A 3.6 pJ/Access 480 MHz, 128 kb On-Chip SRAM With 850 MHz Boost Mode in 90 nm CMOS With Tunable Sense Amplifiers

Stefan Cosemans, *Student Member, IEEE*, Wim Dehaene, *Senior Member, IEEE*, and Francky Catthoor, *Fellow, IEEE*

***Abstract*—An extremely low energy per operation, single cycle 32 bit/word, 128 kb SRAM is fabricated in 90 nm CMOS. In the 850 MHz boost mode, total energy consumption is 8.4 pJ/access. This reduces to 3.6 pJ/access in the normal 480 MHz mode and bottoms out at a very aggressive 2.7 pJ/access in the 240 MHz low power mode. Several techniques were combined to obtain these performance numbers. Short buffered local bit lines reduce the impact of the cell read current on memory delay. Extended global bitlines are used which improves delay and energy consumption and which reduces the number of sense amplifiers in the memory to 32. Cell stability and speed issues are avoided by applying selective voltage scaling. Novel, digitally tunable sense amplifiers and a tunable timing circuit cope gracefully with the stochastic variations in the periphery.**

***Index Terms*—SRAM, low-power, embedded memory, sense amplifier calibration, dynamic cell stability, calibrated timing, bit line hierarchy, selective voltage scaling, variability-aware design.**

I. Introduction

SMALL on-chip SRAMs are essential to enable low-power platforms that rely on distributed scratch pads. In technologies beyond 130 nm, low power SRAM design is severely complicated by intra-die variations and leakage. For SRAM cells, leakage reduction has been obtained with low supply voltages [1], [2] and high threshold (HVT) transistors [3]. The cell stability issues and the increase in worst read delay due to intra-die variations have been mitigated with high cell supply voltages [4], with larger cells [1], [2], [5] or with more complex local peripherals [1], [4]–[6]. For peripherals, the leakage problem has been addressed with dynamic voltage scaling (DVS) and supply gating. Less research effort has addressed two major issues related to uncorrelated stochastic intra-die variations: 1) replica based timing is no longer effective and 2) sense amplifier (SA) performance becomes severely restricted by these variations. In [1], a very good first attempt was made to resolve this problem by introducing SA redundancy.

Manuscript received November 09, 2008; revised January 27, 2009. Current version published June 24, 2009. This work was supported by IMEC and by Cadence Research Labs.

S. Cosemans and W. Dehaene are with the ESAT-MICAS Laboratory, Katholieke Universiteit Leuven, B-3001 Leuven, Belgium (e-mail: stefan.cosemans@esat.kuleuven.be).

F. Catthoor is with IMEC-NES, B-3001 Leuven, Belgium, and also with the Katholieke Universiteit Leuven, B-3001 Leuven, Belgium.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSSC.2009.2021925

TABLE I
MEMORY DIMENSIONS

technology	90nm ; 1P9M ; multi V_T
word length	32 bit
memory size	4K word ; 128Kbit
cell type	single-ended SRAM pull-down path blocked during write
cell size (logic DRC rules)	1.51 μm x 1.3 μm = 1.96 μm²
matrix size	772 μm x 634 μm =490 000 μm²
memory size	504 000 μm²

TABLE II
MEASURED PERFORMANCE

main Vdd [V]	1.2	1.0	0.8	0.6
max frequency [MHz]	850	780	480	240
static energy/access[1] [pJ]	1.3	0.8	0.5	0.7
active energy/access [pJ]	7.1	4.5	3.1	2.0
total energy/access [pJ]	8.4	5.3	3.6	2.7
leakage power[2] [μW]	1123	625	219	162

[1] static energy per access is calculated by integrating the leakage power over one idle clock period at max frequency
[2] already included in total energy/access as static energy/access

This work provides detailed information on an SRAM in a triple-V_T 90 nm CMOS process which targets an extremely low energy per operation at acceptable speed [7]. Table I gives an overview of the main memory dimensions. The memory can be used in different operation modes which cover a 3x wide range of energy/operation and speed, as shown by the measured performance numbers in Table II.

The presented design addresses all of the mentioned issues associated with the increased leakage and uncorrelated stochastic intra-die variations in 90 nm. SA performance is improved by the use of novel, digitally tunable SAs. This allows for lower swings on the long wires from cells to memory output, which reduces both energy consumption and delay. The problem of generating correct timing signals for the memory is addressed by the use of digitally tunable delay lines. This avoids the problems associated with replica based timing and allows for a design with more accurate safety margins, with an optimal timing setting for each speed mode. At the same time, design-time risks are reduced. Cell stability issues in the low voltage, low speed modes are resolved in a rather trivial yet very effective way: the supply voltage for the memory cell and word line is not reduced along with the main power supply. This technique has been used previously under different names (e.g., [4]), but we prefer to use the more general term *selective voltage scaling*. We will show

that this results in more than 95% of the maximal obtainable reduction in dynamic energy while the impact on speed is much smaller and all cell stability issues are avoided. Cell leakage is controlled by using only HVT transistors in the memory cell. Using HVT transistors for the cell reduces the cell read current I_{read}, especially at reduced supply voltages. In traditional designs, cell read current has a large impact on memory speed. The use of selective voltage scaling avoids the dreadful combination of HVT and low voltage. The impact of HVT cells on speed is further reduced by using the short, buffered local bit line and the extended global bit line described in [6]. As an additional benefit, the extended global bit line structure reduces the number of SAs in the design from 256 to 32, which reduces the total area overhead of the tunable sense amplifier scheme to less than 1%. This bit line structure was discussed before in [6], but this paper is the first to provide a quantitative analysis of the behaviour of this bit line structure under intra-die variations, showing a reduced impact of these variations on memory access time. The memory employs a fully subdivided word line and a low power global decoder, which will be described in detail.

This paper is organized as follows. Section II describes the organization and operation of the memory, with special attention for the decoder, the bit line structure and the memory timing. Section III provides details on the local block, including a description of the memory cell and local bit line periphery. Section IV provides a detailed discussion of the tunable sense amplifiers that are used in this design, including a comparison with other available options. Section V discusses the use of selective voltage scaling. Section VI contains the measurement results and provides a short comparison with other state-of-the-art low power memories.

II. Memory Organization and Operation

This section describes the organization and operation of the memory. First, the general organization and floor plan are presented. Next, the decoder is presented in some detail. This is followed by a discussion about the bit line hierarchy covering both read and write operation. This section concludes with an overview of the memory operation and a discussion of the digitally tunable timing solution used in this design.

A. General Organization

Fig. 1 shows the global memory organization. The memory matrix is build up of local blocks, which each contain 32 words (1 K cells). The top level matrix is subdivided in 8 columns by 16 rows of these local blocks. For each two local blocks there is a shared write block, which contains the write receivers. These receivers enable low swing signals on the global bit lines during write operations. The memory is based on a fully subdivided word line (WL) architecture, so each word has its own local WL (LWL). This LWL avoids the problems associated with the half-select condition for cells. It also ensures that only the required cells and bit lines are activated, which is crucial to obtain a low energy per operation. In Fig. 1, there are 512 GWLs for 256 cell rows. This apparent inconsistency is caused by the fact that the cells of two words are interleaved as shown in Fig. 1. Each WL spans 64 cells, but is only connected to 32 of them. This

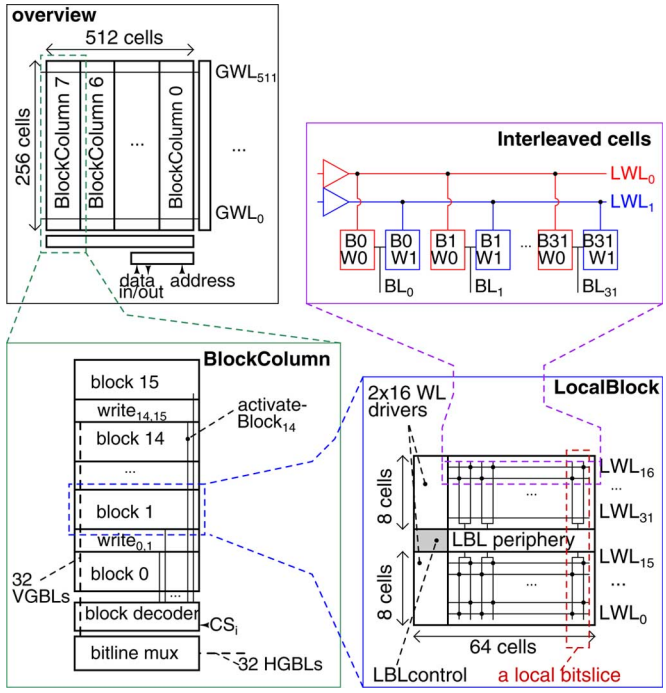


Fig. 1. Global memory organization.

