A Low Leakage 500 MHz 2T Embedded Dynamic Memory With Integrated Semi-transparent Refresh

A. Vignon^{a,*}, S. Cosemans^a, K. Maex^b, W. Dehaene^a

^aESAT - MICAS Laboratory, K.U. Leuven, Kasteelpark Arenberg 10, 3001 Heverlee, Belgium

^bGeassocieerde Afdeling ESAT - INSYS, Integrated Systems, Kapeldreef 75, 3001 Heverlee, Belgium

Abstract

This paper presents a low-leakage 128 kbit dynamic memory based on a 2T dynamic cell. The design is implemented in a logic 90 nm technology and achieves a low static power consumption of 130 μ W and an access time of 2 ns. It has a worst case retention time of 175 μ s. This performance is achieved by introducing an optimized hierarchical organization and peripheral circuits for the read, the write and the refresh operations. A novel writing mechanism for 2T cells using a double phase approach is demonstrated. The area penalty of using short read bitlines is alleviated using a charge transfer sense amplifier (SA). A novel local write sense amplifier (WSA) that can operate as a latch makes it possible to perform the refresh operation at the local level, improving the energy efficiency of the refresh operation.

The memory includes an integrated automatic refresh mechanism. Most read and write operations can still be performed during refresh cycles. In cases where the accessed address conflicts with the refresh operation, the memory handles access recovery internally.

Keywords: 2T, embedded memory, Low power memory design, embedded DRAM, SRAM

1. Introduction

The emergence of new applications for high speed and low power devices requires more and more versatility from the memory design. Most of these applications are enabled by using memories, that are not optimized only for a speed vs density trade-off, but also for a minimum static power consumption. This is due to the restrained power budget in these cases. Reaching lower power consumption for similar speed becomes a requirement in these designs. This type of low power driven design can be found in many memory implementations ([1, 2, 3, 4]). In this context, DRAM benefits from a renewed interest, partly due to the emergence of 3D interconnects, which allows for a reduced power consumption when using a standalone DRAM ([5, 6]). High speed

*Corresponding author, tel: 0032 16 62 11 69, fax: 0032 16 32 19 75.

Email address: anselme.vignon@esat.kuleuven.be (A. Vignon)

Figure 1: Cell matrix organization. The cells sharing the same bitlines are organized in localblocks. These localblocks are splitted in 8 columns.

embedded DRAM (eDRAM) macros have been demonstrated by [?]. In both cases, it has been shown that with recent technology improvement it was possible to nudge the DRAM design space to higher speed and lower power consumption, at the price of a reduced density, compared to conventional DRAM macros.

It has been shown in [?] that DRAM memories can lead to lower static energy than SRAM. This is because DRAM cell leakage is significantly smaller than that of an SRAM cell. The challenge is to obtain a low power consumption for the refresh cycles that are needed when using DRAM.

The goal of this work is to design a DRAM memory consuming less energy than an SRAM at the same moderate speed (500 MHz). This is achieved by, on one hand, reaching a similar dynamic energy as an SRAM matrix, while on the other hand designing a power efficient refresh mechanism. This allows the power consumed by the DRAM refresh cycles to be smaller than the power consumed by the static leakage from the equivalent SRAM cells.

This work presents an implementation of such low static power matrix. We used a 2T capacitive cell inspired by [?], where it was proposed for high bandwidth applications. While both designs use a similar cell, this work is an alternative to moderate speed low power SRAM. The aim of this work is to 2T cell, in order to implement peripherals, that reach an improved static power consumption compared to an equivalent SRAM. This differs from [?], which focuses on offering a higher bandwidth than DRAM with an improved density compared to SRAM. Such difference in design goal leads to a different global architecture (section 2), bitline structure (section 3) and behavior (section 4,5), as well as refresh operation (section 6).

The most significant improvement obtained is the static power consumption reduction. To achieve this result, a significant area of the memory is taken by the peripheral circuitry in this architecture. The area penalty of using additional peripheral circuitry is

Figure 2: Decoder partitioning for the 128kbit 2T DRAM prototype.

balanced by the smaller footprint of a 2T cell, compared with an SRAM cell. It will be shown in section 7 that the resulting memory matrix is still more dense than the equivalent SRAM. This moderately high speed (2 ns access time), 90 nm 128 kbits memory matrix [?] was implemented and features a semi transparent refresh, 2T dynamic cells showing a 175 μ s retention time, as well as an improvement in static energy consumption, by a factor of 2 compared to an SRAM. In low power mode, further static power reduction is achieved.

