word 64-b Read-Renaming Register File”, *IEEE Journal of Solid-State Circuits*, Vol.30, No.11, November 1995.

[3] T. Blalock and R. Jaeger, “A High-Speed Clamped Bit-Line Current-Mode Sense Amplifier”, *IEEE Journal of Solid-State Circuits*, Vol.26, No.4, April 1991.

[4] T. Blalock and R. Jaeger, “A High-Speed Sensing Scheme for 1T Dynamic RAM’s Utilizing the Clamped Bit-Line Sense Amplifier”, *IEEE Journal of Solid-State Circuits*, Vol.27, No.4, April 1992.

[5] G. Blanck and S. Krueger, “SuperSPARC: A Fully Integrated Superscalar Processor.” in Hot Chips III, Palo Alto, August 1991.

[6] J. Brockman, V. Zyuban, et al, “SRAM with Multiple Sensing Schemes”, *Notre Dame CSE Technical Report No.97-15*, June 1997.

[7] D. Dobberpuhl, et al, “A 200 MHz 64b Dual-Issue CMOS Microprocessor”, in Digest of Technical Papers, 1992 IEEE International Solid-State Circuits Conference, pp. 106-107, February 1994.

[8] Robert Evans and Paul Franzon, “Energy Consumption Modeling and Optimization for SRAM’s.” *IEEE Journal of Solid-State Circuits*, Vol.30, No.5, May 1995.

[9] Robert Evans, “Energy Consumption Modeling and Optimization for SRAM’s”, Ph.D. dissertation, ECE Dept., North Carolina State Univ., Raleigh, NC, July 1993.

[10] K. Farkas, N. Jouppi, P. Chow, “Register File Design Considerations in Dynamically Scheduled Processors.” Technical Report 95/10, Digital Equipment Corporation Western Research Lab, November 1995.

[11] H. Geib, W. Weber, E. Wohlrab and L. Risch, “Experimental Investigation of the Minimum Signal for Reliable Operation of DRAM Sense Amplifier.” *IEEE Journal of Solid-State Circuits*, Vol.27, No.7, July 1992.

[12] Chenming Hu, “Future CMOS Scaling and Reliability.” *Proceedings of the IEEE*, Vol.81, No.5, month 1993.

[13] K. Ishibashi, et al., “A 12.5-ns 16-Mb CMOS SRAM with Common-Centroid-Geometry-Layout Sense Amplifies”, Vol.29, No.4, April 1994.

[14] K. Itoh, K. Sasaki and Y. Nakagome, “Trends in Low-Power RAM Circuit Technologies.” In: *Proceedings of the IEEE*, Vol. 83, No.4, April 1995.

[15] M. Izumikawa and M. Yamashina, “A Current Direction Sense Technique for Multiport SRAM’s.” *IEEE Journal of Solid-State Circuits*, Vol.31, No.4, April 1996.

[16] Richard Jolly, “A 9-ns, 1.4-Gigabyte/s, 17-Ported CMOS Register File.” *IEEE Journal of Solid-State Circuits*, Vol.26, No.10, October 1991.

[17] T. Kuroda, T. Fukunaga, et al. “Automated Bias Control (ABC) Circuit for High-Performance VLSI’s.” *IEEE Journal of Solid-State Circuits*, Vol.27, No.4, April 1992.

[18] H. Mizuno and T. Nagano, “Driving Source-Line Cell Architecture for Sub-1-V High-Speed Low-Power Applications.” *IEEE Journal of Solid-State Circuits*, Vol.31, No.4, April 1996.

[19] J. Montanaro, et al, “A 160-MHz, 32-b, 0.5-W CMOS RISC Microprocessor.” *IEEE Journal of Solid-State Circuits*, Vol.31, No.11, November 1996.

[20] Y. Nakagome, K. Itoh, et al, “Sub-1-V Swing Internal Bus Architecture for Future Low-Power ULSI’s.” *IEEE Journal of Solid-State Circuits*, Vol.28, No.4, April 1993.

[21] S. Palacharla, N. Jouppi, J. Smith, “Complexity-Effective Superscalar Processor.” In: *Proceedings of the 24th Annual International Symposium on Computer Architecture*. pp. 206–218, June 1997.

[22] U.S. Patent No. 5,657,291, issued Aug. 12, 1997 to A. Podlesny, G. Kristovsky, A. Malshin, “Multiport Register File Memory Cell Configuration for Read Operation.” Assignee: Sun Microsystems, Inc., Mountain View, CA.

[23] R. Sarpeshkar, J. Wyatt, N. Lu and P. Gerber, “Mismatch Sensitivity of a Simultaneously Latched CMOS Sense Amplifier.” *IEEE Journal of Solid-State Circuits*, Vol.26, No.10, October 1991.

[24] A. Sekiyama, T. Seki, S. Nagai, A. Iwase, N Suzuki, and M. Hayasaka, “A 1-V Operating 256-kb Full-CMOS SRAM.” *IEEE Journal of Solid-State Circuits*, Vol.27, No.5, May 1992.

[25] M. Tremblay, B. Joy and K. Shin, “A Three Dimensional Register File For Superscalar Processors.” In: *Proceedings of the 28th Annual Hawaii International Conference on System Sciences*. pp. 191–201, January 1995.