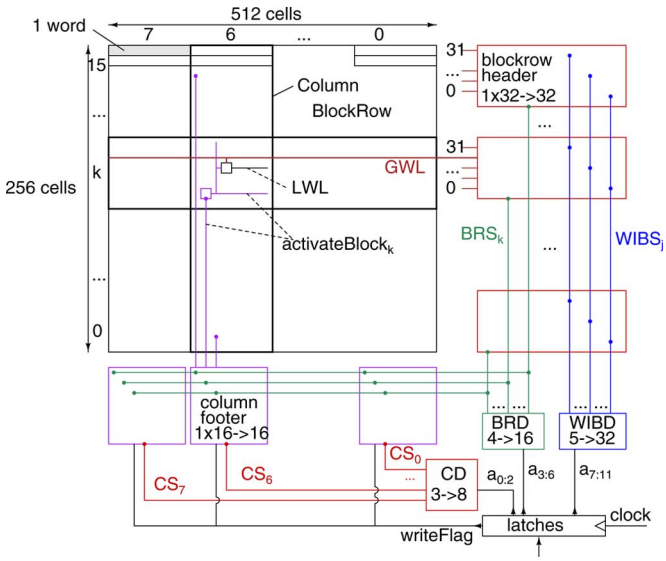


Fig. 2. Decoder organization.

requires two WLs per cell row. This approach reduces the bit line capacitance: for a given number of cells connected to the bit line, the bit line wire length is halved, which reduces its parasitic capacitance.

B. Decoder

Fig. 2 gives an overview of the three-stage decoder that is used in this design. Three first-level decoders are used to decode the 12 address bits into three sets of one-hot output wires. The *column decoder* (CD) decodes address bits $a_{0:2}$ into 8 column select (CS) wires. The *block row decoder* (BRD) decodes address bits $a_{3:6}$ into 16 block row select (BRS) signals, and the *within block decoder* (WIBD) decodes the remaining address

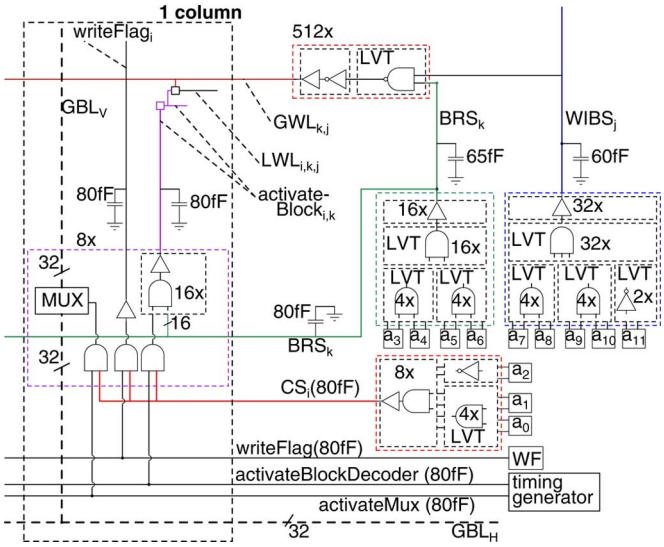


Fig. 3. Decoder implementation.

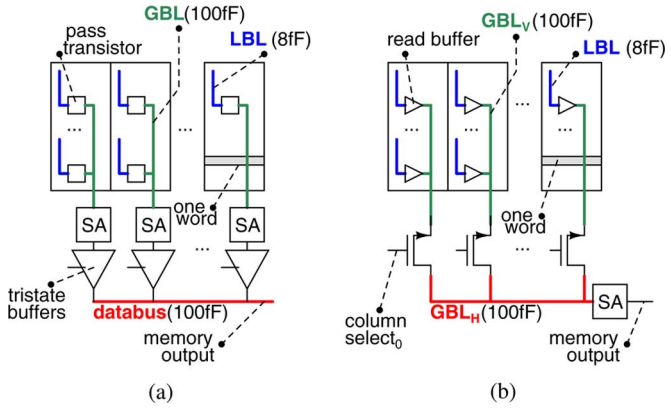


Fig. 4. Solutions to transfer data from the cell to the memory output using hierarchical bit lines and subdivided word lines. (a) Traditional solution, (b) Solution proposed in [6]: readbuffer and extended GBL. This structure is used in this design as well.

bits $a_{7:11}$ into 32 within block signals (WIBS). These WIBS signals indicate which word within the selected local block should be enabled.

There are two groups of second-level decoders. In the column footers, the 16 BRS signals are combined with the column select signal CS_i to generate the *activateBlock* signals. Each of these signals is directed to a single local block, where it is amplified before being used to control local block operation. In the block row headers, the 32 WIBS signals are combined with BRS_k to generate the global word line (GWL) signals for this block row. Notice that the BRS signals are used twice in this decoder—once to obtain GWL and once to obtain *activateBlock*. This results in a deeper logic depth but in a lower capacitive load and energy consumption. The last level decoder combines the *activateBlock* signal with the GWL signal to activate a single LWL (one out of $2^{12} = 4096 = 8 \cdot 32 \cdot 16$).

Fig. 3 contains a more detailed view of the decoder. The mentioned capacitances are only the wire parasitics. The length of the wires in some wire sets varies, e.g., for CS_i. In these cases, worst case values are shown here. All decoder stages are and-type. The decoder consists of static CMOS gates only. The small, highly active stages in the decoder and the input latches use low threshold (LVT) transistors to improve performance at reduced supply voltages without introducing excessively large leakage currents.

The BRS, WIBS and CS signals only change their values when their input address bits change. However, *activateBlock* must be return-zero to ensure a correct precharge operation for the local bit lines. This behaviour is obtained by introducing a timing pulse *activateBlockDecoder* into the decoder gates that generate *activateBlock* in the column footers. To avoid glitches, the start of this pulse is delayed enough to ensure that the GWL signal arrives first at the local block.

The *writeFlag* signal is needed inside the local block. It is routed to the column footer, where it is combined with the static CS signal. The resulting signal $\mathrm{writeflag}_i$ is routed to all local blocks in the active column. The CS signal is also used to control the connection between vertical global bit lines GBL_V (one set per column) and the horizontal global bit lines GBL_H (one set for the memory). Timing for this operation is controlled by the *activateMux* timing signal.

C. Bit Line Hierarchy: Read

The memory design presented in this paper employs a short, buffered local bit line (LBL) and extended global bit lines (GBL), as previously discussed in [6]. This basic structure will now first be compared with available alternatives without delving into too much implementation details. There is some overlap with the explanation in [6], but this provides useful background information for the new quantitative analysis of the behaviour under intra-die variations that is presented in II.C.2. The implementation details of the GBL operation will be provided in Subsection II.E. Implementation details regarding the LBL operation will be discussed in Section III.

1) Buffered LBL and Extended GBL: Fig. 4(a) shows the approach normally used to transfer data from the cell to the memory outputs when using hierarchical bit lines and subdivided word lines. At the column level, sense amplifiers amplify the voltage difference on the bit lines to a full level signal. In traditional designs, this amplification at the column level is required to limit the impact of $I_{\mathrm{read,cell}}$ on the memory speed. After this amplification, the data still needs to be transmitted to the memory output.

In the buffered LBL approach, the readbuffer can easily deliver more current. This allows the global bit lines to be extended to the memory output, as shown in Fig. 4(b). This is the bit line structure used in the SRAM prototype. A major advantage of this architecture is the fact that only one set of sense amplifiers is needed for the entire memory, rather than one set per column. This allows the use of more advanced sense amplifiers. As discussed in [6], this architecture also provides a performance improvement over the more traditional setup from Fig. 4(a) and it enables the use of dynamic read stability.

The approach with readbuffer has some potential drawbacks. The full-swing transition on the LBL consumes energy and introduces an additional delay step. In very small memories, this cost will not be compensated for by the faster and more energy efficient GBL operation. The GBL precharge voltage must be

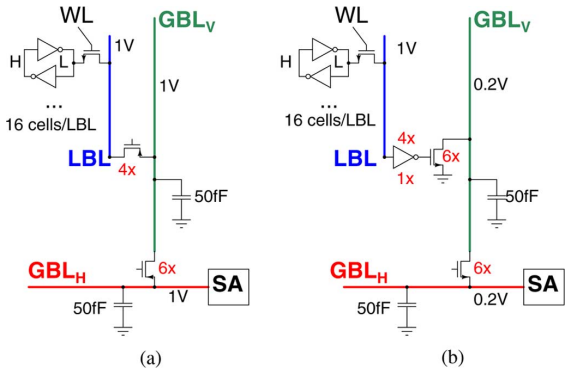


Fig. 5. Simulation setup to compare extended GBL schemes with and without readbuffer. The capacitances in this test case are representative for a smaller memory than the 512×256 cells of the implemented prototype SRAM. Precharge voltages are displayed next to the wires. Cell supply voltage is 1.2 V, all other supplies are at 1 V. Transistor sizings are displayed next to the transistors in multiples of 120 nm. Both cells are traditional 6 T cells with all standard performance HVT transistors with width $= 220$ nm and length $= 90$ nm. All other transistors are LVT. (a) Without readbuffer. (b) With readbuffer.