2. Global Architecture

This memory matrix uses 2T DRAM cells, that can be used as eDRAM. The cell can be written (Fig. 3) by driving the cell WWL at 1.2 V. In this case, the storage node is driven by the WBL voltage. When not accessed, the cell storage node is left floating. The cell WBL is driven low, at -0.2 V. Both $\overline{RBL}$ and $\overline{RWL}$ are kept high, at 1.2 V. To read the cell, $\overline{RWL}$ is driven low, while $\overline{RBL}$ is left floating.

In order to achieve a low dynamic energy as well as a low access time, a fine granularity matrix was used [?]. The address space consists of 4096 words of 32 bits, as described in Fig. 1, for a total of 128 kbits. It is divided in 8 columns. Each column is one word wide. In the horizontal direction, the column vertical bitlines are muxed to 32 global sense amplifiers (SA), that are shared by the entire memory.

A column is divided in 8 localblocks of 64 words that share the same write sense amplifiers (WSA) (8 localblocks per column). Each of these localblocks is divided in 4 blocks of 16 words that use the same local read sense amplifiers (RSA).

To reduce dynamic energy, the address decoding is organized as shown in Fig. 2: Three different decoders translate the address into coordinates for, respectively, the column, the localblock row, and the wordline inside the localblock, that is being accessed (*WIB*). These decoders are all static. While the column address is used to enable the corresponding mux, the row information is combined with the wordline, in order to activate statically the correct global wordline *GWL*.

A synchronisation block provides the timing for activating every dynamic signal. The *activateCol* signal, generated from the static column decoding, is only used to select which column is accessed, and to conclude an access cycle. The static row signal is used to generate a dynamic *activateRWL* pulse, which activate the read-wordline buffer during an access, as described in sections 4 and 5. It is also used to activate the correct SA for the operation, by enabling *activateRowRead* and the RSA in case of a read access, or *activateWriteBlock* and the WSA in case of a write access. This allows to activate only the needed SA during either of these operations, while sending only one signal per access to the localblock.

Table 1: Area comparison between different embedded memories. LR means that logic design rule were followed. LO means that the cell was optimized for lithography. The local relative size is the relative size of the peripherals in the matrix. The glob. relative size represents the relative size of the rest of the peripherals.

	[?]	[?]	This Work	[?]
Tech.	90 nm	65 nm	90 nm	65 nm
Node	LR	LO	LR	LO
Mem. type	SRAM	eDRAM	2T	
Mem. size	128 kbits	2 Mbits	128 kbits	
Cell area	1.96 μm^2	0.13 μm^2	0.79 μm^2	
Tot. area	0.64 mm ²	0.67 mm ²	0.53 mm ²	
Rel. size	Cells:40%	Cells:38%	Cells:20%	Cells:30%
	Local:57%	Local:19%	Local:48%	Perifs:70%
	Glob.:3%	(BL only) Glob.:57%	Glob.:32%	

An important specification of an architecture, is its area efficiency. This efficiency is however difficult to assess. First of all an architecture implementing short bitlines typically includes several additional peripherals, thus degrading the overall density. However, short bitline memories allow to reach a better trade off in speed and power consumption, which is required for low power applications. In addition, most memories used in industry are typically optimized for lithography. This allows to implement memory cells, that are more dense than by following logic design rules. Such litho-optimized cells are not available to the academic community. It is complex to compare cells, that don't follow the same layout rules, as any commercial implementation would use litho-optimized cells.

A comparison of the relative contribution of peripherals in an architecture is given in Table 1, for several memories implementing short bitlines. Clearly this memory doesn't compete on density with other DRAM architectures. That makes sense, as the area gain obtained by using DRAM is used to implement additional peripherals.

Figure 3: Typical read and write operations for an NMOS 2T cell. During write, the access transistor is conducting, and the data is transmitted from the WBL to the storage node. During read, the storage node voltage determines if the RBL is discharged or not by $\overline{RWL}$.

However, the total memory density is still higher, than for the equivalent SRAM.

3. 2T memory cell, standard operation

The 2T cell (Fig. 3) consists of two transistors, connected by four terminals: the write wordline WWL , the write bitline WBL , the read wordline $\overline{RWL}$, and the read bitline $\overline{RBL}$. The access transistor is used for writing operations. The gate of the read transistor is used as a capacitance for storing the data. This read transistor is also typically used for reading, by probing it at the drain and source to check if the transistor is not conducting.

In this architecture (Fig. 4(a)), WBL is connected to a buffered output of the WSA. $\overline{RBL}$ is connected to the RSA. Both WWL and $\overline{RWL}$ timing are local control signals, controlled by the static GWL , to choose which word is being accessed.

2T cells have been proposed in both NMOS [?] and PMOS [?] implementations. These 2T cells are different from [?], as such cell does not store data in a transistor floating body, but in a transistor gate. This allows to make such cell compatible with any NMOS technology, without requiring finFETs or SOI technology. In this work, a 2T NMOS cell was used, as it offers an higher retention time, for the technology used here. This advantage is due to an improved readability when the stored charge is degraded.

There are two main advantages for a 2T dynamic memory (Fig. 3), when compared with a typical DRAM. First, the read access is non-destructive. During a read access, the node storing the data is not discharged and does not require a write after read operation. A second critical advantage, is that the 2T cell can be implemented using only two transistors. This means that a 2T cell is compatible with any technology already implementing CMOS transistors, particularly CMOS logic-only technologies. This means that, compared with other eDRAM technologies, the 2T cell doesn't require any additional process steps. This design was implemented using a 90nm logic-only technology. On the other hand, the 2T cell is typically less dense than a deep trench-based DRAM cell. Also, as the storage capacitance is formed by a gate capacitance, the charge stored by a cell is limited. This typically translates into a reduced retention time, when comparing a typical 2T cell with a DRAM cell.

The static power consumed by a dynamic energy cell depends on two factors: the retention time, which determines how often a cell's data needs to be refreshed, and

the dynamic energy spent performing this refresh operation. Therefore, the usually reduced retention time of a 2T cell translates into a higher static energy per bit. That is especially true when the refresh operation is not optimised for power consumption. As this work focuses on reducing the memory static power consumption, the architecture presented here aims at mitigating this retention time issue. This was done by optimising both the writing (section 4) and the reading (section 5) operations for a higher retention time, as well as improving the dynamic energy for the refresh operation (section 6).

4. Write bitline operation

The proposed bitline (Fig. 4(a)) was designed in order to maximise the 2T cell retention time (Fig. 5), to reduce the passive power. The retention time of a cell depends not only on the leakage of the access transistor, but also on the charge that can be stored during write. When writing '1', the cell is first charged to a voltage value $V_{written}$ (Figure 7), during either a write or a refresh operation. This charge is the maximum charge stored in the cell. Over time, the charge leaks away, down to the minimum charge, that can be resolved by the sense amplifier. This minimum charge corresponds to a minimum required voltage $V_{minread}$, to discharge the BL, ensuring a correct read operation. The retention time corresponds to the duration between these two events. The retention time for storing the can be written as:

$$t_{retention} = \frac{C_{cell}}{I_{cell}} (V_{written} - V_{minread}) \quad (1)$$

Where C_{cell} is the storage capacitance of the cell, and I_{cell} represents its leakage current. Improving the voltage written in the cell is therefore important to improve the retention time.

During a typical write operation (Fig. 3) for a dynamic memory, the access transistor is conducting and the WBL is charged with the data to be written. Because the 2T cell is implemented using NMOS transistors, the cell storage node is fully discharged if a "0" is written, but the voltage stored is limited by the threshold voltage (V_T) of the cell access transistor, when the written data is "1". This issue exists on most dynamic memories. It is usually solved by increasing the voltage on the access transistor gate during a write operation. Such solution is however less efficient for a memory such as a 2T cell, that must be fully compatible with a logic process. That is because the voltage range allowed at the gate of a logic transistor is limited, due to reliability issues. Another solution for this problem was developed, which doesn't require high voltage driving, nor an additional voltage source. This technique is called double phase writing.

In the first phase of a write access (Fig. 6), a regular write is performed, with WWL at 1.2 V and WBL at the value to write. If the value written is "1", WBL is at V_{dd} , and the voltage stored increases up to a value depending on the access transistor V_T . During this phase, $\overline{RWL}$ is driven low and $\overline{RBL}$ is floating. During the second phase, $\overline{RWL}$ is driven at 1.2 V while the access transistor remains conducting. Because of the drain-gate and source-gate capacitances of the read transistor, this results in a higher voltage (Fig. 7) into the storage node. If the data written is '0', the WBL is at 0 V, and

(b) Simulated read '0' (left) and read '1' (right).