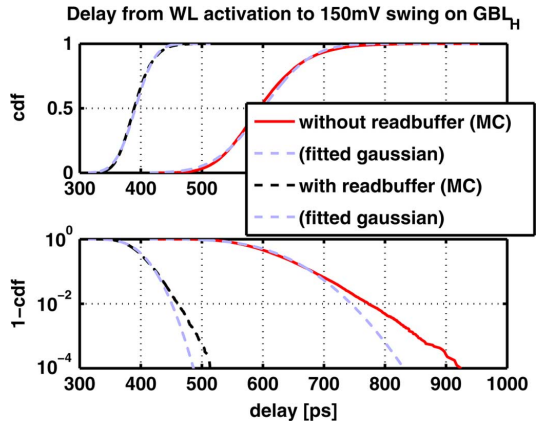


Fig. 6. Distributions of the delay from WL activation to 150 mV swing on GBL_H for the two setups in Fig. 5 as obtained by performing 20 K Monte Carlo runs. Gaussian distributions have been fitted to both full datasets. All curves are displayed both on a linear and a logarithmic scale.

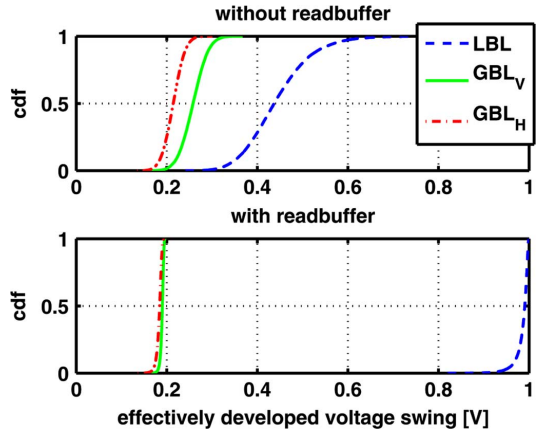


Fig. 7. Distribution of the effectively developed swings on the different bit lines at the time when 99.9% of the cells have developed a swing of 150 mV or more on GBL_H. In the text, this is called the idealised stop time.

generated in a rather stable way. Area consumption and leakage power will have to be analysed on a case-by-case basis as the involved trade-offs are rather subtle.

2) Behaviour Under Intra-Die Variations: The insertion of a readbuffer between LBL and GBL reduces the impact of the cell read current $I_{\mathrm{read,cell}}$ on the memory speed. This becomes especially beneficial in technology nodes with large intra-die variations. In these technology nodes, the cell read current varies widely because cell transistors must be kept as small as possible to preserve area. Another problem for $I_{\mathrm{read,cell}}$ occurs with low-power designs: to limit the amount of standby leakage, the cells must be implemented using high threshold transistors. This reduces cell read current, and further aggravates the issue of variations. The readbuffer current suffers less from these intra-die variations for two reasons. The readbuffer does not have to use HVT transistors, and because the size of the readbuffer transistors has less impact on the memory area, they can be made somewhat larger.

Fig. 5 shows the two simulation setups for which the performance under intra-die variations will be compared in the next paragraphs. This quantitative analysis was not provided in [6]. In the setup without readbuffer, a pass transistor connects LBL and GBL, and their precharge voltage must be equal. Due to cell stability requirements, the precharge voltage of the LBL cannot be close to 0 V. A precharge voltage around vdd/2 would result in a problematic switch resistance because both NMOS and PMOS have only a small overdrive voltage at this source voltage. In this exercise, we use a 1 V precharge voltage. In the setup with readbuffer, the precharge voltage on GBL can be selected independently from that of the LBL. In this setup, GBL is precharged to 0.2 V. The use of a DC-DC converter to generate this precharge voltage could result in an even further energy reduction, but this has not been considered for this comparison. In both setups, it is assumed that the SAs require an input swing of 150 mV to perform reliable sensing, so delay is measured from WL activation to 150 mV swing on GBL_H.

20000 Monte Carlo (MC) runs were performed to obtain a good impression of the impact of intra-die variations. Fig. 6 shows the resulting cumulative distribution of the delay for both setups. The nominal delay for the setup without readbuffer is 600 ps, for the setup with readbuffer, it is only 400 ps, so the readbuffer provides a significant speedup. The ratio of the 99.9% percentile to the mean value can be considered as a crude estimate for the sensitivity to intra-die variatons. For the setup with readbuffer, this ratio is $(\mathrm{delay}_{99.9\%})/(\mathrm{delay}_{50\%}) = 126.5\%$, while it is 143.3% for the setup without readbuffer. The setup with readbuffer obviously suffers less from intra-die variations. Fig. 6 also shows the normal distributions that have been fitted to both full datasets. Notice that the tail of the distribution is not well predicted by such a normal distribution. This has been discussed in more detail in [8]. Also notice that safe operation requires a much higher percentile, e.g., 1 failure in 10^9 rather than 1 failure in 10^3, so the overhead due to intra-die variations will be twice as big as those numbers suggest (about 6σ rather than about 3σ).

Fig. 7 shows the cumulative distribution of the effectively developed swings on the different bit lines at the time when 99.9% of the cells have developed a swing of 150 mV or more on the GBL_H. In the assumption that the associated yield is acceptable, this would be the ideal moment to stop the discharging of

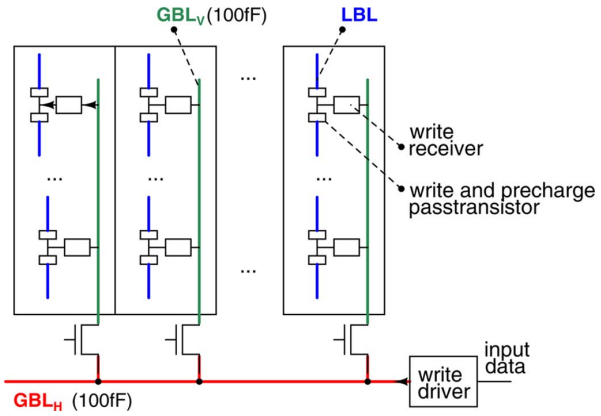


Fig. 8. Bit line hierarchy used for the write operation.

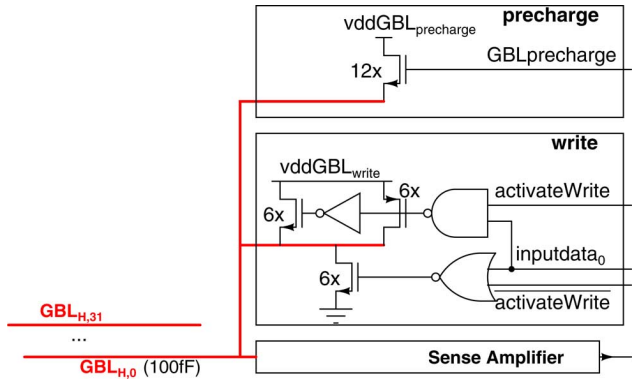


Fig. 9. Precharge and write circuits for the horizontal GBLs. Transistor sizes in multiples of 120 nm.

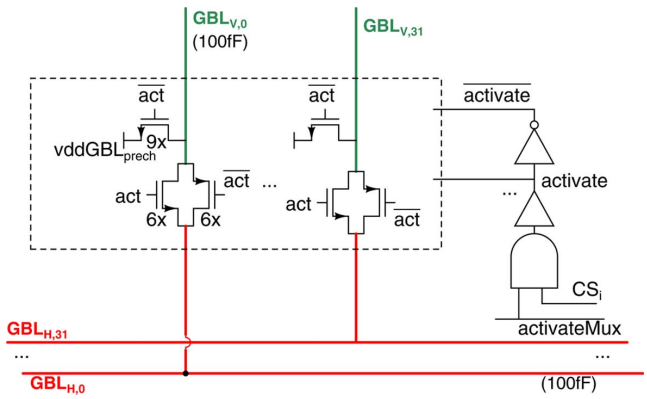


Fig. 10. Control circuitry to precharge the vertical GBLs, and to connect the vertical GBLs from the activated column to the horizontal GBLs. Transistor sizes in multiples of 120 nm.

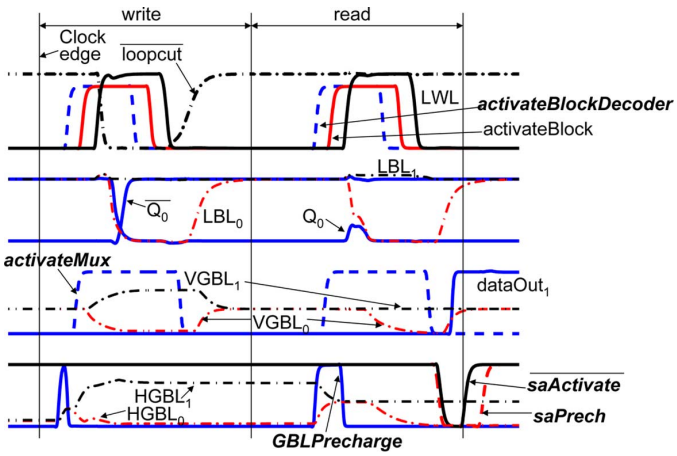


Fig. 11. Simulated timing diagram.

the GBLs. In reality, it is very hard to control this stop time accurately, and GBL discharge will continue for quite a while after this ideal stop time, wasting energy for no good reason.

In the setup without readbuffer, the average discharge of GBL_H and GBL_V is significantly larger than the 150 mV which was actually required for the SA. The average swing on the LBL is almost 450 mV, significantly larger than the swing on the GBLs.

For the setup with readbuffer, the spread on the swing is much smaller, and the main reason for this is simple: because GBL_H and GBL_V were precharged at only 0.2 V, not even the fastest path can generate a swing larger than this 0.2 V. The LBL always makes a full swing transition, as was intended. With this bit line architecture, there is no need to control the end of the GBL discharge period accurately. With idealised timing fixed at the 99.9% delay percentile, energy consumption for both schemes is about equal in this setup. With real timing or in a larger matrix, the advantages of the read buffer setup become more pronounced.

D. Bit Line Hierarchy: Write

Fig. 8 shows the bit line hierarchy used in this design. Shared write receivers are used to enable a reduced voltage swing on the GBLs during write operations, as previously discussed in [6]. The write receivers are triggered by a signal derived from the combination of *activateBlock* and $\mathrm{writeFlag}_i$. In this design, a skewed buffer is used as receiver. However, a sense amplifier as used in [6] or [9] might have been a better choice because it is less sensitive to inter-die variations and does not suffer from static leakage.

E. GBL Implementation Details

Fig. 9 shows the circuit used to precharge and to write the horizontal GBLs. For the write driver, both NMOS and PMOS pull-up driver were provided to enable experimentation with both high and low $\mathrm{vddGBL}_{\mathrm{write}}$. Fig. 10 shows the circuit used to precharge the vertical GBLs and to connect the vertical GBLs from the activated column to the horizontal GBLs. This circuit is repeated in each column footer. The pass transistors and precharge transistors that have to operate at reduced V_{gs} are implemented as LVT transistors.

F. Operation and Timing

Fig. 11 shows the simulated timing diagram of this memory. All critical timing signals are derived from digitally tunable delay lines. Fig. 12 shows the implementation of the tunable delay lines used in this design. The delay is programmed using four configuration bits, enabling 16 possible delay values in steps of about 100 ps. The minimal delay value is 100 ps, the maximal delay value is about 1.7 ns. These delay elements are also used in pulse generators to obtain a tunable pulse width,

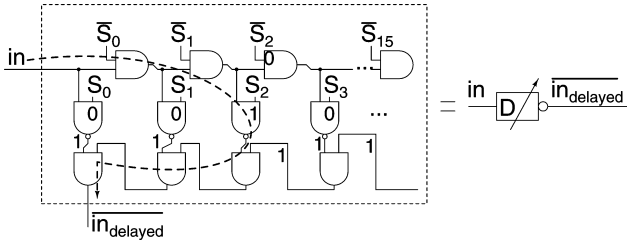


Fig. 12. Inverting tunable delay line.

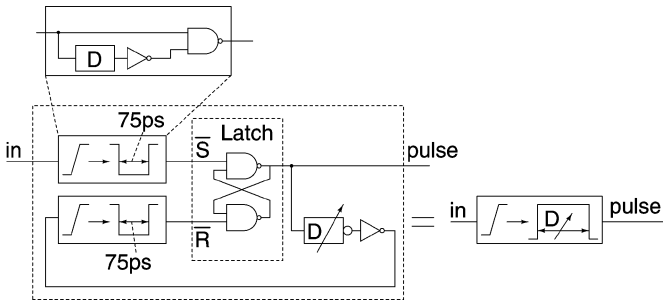


Fig. 13. Pulse generator with tunable width. This implementation can handle input pulses that are shorter than the desired output pulse width.

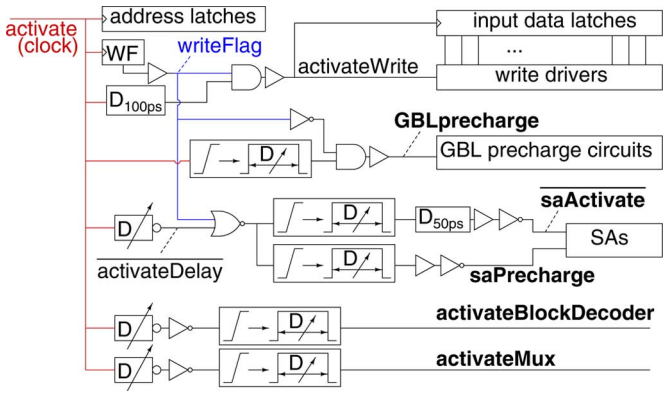
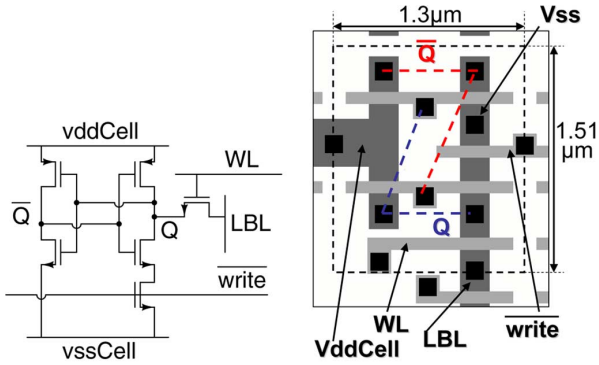


Fig. 14. Generation of critical timing signals.

as shown in Fig. 13. The current implementation of the tunable delay elements is rather energy hungry. The equivalent capacitance is about 12.5 fF per 100 ps segment, which sums up to about 500 fJ at 1 V, which is about 10% of the total memory energy consumption. Given their relaxed constraints, it is easy to conceive low power tunable delay elements for future designs. Current starved approaches such as used in [10] appear promising.

Fig. 14 shows how the different critical timing signals for this memory are generated. Only the rising edge of the activate/clock signal is used. In this design, eight fully tunable delay lines are used to generate five important control signals, displayed in bold in the figure. This large number of tunable delay elements provides maximum timing flexibility and reduced the layout effort. The price payed for this flexibility is energy: the tunable delay system consumes 10% of the total memory energy consumption during read accesses. Future designs can reduce this contribution significantly, either by using better delay elements or by removing the flexibility that is not strictly needed.



(a) (b)

Fig. 15. Single-ended cell with write assist transistor that cuts the pull-down path during write. All transistors are HVT and have $W = 200$ nm and $L = 80$ nm. (a) cell schematic and (b) cell layout.

Tunable timing requires a calibration phase to obtain the optimal settings for the die and operation mode at hand. As the memory always functions correctly in the most relaxed timing setting, this calibration consists of a simple loop in which each timing delay is reduced until the memory stops working reliably. Then, margins are introduced to guard against time varying effects, such as power supply ripple and temperature variations. In the current implementation, this calibration loop is controlled externally.

Tunable timing has many advantages. It avoids the need for replica-based timing, which is complicated due to the low-swing signals on the GBLs. In the context of large, uncorrelated intra-die variations, replica-based timing has another, more fundamental problem. The basic operation principle of replica-based timing is to transform the timing problem from accurately controlling an absolute delay to accurately controlling the difference between two delays. This is great if the delay variations of the two paths are strongly correlated. However, in the case of uncorrelated variations, $\sigma_{\Delta\text{delay}} = \sqrt{2} \cdot \sigma_{\text{delay}}$. In this case, replica-based timing actually makes things worse. In multi-mode memories, tunable timing can apply optimal delay settings to each operation mode of the memory individually, and can adjust this setting to the statistics of the die at hand. Compared to static timing approaches, this reduces both the required design margins and the design risks.

III. Local Block

This section describes the local block of the memory. Paragraph III.C describes improvements that should be considered in future designs.

A. Memory Cell and Local Bit Line Periphery

1) Memory Cell: Fig. 15 shows the cell used in this design. It is a single-ended cell, which uses an additional write assist transistor that cuts the pull-down path during write to overcome the write-1 problem. This cell requires two control signals: WL and $\overline{\text{write}}$. It is possible to share $\overline{\text{write}}$ between multiple words though. In this design, it is shared between the two words that are interleaved on the same row.

2) Local Bitslice: Fig. 16 shows the local bitslice used in the design. Each LBL is split in 4 sub LBLs LBL_A to LBL_D in an

TABLE III
REQUIRED SIZE OF SENSE AMPLIFIER (SA) INPUT TRANSISTORS TO COPE WITH SA OFFSET

node	$A_{\Delta Vt}$ [mv μm] (estimated)	required SA upscaling[1] and SA energy contribution[2] Vdiff[3]=100mV	Vdiff=75mV	Vdiff=50mV	Vdiff=25mV
90nm	4.5	6.8x (10%)	12.0x (18%)	27x (40%)	108x (160%)
65nm	4	9.8x (15%)	17.5x (26%)	39x (58%)	158x (233%)
45nm	2.5 [4]	7.7x (11%)	13.7x (20%)	31x (46%)	123x (182%)
32nm	2.5 [4]	15.6x (23%)	27.8x (41%)	63x (93%)	250x (380%)

[1] required size for the SA input transistors (for 6σ) [multiples of the technology's W_{min}]
[2] SA energy consumption [fraction of consumption of total memory, SAs excluded]
[3] voltage difference between inputs. Single-ended SAs require twice this as BL swing.
[4] very optimistic estimates