(c) Simulated write '0' (left) and write '1' (right) access.

Figure 4: (a) Schematic of a bitline, with read sense amplifier and write sense amplifier (WSA). Simulated read (b) and write (c) access waveforms.

Figure 5: Measured cumulative density distribution of the retention time for a single cell.

Figure 6: Double phase writing for a 2T cell.

the resulting charge increase on the storage node is absorbed via the conducting access transistor. If WBL is at V_{dd} , on the other hand, the storage node voltage increases, therefore increasing ΔV , when WBL is disabled with RWL kept at 1.2 V, at the end of the cycle.

Fig. 7 presents the simulated spread of both voltage values written on the cell, and cell retention time. Such values are simulated for single, and double phase writing. It is clear, that the double phase writing improves the minimum written voltage. This results in a 400 ms simulated retention time. Such high retention time is obtained over 1000 test samples. A more accurate retention time, measured across multiple dies, is given in section 7. Because the double phase writing depends on the drain-gate capacitance of the read transistor, the written voltage value depends also on the threshold voltage of the read transistor. This explains the increased spread of the double phase writing voltage, compared to the single phase written voltage. In this implementation, the memory was designed, so that the written voltage in any conditions, would be equal to at least the full logical level (1 V). In this architecture, the read bitline is very short. This makes the additional energy consumed on RBL during write negligible, when compared to the energy spent on the global bitlines.

The WBL is connected to the WSA (Fig. 4(a)). During write, the WSA acts as a differential SA, and sense the data from the global write bitlines. During refresh, however, the WSA acts as a latch. This latch is transparent during the first part of the refresh. During this period, the WSA stores the data read from the RSA. The refresh continues afterwards with a double phase writing of the data, that is stored in the WSA.

Figure 7: Simulated retention time and voltage stored on a 2T cell written with either typical or double phase writing. Simulation over 1000 samples.

This allows to write back the data read from the cell, without using the global bitlines, as will be explained in section 6.

5. Read bitline operation

Reading a 2T cell (Fig. 3) is easier than for a typical DRAM matrix. Typically, the $\overline{RWL}$ is discharged during a read operation, with the $\overline{RBL}$ having been precharged high. If the cell stores a “1”, the read transistor of the cell is conducting, and $\overline{RBL}$ is discharged by $\overline{RWL}$. After a read delay the voltage level in $\overline{RBL}$ is sensed, and the data in the cell is resolved by the read sense amplifier (RSA). Such reading operation is not destructive. Restoring the data on the cell after a read operation is therefore not needed. Furthermore, it has the advantage to be only time dependent: the voltage drop obtained on the read bitline is not limited by the charge of the cell. On a typical DRAM, the storage node is discharged in the bitline. This discharge is responsible for creating a ΔV , that is being sensed afterwards to read the cell data. A consequence of this, is that the voltage drop sensed is limited by the charge on the cell. A 2T DRAM, on the other hand, allows to have a bitline discharge of more than the charge of the cell. The voltage drop is therefore only limited by the delay spent reading.

However, several limitations constrain this read operation. First of all, The voltage on the cell is not always at a high voltage, even when a “1” has been written. Its read current is therefore relatively low. During retention time, the charge stored is indeed reduced by the cell leakage. Therefore the read transistor is not fully conducting. In addition, it is not possible to use a full swing bitline to resolve the data stored in the cell. That is because, if the voltage in $\overline{RBL}$ is low enough, it is possible to have a leakage between the $\overline{RWL}$ of the cells sharing the same $\overline{RBL}$, if these neighbours read transistor is also conducting.

The small read current requires using a short bitline, to reduce the matrix latency. On the other hand, because a full swing bitline cannot be used, a relatively accurate RSA is also needed. When the bitline is short, the RSA behavior becomes critical for

Figure 8: Estimated ratio of cycles lost due to conflicts between the memory accesses and a refresh operation. In approach 1, the refresh blocks the entire memory, and a single word is refreshed per refresh cycle. In approach 2 and 3, the refresh is semi-transparent: only accesses to addresses on the localblock that is being refreshed are blocked.

both the density and the static power figures of the memory. This is why a novel type of RSA was used, called a charge transfer SA. The charge transfer sense amplifier [?] allows for a reduced area penalty when using short read bitlines.