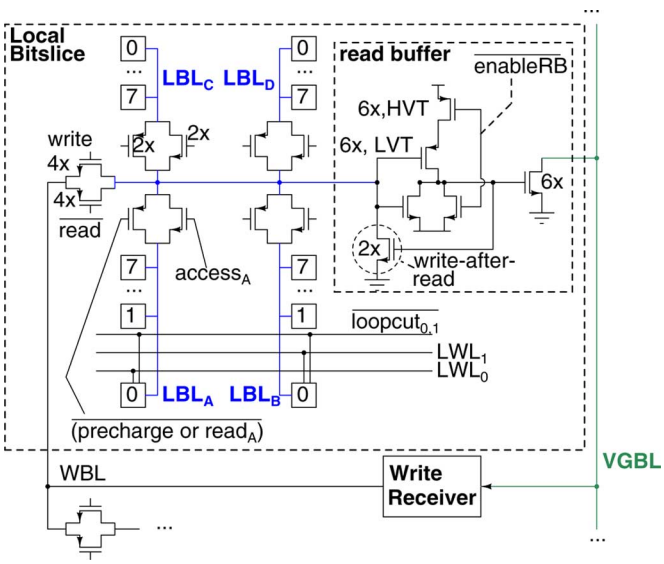


Fig. 16. Local bitslice. All transistor widths expressed in multiples of 120 nm.

attempt to reduce the bit line capacitance. They connect through complementary pass transistors to a central point, which is connected to the readbuffer and to the precharge and write pass transistors which connect to the write bit line (WBL). The WBL is used for both precharge and write operation. Each write receiver is shared between two local blocks. The readbuffer is equipped with a write-after-read transistor to speed up and stabilize the slowest, most unstable memory cells during read operations.

B. Local Block Timing and Control

In this design, the only signal that is used to control the timing of the local block is *activateBlock*, which is a pulsed signal. All other global signals only change when the corresponding input address bits change. The timing order between four events in the local block must be preserved. The precharge must be disabled before the word line is enabled, and the word line should be disabled before the precharge can be enabled again. To ensure this order while starting from a single timing pulse requires a circuit with unequal rise and fall delay along the WL and precharge control paths.

To control the pass gates of the sub LBLs requires knowledge about which sub LBL is accessed. This information is not provided by the global decoder. For each sub LBL, an eight-input dynamic or gate combines the eight relevant WLs to find out whether it is accessed.

C. Future Enhancements

The choice for a single-ended read operation is reasonable as it reduces the average number of full-swing LBL transitions. The single-ended write and the complicated LBL structure are less bright ideas. They are fully functional though. This implementation is a consequence of the fact that we tried to design an embedded DRAM with a single-ended cell in a parallel effort.

Anyway, the current implementation of the local block is not a critical contributor to the strong performance of our design, so other implementations could be considered, both for the memory cell and for the LBL structure. A normal 6 T cell with HVT transistors would result in very similar performance values. Because there is one less transistor in the cell pull-down path, it would even be a bit faster. The LBL structure with sub LBLs is probably too complicated, as it results in rather awkward control circuitry. With this control overhead correctly taken into account, a more straightforward LBL implementation would most likely result in a slightly better memory performance for equal area.

IV. TUNABLE SENSE AMPLIFIERS

This section first explains the impact of uncorrelated stochastic intra-die variations on SA performance and it provides a short overview of previously proposed solutions, especially SA redundancy. After this introduction, SA tuning [7], the novel solution used in our design, is discussed, and a comparison with SA redundancy is made. Finally, a practical calibration algorithm for tuned SAs is provided.

A. The Sense Amplifier Offset Problem

During a read operation, a small cell must develop a signal swing on a BL. Even when a subdivided bit line architecture is employed, the GBL is a huge capacitive load, especially when compared to the driving capability of the cell. If a large swing is required on this BL, both access time and energy consumption of the memory will increase significantly.

A good SA will reduce the required BL swing as much as possible. The ability of a SA to sense small input differences is limited by its input-referred offset. When the designed input swing is reduced, a traditional SA must be made larger and its dynamic

energy consumption increases. Table III contains a lower bound on the required upscaling of the input transistors of a SA to obtain "6σ" yield (a failure rate of about $2 \cdot 10^{-9}$). This table takes only the threshold voltage mismatch of the input transistors into account. According to Pelgrom's law [11], the required size increases as $V_{\text{diff}}{}^{-2}$, where V_{diff} is the available voltage difference between the SA inputs. The required relative size for a fixed V_{diff} increases as geometries shrink because the mismatch between minimal devices increases.

The table also contains an estimate of the energy consumption associated with these SA sizings, relative to the energy consumption of all other memory parts combined. These energy numbers should be compared to the energy consumed in the GBLs, and the optimal combination of GBL swing and SA energy consumption should be selected. In our design, A 100 mV single-ended GBL swing consumes 10% of the total memory read energy. This 100 mV swing results in a V_{diff} of 50 mV. To be able to sense this 50 mV V_{diff}, a SA which adds 40% to the total memory read energy consumption would be needed. These numbers indicate that for traditional SAs, differential input swings below 75 mV are not an energy-efficient choice for this memory size in 90 nm technology.

A technique that can reduce the required input swing with a factor of 2 to 4 for a fixed SA energy consumption would certainly improve the energy-delay performance in today's memories. Larger reductions can prove useful in larger memories, in technologies with larger variations as well as in other applications.

B. Sense Amplifier Calibration

A lower input swing can be obtained when circuit techniques are employed that solve the offset problem without enlarging SA transistors. There are two families of circuit techniques that can be used: offset compensation and calibration.

Offset compensation uses analog feedback to cancel out transistor mismatch at run-time, typically using several different phases during the course of a memory access. Interesting examples of this approach are [12] and [13]. In the operation of some current sense amplifiers, transistor mismatch is implicitly compensated for, as in [14]. The main disadvantage of offset compensation is that all approaches require static currents during the compensation stage.

Calibration on the other hand is performed in a separate calibration phase, for example after fabrication, while booting or whenever temperature has changed by 5 degrees. During this phase, the best setting for each component is obtained and stored. During normal accesses, these settings can be used without much energy and delay overhead and without static current consumption. The next subsection will discuss SA redundancy, a type of SA calibration introduced in [1]. The subsequent section will discuss SA tuning, a novel type of SA calibration used in our design. The two approaches will be compared, and a practical way to calibrate the tunable SAs will be presented.

C. Sense Amplifier Redundancy

In [1], SA redundancy was introduced to improve SA yield for a given SA area. Fig. 17 illustrates the concept. Each SA is replaced with a set of N (≥ 2) SAs. During calibration, the best SA of each set is determined. Afterwards, only the best SAs will be used. The resulting effective SA offset is the minimum of the offsets of the SAs in the set, which results in a much more favorable effective offset distribution. The actual selection of the signal from the best SA to the output must happen at run time. This introduces an overhead on access time and energy. As such, the scheme does not scale well to large values of N.

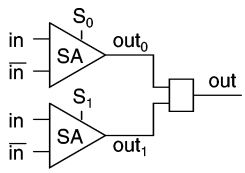


Fig. 17. Setup for SA redundancy ($N = 2$).

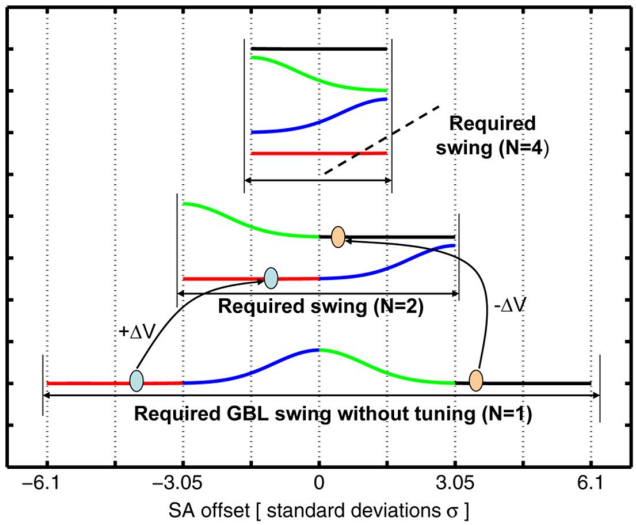


Fig. 18. Effective offset of a tuned sense amplifier.

D. Sense Amplifier Tuning

1) Concept: Our design uses a novel technique: SA tuning. Here, each SA individually receives the most appropriate reference voltage based on its offset. Our design target is not better yield for a given area, but lower energy consumption for a fixed failure rate of 10^{-9}. Area was considered a lesser constraint, mainly because only one set of SAs is needed for the entire memory because of the extended GBLs (see II.C).

Fig. 18 illustrates how such a scheme changes the effective offset distribution for the SA after tuning. The probability density function (PDF) at the bottom is the offset distribution for a SA before calibration has been applied. If we accept a failure rate of 10^{-9}, we must only consider SAs whose native offset lies in the -6.1 to $+6.1$ σ range. Consider a simplified setup: a SA with one input connected to a single-ended GBL and the other input connected to 0 V. Without calibration, the GBL voltage must be at least $+6.1\sigma_{V_{\text{offset}}}$ to ensure it is detected as a high input for all considered SAs. Conversely, the GBL voltage must be below -6.1σ to ensure it is detected as a low input. If we assume the GBL is precharged high and is pulled down conditionally, the minimal required single-ended GBL swing is 12.2σ.

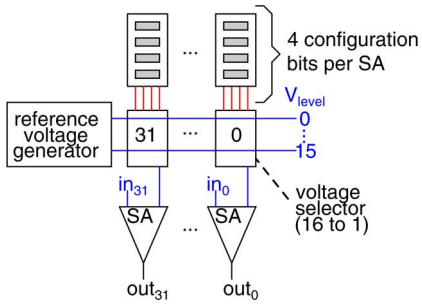


Fig. 19. Setup for SA tuning ($N = 16$; word length $= 32$).

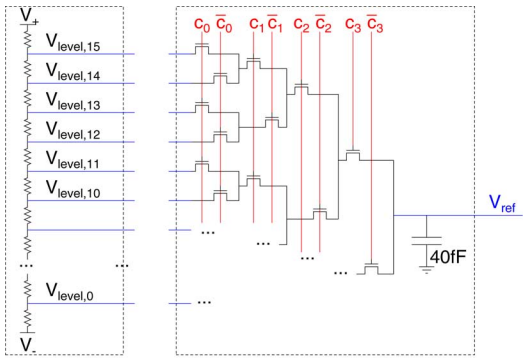


Fig. 20. Prototype implementation of (a) reference voltage generator, and (b) voltage selector.

Now we introduce tuning with two voltage levels ($N = 2$). If a fabricated SA has a negative native offset, the offset is shifted by $+\Delta V$. If the fabricated SA has a positive native offset, the offset is shifted by $-\Delta$ V, with $\Delta V = 3.05\sigma$. The distribution after tuning with $N = 2$ is the sum of the two partial PDFs displayed in the figure. The effective offset range after tuning is only half of the native offset range. In our simplified setup, a single-ended GBL swing of 6.1σ is sufficient. The most straightforward way to shift the offset voltage in the simplified setup is to change the voltage that is applied at the second input from 0 V to $+$ or $-\Delta$ V. The same approach can be used with larger values of N.

Fig. 19 illustrates the setup for tunable SAs with 4 configuration bits ($N = 16$). An on-chip reference voltage generator generates 16 reference voltages $V_{\text{level},0} \ldots V_{\text{level},15}$. Each SA is accompanied by 4 memory elements. The content of these memory elements controls a voltage selector which selects one of the 16 available voltage levels to be used as the reference voltage input for this SA. Tuning does not require any selection in the critical logic path, so it scales gracefully to large values of N.

2) Implementation: In this prototype implementation, the reference voltages are generated on-chip with a simple resistive divider as shown in Fig. 20(a), which consumes 13 μA DC current. In our design, the high DC impedance of the voltage generator does not pose a problem since the SAs that are used do not draw any DC current from their inputs. The voltage selectors are implemented as in Fig. 20(b). Notice the large decoupling capacitance at the output of each voltage selector. This reduces the AC-impedance of the reference voltage tree to an acceptable

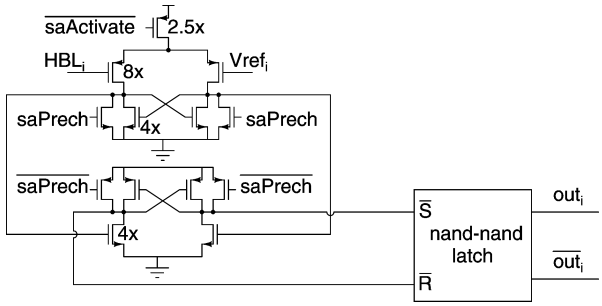


Fig. 21. SA schematic. Transistor widths expressed in multiples of 120 nm. All other transistors are 200 nm wide. All transistors have minimal length ($L_{\text{drawn}} = 80$ nm).

TABLE IV
SA TUNING: PARAMETERS AND SPECS

measured $\sigma_{Voffset}$	19mV
minimal required tuning range (12.2 $\sigma_{Voffset}$)	230mV
tuning range used	300mV
config bits (# settings)	4 (16)
tuning step	~20mV
minimal single-ended input swing that was measured reliably	40mV
total area overhead	<1%

level to avoid excessive kick-back noise from one SA to the reference voltages of other SAs, which would result in data-dependent problems.

Fig. 21 is a schematic of the sense amplifier that is used in this design. Notice that the first stage draws static current as long as $\overline{\text{saActivate}}$ remains low. When the pulse width of $\overline{\text{saActivate}}$ is well controlled, this pulsed-mode SA consumes less energy than the self-stopping SA used in [6].

The offset of the 32 SAs on a single die was measured. The distribution has a standard deviation of $\sigma_{V_{\text{offset}}} = 19$ mV and appears to be Gaussian, as expected. To obtain a failure rate of less than 10^{-9} with these SAs without the use of tuning would require a single-ended input swing of at least $2 \cdot 6.1 \cdot \sigma_{V_{\text{offset}}} = 230$ mV. The novel design with tunable SAs employs a sufficient tuning range of 300 mV and 16 levels, which in theory allows for a single-ended swing of just under 20 mV. However, some margin is needed to guard against other noise sources and to obtain high speed operation. With relaxed timing settings (2 ns memory access), no errors were observed when reading with 40 mV single-ended GBL swing. In normal operation, 100 mV swing is used and the SA reference voltages are chosen as to detect inputs below about 60 mV as low logic inputs and inputs above 80 mV as high inputs. This setting results in a faster access than the symmetric alternative, where the average threshold voltage would be at 50 mV.

The area of the tuning circuitry per SA is rather large. However, as the employed GBL architecture requires only 32 SAs, the total area overhead is less than 1%. This overhead decreases further for larger memories. Table IV summarizes the main properties of the tuning system used.

E. Comparison Between Redundancy and Tuning

Table V compares tuning and redundancy based on the theoretical remaining SA offset for a given yield target as function

TABLE VI
APPLICATION OF THE CALIBRATION ALGORITHM

setting	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	lowest setting that results in '0'	setting to use	Vref used [mV]	GBL toggle voltage with tuning
V_{ref}[mV]	-80	-60	-40	-20	0	20	40	60	80	100	120	140	160	180	200	220				
native SA offset	sensing outcome with GBL @ 100mV																			
19mV	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	9	7	60	79mV
1mV	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	9	7	60	61mV
-95mV	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	14	12	160	65mV
95mV	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	5	3	-20	75mV
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15				

TABLE V
COMPARISON OF THE EFFICIENCY OF SA TUNING AND SA REDUNDANCY AS FUNCTION OF THE NUMBER OF CONFIGURATION BITS

# config bits	theoretical minimum single-ended input swing to overcome intra-die variations (1 failing SA in 10^9 SAs) redundancy	tuning	improvement[1]
0	12.2 $\sigma_{Voffset}$ [100%]	12.2 $\sigma_{Voffset}$ [100%]	1.00x
1	8.32 $\sigma_{Voffset}$ [68%]	6.10 $\sigma_{Voffset}$ [50%]	1.36x
2	5.54 $\sigma_{Voffset}$ [45%]	3.06 $\sigma_{Voffset}$ [25%]	1.81x
3	3.56 $\sigma_{Voffset}$ [29%]	1.52 $\sigma_{Voffset}$ [12%]	2.34x
4	2.18 $\sigma_{Voffset}$ [18%]	0.76 $\sigma_{Voffset}$ [6%]	2.87x

[1] ratio of the required input swing with tuning and with redundancy when the same number of configuration bits is used

of the number of configuration bits ($= \log_2(N)$). Tuning significantly outperforms redundancy on this metric. For example, when 4 configuration bits are used, SA redundancy has a required input swing of $2.18\sigma_{V_{\text{offset}}}$, while SA tuning only needs $0.76\sigma_{V_{\text{offset}}}$, almost a 3x improvement. A main advantage of SA tuning over SA redundancy is that it scales gracefully to higher values of N, as there are no changes needed in the actual access path. The main disadvantages of SA tuning is probably its complexity.

To clarify the advantages, consider the following example. Assume that a reduction of required input swing with a factor of 3x is desired. This can be obtained with SA redundancy with $N = 7$, which requires 3 configuration bits and a 7-to-1 selector in the critical path. With tuning, the same reduction is obtained with $N = 3$, which requires 2 configuration bits. In this case, it would make sense to select $N = 4$ for safety, as the added cost is small. No selector is needed in the critical path.

F. Calibrating the Tunable Sense Amplifiers

In the calibration phase, the best setting for the four calibration bits of each SA must be determined. In our system, the GBLs are precharged to 100 mV. If a cell stores a 1, the associated GBL will remain floating at 100 mV. If the cell stores a 0, the GBL will be pulled down by the readbuffer. As the driver transistor operates in the linear region, the resulting waveform is a decaying exponential that settles only slowly towards 0 mV. In a high performance memory, it is obviously not desirable to have to wait until the signal settles down. Rather, all signals sufficiently below 100 mV should be sensed as low values. The optimal threshold depends on the noise on the floating GBLs and the required level of robustness. In the calibration method that is proposed in the next paragraphs, all GBL values above 80 mV

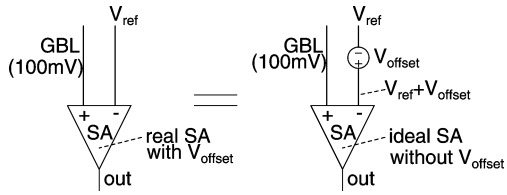


Fig. 22. Setup during the calibration phase.

are treated as high values, everything below 60 mV as low. In this text, the GBL voltage at which the outcome of SA sensing changes from high to low is called the SA's GBL threshold voltage.