As explained before, a reading (Fig. 4(c)) or a refreshing access starts by pulling $\overline{RWL}$ low. If the cell stores “0”, $\overline{RBL}$ remains high. If the cell stores “1”, however, $\overline{RBL}$ is discharged by $\overline{RWL}$. The resulting voltage difference in $\overline{RBL}$ is resolved by the RSA, by lowering the V_{bias} voltage from 1.2 V to a lower reference voltage V_{ref} . If at that time $\overline{RBL}$ is still fully charged, the charge at $\overline{RBL}$ will transfer to the drain of the sensing transistor, causing $WRBL$ to be discharged fully. If, on the other hand, $\overline{RBL}$ is at a voltage less than approximately $V_{Tp} + V_{ref}$, where V_{Tp} is the sensing transistor V_T , no charge transfer occurs. $WRBL$ remains charged.

If the memory is being read, a $WRBL$ discharge causes the global bitline $\overline{VRBL}$ to be charged. In case of refreshing, $\overline{VRBL}$ is gated, while $WRBL$ is used as input to the WSA.

Due to the other cells connected to the $\overline{RBL}$, with their $\overline{RWL}$ at 1.2 V, the $\overline{RBL}$ discharge must be limited, to avoid a short circuit current between the cells and its neighbours, as described in [?]. In this case, using the read sense amplifier described above allows for choosing a reference voltage. This is an advantage over the single transistor sense amplifier, where the reference voltage is determined by the sensing transistor V_T . The charge transfer sense amplifier is, on the other hand, more compact than a differential sense amplifier using a voltage reference.

6. Refresh operation

In any dynamic memory matrix at least the refreshed bitline becomes inaccessible during a refresh cycle. For this reason, it would be complex and costly to build a dynamic memory, that is fully accessible during refresh. On the other hand, having

to handle failed or delayed accesses generates a cost at the system level. This cost increases with the probability of having to deal with such abnormal accesses.

A second important parameter for refreshing is the active energy consumed during the cycle. The static energy consumed by the memory is directly proportional to the active energy consumed per cell refreshed [?]. Therefore, special care was taken to reduce the energy consumption during refresh.

Because reading a 2T cell is not destructive, the read operation does not include a write after read operation, as is usually the case for DRAM matrices. For this reason, the refresh operation of a 2T cell consists of two distinct operations, first read the cell value at local level, then re-write the read value into the cell. This is made possible by storing the data locally, using the WSA.

By connecting the two sensing nodes of the WSA (Fig. 4(a)) to the ground, the WSA is effectively transformed into a transparent latch during refresh. During a first phase of refresh, the data of the cell is sensed by the RSA. The resulting voltage is then stored by the WSA. This voltage can then be restored into the cell, by performing a local write using the data stored in the WSA. The main advantage of such refresh operation, is that the global bitlines are not used during a refresh. This not only saves on dynamic energy, but also allows for a concurrent access of the memory to be performed in parallel with the refresh operation.

In this work, the refresh accesses are scheduled internally from an external refresh clock, that can be asynchronous with the access clock. Every access is accepted, but there is a possibility for the access to be delayed in case of refresh conflict, in which case an *accessDelayed* flag is raised until the access is processed. To achieve this result, both refresh and access controls are generated by two different timing circuits, handled by a synchronization circuit, that arbitrates between access, refresh, or both cycles.

The synchronization circuit gives initially the priority to the access over refresh cycles: A refresh access cannot be started if an access is running. The refresh cycle is however immediately started at the end of the running access, or when received, if no access is running at that time. If an access is started during the refresh cycle, its compatibility with the refresh cycle is verified while its address is decoded. If the access is not compatible, the timing circuit for accessing the memory is blocked, which effectively blocks the access. This synchronisation topology presents two main advantages. The access cycle has the priority, and the concurrency check is not on the memory critical path. There is therefore no latency penalty for the unblocked accesses. In addition, this type of circuit alleviates the synchronisation issues between refresh and activation accesses.

This refresh mechanism blocks accesses to every address from the local block of the word being refreshed, while the rest of the memory is still accessible. It was done in order to reduce the probability of having to block an access during refresh, as explained in Fig. 8.