During calibration, all GBLs are precharged to 100 mV, as in the normal operation mode. Fig. 22 shows the circuit setup. The reference voltage setting for all SAs is set to 0. All SAs are activated. As a wide enough tuning range is provided, all SAs should return a high SA output: when a very low V_{ref} is applied, 100 mV is considered high, even when a large native offset is present. This step is repeated for each reference voltage setting up to 15. Table VI shows the result of this sweep for four SAs with different native offset.

The lowest reference voltage setting for which the ith SA detects the 100 mV on the GBL as a low value, is now called n_i. When the ith SA is used with voltage reference setting n_i, its GBL threshold voltage will always be somewhere between 100 mV and 120 mV. When voltage reference setting $n_i - 2$ is applied to this ith SA, the GBL threshold voltage is located between 60 mV and 80 mV, as was desired. In the prototype implementation, this calibration loop was controlled off-chip. The calibration bits were set by means of an input shift register.

This algorithm could be implemented as a small FSM. It would require a 4-bit counter to loop over the 16 reference settings. The counter values are copied into the 4-bit calibration registers of those SAs whose output is still 1. After the loop is completed, all calibration bits contain the lowest setting that results in the detection of 0. We still need to subtract 2 from these values, which requires 32 4-bit "subtract 2" units if full parallelism is desired. If so, calibration could be completed in slightly more than 16 clock cycles. In many systems, the calibration could also be delegated to software.

The method described here deviates slightly from what was discussed in [7]. The main advantage of this new method is that it only requires the normal GBL precharge voltage and the reference voltages which are already needed during normal operation.

TABLE VII
MEASURED PERFORMANCE

Measurement results [1]																
Maximum frequency [MHz]	850				780				480				240			
Main/Wordline/Cell Vdd [V]	1.2 / 1.2 / 1.2				1.0 / 1.2 /1.0				0.8 / 1.0 / 0.9				0.6 / 1.0 / 0.9			
	read		**write**		**read**		**write**		**read**		**write**		**read**		**write**	
active energy / access [fJ]	6200	100%	8935	100%	3880	100%	5833	100%	2815	100%	3686	100%	1965	100%	2074	100%
Global decoder + GBL control	1140	18%	2160	24%	610	16%	1105	19%	376	13%	684	19%	204	10%	369	18%
Sense amplifiers	869	14%	0	0%	583	15%	0	0%	403	14%	0	0%	309	16%	0	0%
cells	84	1%	208	2%	0	0%	153	3%	0	0%	75	2%	0	0%	41	2%
WL and write WL (writeBar)	84	1%	150	2%	84	2%	150	3%	70	2%	125	3%	70	4%	125	6%
local bit lines, readbuffer, write bit lines and receivers	750	12%	1638	18%	415	11%	1163	20%	262	9%	614	17%	165	8%	254	12%
global bitlines (100mV swing)	516	8%	1176	13%	340	9%	990	17%	264	9%	772	21%	381	19%	522	25%
local control and decoder	879	14%	2373	27%	583	15%	1438	25%	368	13%	776	21%	197	10%	416	20%
timing (includes delay elements and control wires)[2]	1878	30%	1230	14%	1265	33%	835	14%	1072	38%	640	17%	639	33%	348	17%
	static				**static**				**static**				**static**			
static energy / access [fJ] [3]	1322	100%			801	100%			457	100%			674	100%		
Global decoder + GBL control [4]	236	18%			82	10%			-16	-3%			-145	-22%		
Sense amplifiers	4	0%			4	1%			2	0%			2	0%		
cells	330	25%			96	12%			102	22%			205	30%		
WL and write WL (writeBar)	414	31%			454	57%			255	56%			511	76%		
local bit lines, readbuffer, write bit lines and receivers	76	6%			45	6%			38	8%			37	5%		
global bitlines	3	0%			4	0%			5	1%			8	1%		
local control and decoder	254	19%			114	14%			69	15%			55	8%		
timing (includes delay elements and control wires)	5	0%			2	0%			2	0%			2	0%		
	read		**write**		**read**		**write**		**read**		**write**		**read**		**write**	
Total energy/access [fJ]	7522		10257		4681		6634		3272		4143		2638		2748	
	average				**average**				**average**				**average**			
average[5] energy/access [fJ]	8433				5332				3562				2675			
leakage power [uW][6]	1123				625				219				162			

[1] At room temperature. All energy values calculated as Q*VddMain, except for WL and cells.
[2] includes the tunable delay elements (~500fF during write, 550fF during read), but also the associated control wires and controlled transistor gates: activateBlockDecoder (125fF), activateMux(125fF), GBLPrecharge(100fF), saActivate (150fF) and saPrecharge (200fF). Capacitances were estimated by integrating the currents in back-annotated simulations
[3] The static energy per access is calculated by integrating the leakage power over one clock period at the maximum frequency
[4] There is significant leakage from the WL supply to the global decoder supply
[5] 1/3 write, 2/3 read
[6] this leakage power is already included in total and average energy/access

V. SELECTIVE VOLTAGE SCALING

Reducing the supply voltage helps to reduce both dynamic energy consumption and leakage power. Therefore, it makes a lot of sense to reduce the supply voltage when delay slack is available. However, reducing the supply voltage of all circuits at once is not the best solution if one aims at the lowest possible energy consumption at reasonable speeds.

In our design, $\mathrm{vdd_{cell}}$ and $\mathrm{vdd_{WL}}$ combined consume less than 5% of the total dynamic energy consumption (see Table VII). Reducing these two supplies would only have a limited effect on the dynamic energy consumption (less than 5%). However, as the cells are implemented with the highest available threshold voltage to reduce leakage in the most effective way, reducing these supplies (during operation) would be detrimental for the memory speed. Below a certain voltage, cell functionality is endangered as well. The other circuits, such as the decoder and the SAs are typically implemented with lower threshold voltages, with larger transistors and without use of ratioed logic. Reducing only the supply voltages for these circuits, not for the memory cells, results in a smaller delay penalty and less robustness issues. Because of this reduced impact, lower supply voltages can be used. From this example, it is clear that the best results are obtained when one is selective in which voltages to reduce. Selective voltage scaling has been applied before in memories, most often to avoid stability and reliability issues and always under different names. In [4] for example, the technique is labeled as a dual supply SRAM array.

Notice that the arguments against reduced cell voltage apply only to active cells. Cells that are not accessed have significantly larger operation margins, and reducing their cell supply can be a very effective way to reduce leakage power [15].

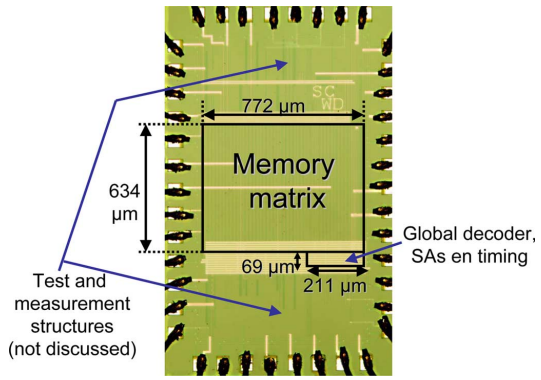


Fig. 23. Die micrograph of the 90 nm 128 kb SRAM.

Our design uses separate supplies for cells and WLs. These voltages are reduced much less aggressively than the main power supply. A standby mode was not implemented.

VI. PROTOTYPE SRAM: MEASUREMENT RESULTS AND COMPARISON

A. Measurement Results

Fig. 23 shows the micrograph of the 128 kb SRAM. Table VII shows the measured speed and energy consumption of the memory. Even though the WL and cell voltage are not scaled down, their energy consumption remains small. The limited currents make it feasible to generate these supplies with an on-chip converter in a future design. The memory covers a wide performance range, consuming between 8.4 pJ/access at 850 MHz and 2.7 pJ/access at 240 MHz.

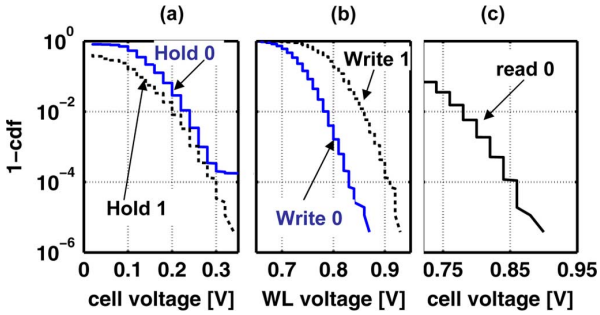


Fig. 24. Measured cumulative distribution of cell operation margins for all 128 K cells on a die. (a) Fraction of cells that fail to hold data at reduced $V_{\rm cell}$ [$V_{\rm WL} = 0$ V] (b) Fraction of cells that can not be written correctly at reduced $V_{\rm WL}$[$V_{\rm cell} = 0.9$ V, $V_{\rm LBL} = 0.6$ V]. (c) fraction of cells that are not read correctly at reduced $V_{\rm cell}$ [$V_{\rm WL} = 1.2$ V, $V_{\rm LBL} = 1.0$ V].

Fig. 24 indicates that the cells operate reliably under intra-die variations, although the limited write margins suggest the use of a double-ended write for designs in future technology nodes. Such a change would have little impact on the reported performance.

B. Comparison

Our design significantly outperforms previous 90 nm designs such as the 64 kb, 16 bits/word memory that consumes 12.9 pJ/access at 833 MHz [5]. The measured total energy per access is lower than that of state-of-the-art 65 nm sub-threshold designs such as [16], which reaches a minimal energy point of 10 pJ/access at 0.4 V. At this voltage, operation speed is about 300 KHz. As our SRAM was not designed for such low operation speeds (leakage would be devastating), such a direct comparison is not entirely fair. The lower energy/access obtained with our design however does indicate that targeting the lowest supply voltage is not necessarily the best road to ultra low power memories.

In Section IV.E of this paper, we have shown that SA tuning provides significant advantages compared to SA redundancy [1] when the target is to enable a small GBL swing or to reduce SA energy consumption. We did not compare the required area, as this is less important in our memory architecture which requires only 32 SAs in total.

VII. Conclusion

Variability-aware circuit techniques allow to cope with stochastic intra-die variations without introducing large margins. As an added benefit, they reduce design-time risks. The use of a buffered local bit line and extended global bitlines, digitally tunable sense amplifiers, tunable timing circuitry and selective voltage scaling resulted in an extremely low energy per operation single-cycle 32 bit/word, 128 kb SRAM design that was fabricated in 90 nm CMOS. Measurements show that in the 850 MHz boost mode, energy consumption is only 8.4 pJ/access. In the normal 480 MHz mode, the energy consumption reduces to 3.6 pJ/access to bottom out at a very aggressive 2.7 pJ/access in the 240 MHz low power mode. In future technology nodes, even larger intra-die variations are encountered. This will most likely further increase the improvements that can be obtained with variability-aware circuit techniques such as those employed in this design.

References

[1] N. Verma and A. P. Chandrakasan, "A 256 kb 65 nm 8 T subthreshold SRAM employing sense-amplifier redundancy," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 141–149, Jan. 2008.

[2] B. Zhai, D. Blaauw, D. Sylvester, and S. Hanson, "A sub-200 mV 6 T SRAM in 0.13 m CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2007, pp. 332–333.

[3] Y. Wang *et al.*, "A 1.1 GHz 12 μA/Mb-leakage SRAM design in 65 nm ultra-low-power CMOS technology with integrated leakage reduction for mobile applications," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 172–179, Jan. 2008.

[4] J. Pille, C. Adams, T. Christensen, S. Cottier, S. Ehrenreich, T. Kono, D. Nelson, O. Takahashi, S. Tokito, O. Torreiter, O. Wagner, and D. Wendel, "Implementation of the CELL broadband engine in a 65 nm SOI technology featuring dual-supply SRAM arrays supporting 6 GHz at 1.3 V," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2007, pp. 322–323.

[5] K. Takeda, Y. Hagihara, Y. Aimoto, M. Nomura, Y. Nakazawa, T. Ishii, and H. Kobatake, "A read-static-noise-margin-free SRAM cell for low-VDD and high-speed applications," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 113–121, Jan. 2006.

[6] S. Cosemans, W. Dehaene, and F. Catthoor, "A low power embedded SRAM for wireless applications," *IEEE J. Solid-State Circuits*, vol. 42, no. 7, pp. 1607–1617, July 2007.

[7] S. Cosemans, W. Dehaene, and F. Catthoor, "A 3.6 pJ/access 480 MHz, 128 kb on-chip SRAM with 850 MHz boost mode in 90 nm CMOS with tunable sense amplifiers to cope with variability," in *Proc. European Solid-State Circuits Conf. (ESSCIRC)*, Sep. 2008, pp. 278–281.

[8] T. Doorn, J. Ter Maten, J. Croon, A. Di Bucchianico, and O. Wittich, "Importance sampling Monte Carlo simulations for accurate estimation of SRAM yield," in *Proc. European Solid-State Circuits Conf. (ESSCIRC)*, Sep. 2008, pp. 230–233.

[9] B. D. Jang and L. S. Kim, "A low-power SRAM using hierarchical bit line and local sense amplifiers," *IEEE J. Solid-State Circuits*, vol. 40, no. 6, pp. 1366–1376, Jun. 2005.

[10] G. Kim, M. K. Kim, B. S. Chang, and W. Kim, "A low-voltage, low-power CMOS delay element," *IEEE J. Solid-State Circuits*, vol. 31, no. 7, pp. 966–971, Jul. 1996.

[11] J. A. Croon, M. Rosmeulen, S. Decoutere, W. Sansen, and H. E. Maes, "An easy-to-use mismatch model for the MOS transistor," *IEEE J. Solid-State Circuits*, vol. 37, no. 8, pp. 1056–1064, Aug. 2002.

[12] Y. Watanabe, N. Nakamura, and S. Watanabe, "Offset compensating bit-line sensing scheme for high density DRAM's," *IEEE J. Solid-State Circuits*, vol. 29, no. 1, pp. 9–13, Jan. 1994.

[13] N. Verma and A. P. Chandrakasan, "A high-density 45 nm SRAM using small-signal non-strobed regenerative sensing," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2008, pp. 380–381.

[14] E. Seevinck, P. J. van Beers, and H. Ontrop, "Current-mode techniques for high-speed VLSI circuits with application to current sense amplifier for CMOS SRAMs," *IEEE J. Solid-State Circuits*, vol. 26, no. 4, pp. 525–536, Apr. 1991.

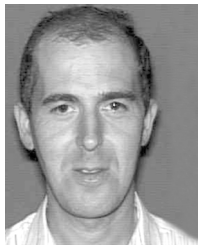
[15] P. Geens and W. Dehaene, "A dual port dual width 90 nm SRAM with guaranteed data retention at minimal standby supply voltage," in *Proc. European Solid-State Circuits Conf. (ESSCIRC)*, Sep. 2008, pp. 290–293.

[16] M. Sinangil, N. Verma, and A. Chandrakasan, "A reconfigurable 65 nm SRAM achieving voltage scalability from 0.25–1.2 V and performance scalability from 20 kHz–200 MHz," in *Proc. European Solid-State Circuits Conf. (ESSCIRC)*, Sep. 2008, pp. 282–285.

Stefan Cosemans (S'04) was born in Mol, Belgium, in 1981. He received the M.Sc. degree in electrical engineering from the Katholieke Universiteit Leuven (K.U. Leuven), Heverlee, Belgium, in 2004. Currently, he is a research assistant at the ESAT-MICAS Laboratory of the Katholieke Universiteit Leuven. He is working towards the Ph.D. degree on the variability-aware design of low-power embedded memories.

Wim Dehaene (S'88–M'97–SM'04) was born in Nijmegen, The Netherlands, in 1967. He received the M.Sc. degree in electrical and mechanical engineering in 1991 and the Ph.D. degree in 1996 from the Katholieke Universiteit Leuven, Belgium. His thesis was entitled "CMOS integrated circuits for analog signal processing in hard disk systems".

After receiving the M.Sc. degree, he was a research assistant at the ESAT-MICAS Laboratory of the Katholieke Universiteit Leuven. His research involved the design of novel CMOS building blocks for hard disk systems. The research was first sponsored by the IWONL (Belgian Institute for Science and Research in Industry and agriculture) and later by the IWT (the Flemish institute for Scientific Research in the Industry). In November 1996, he joined Alcatel Microelectronics, Belgium. There he was a Senior Project Leader for the feasibility, design, and development of mixed-mode systems-on-chip. The application domains were telephony, xDSL, and high-speed wireless LAN. In July 2002, he joined the staff of the ESAT-MICAS Laboratory of the Katholieke Universiteit Leuven, where he is now a Professor. His research domain is circuit level design of digital circuits. The current focus is on ultralow-power signal processing and memories in advanced CMOS technologies. Part of this research is performed in cooperation with IMEC, Belgium, where he is also a part-time Principal Scientist. He teaches several classes on digital circuit and system design.



Francky Catthoor (S'86–M'87–SM'98–F'05) received the engineering degree and the Ph.D. degree in electrical engineering from the Katholieke Universiteit Leuven, Belgium, in 1982 and 1987, respectively.

Between 1987 and 2000, he headed several research domains in the area of high-level and system synthesis techniques and architectural methodologies, including related application and deep-submicron technology aspects, all at the Inter-university Micro-Electronics Center (IMEC), Heverlee, Belgium. Currently he is an IMEC Fellow. He is also a part-time full Professor in the Electrical Engineering Department of the K.U. Leuven.

Dr. Catthoor received the Young Scientist Award from the Marconi International Fellowship Council in 1986. He has been an associate editor for several IEEE and ACM journals, including the *Transactions on VLSI Signal Processing*, *Transactions on Multimedia*, and *ACM TODAES*. He has been the program chair of several conferences including ISSS'97 and SIPS'01.