A semi-transparent refresh was demonstrated in [?], by implementing registers to distribute the refresh accesses, and allowing concurrent accesses to a different bank than the one that is being refreshed. This work uses the same type of modified decoder to control the refresh operation. Two differences are, however, to be noticed: First of all, the proposed architecture allows a concurrent access even to the refreshed bank, and only block access to the refreshed local bitlines. This improves the memory availability,

Figure 9: Measured number of bits in the memory that do not achieve the retention time.

as the number of unaccessible words is reduced. In addition, while [?] implements pipelined accesses, this work allows a concurrent refresh and access on a single cycle latency memory.

To reduce the energy consumed by refreshing, but also to increase the matrix availability during a refresh cycle, multiple refresh accesses are performed during the same cycle. During each refresh cycle, one address for every column gets refreshed (Fig. 8).

Since every column is accessed at once during refresh, every column is always selected for refresh. The selection of the next *GWL* and localblock row to refresh is performed at the end of the refresh cycle. It reduces the refresh access time, as the refreshed address selection is not in the critical path. Furthermore, it avoids synchronization issues between refresh and access, as the next address to be refreshed is available before the refresh signal is received. To choose which *GWL* should be refreshed, a counter is implemented in every header of every localblock. This counter iterates over the *GWL* of the localblock row, each time the row is selected for refresh. The selection of the localblock row is handled at global level. This is because any access addressing the same row as the one being refreshed should be delayed during a refresh cycle. A global binary counter chooses which row should be refreshed.

7. Measurement results

The proposed architecture was implemented in a 90 nm, logic low power multi-threshold technology. The figures presented here were measured at room temperature. A performance summary can be found in Table 2. The worst case retention time, as measured on five different dies is 175 μ s (Fig. 9). This allows for a really aggressive static energy figure, of 130 μ W in low power (LP) mode.

The retention time is measured as the minimal period between either cycles (read, write or refresh), when every cell is functional in a given die (Fig. 9). The lowest static energy figure of 130 μ W is obtained by reducing the global decoder and timing control of the memory to the supply of the global write bitlines (0.75 V). In this mode, the memory is still functional, with a clock frequency of 250 MHz.

Table 2: Performance summary. The measured static power includes the static leakage, plus the power spent refreshing the memory.

Memory size	128 kbits
Word size	32 bits
Technology node	90 nm
Cell area	$0.79 \mu\text{m}^2$
Access time	2 ns
Static power	$130 \mu\text{W}$
Dynamic read energy	3.7 pJ/word
Dynamic write energy	5.2 pJ/word
Dynamic refresh energy	20.4 pJ/16 words

Figure 10: Static power consumption in high speed (HS) and low power (LP) modes.

The static energy partitioning is shown in Fig. 10. For this architecture, the power due to the periodic refresh cycle is not the dominant energy cost.

We compared the measured results with two other published memories in Table ??: the first one is an implementation of a faster memory in the same technology, that also aims at reducing static power consumption [?]. The second is another 2T implementation, aimed at reaching higher density. In both cases, this work shows a lower static power.

The density figure is higher than the one for the equivalent SRAM. However, the bit density is significantly reduced, when compared with the other 2T cells. As explained in section 2, it is difficult to draw significant conclusions from this figure, as several factors come into explaining this difference. The minimal feature size of [?] is lower, and part of the difference can be explained by the area increase between a cell layout optimized for lithography, and a cell following logic based layout rules. We did not have access to lithography optimized cell for this layout. Yet it is clear that, other things being equal, high granularity DRAM matrix have a lower density than more standard layout, as this type of architecture is focused on energy rather than on density. For the same energy/speed trade off, however, the density comparison with [?] is to the advantage of this work. This make sense, as we are using the cell density improvement brought by the eDRAM to implement more efficient peripherals, rather than directly as a global density improvement. Despite this fact, the proposed memory still offers an improved density compared with [?]. To conclude, the dynamic power for this implementation is similar, when compared with other SRAM memories of the same

speed.

Table 3: Comparison between this work, an SRAM with similar latency and active energy, and the reference 2TDRAM [?]. The static energy figures compared here are the best available at 500 MHz.

	This work	[?]	[?]
Technology	90 nm	90 nm	65 nm
Size	128 kbits	128 kbits	2 Mbits
Min. cycle time	2 ns	1.2 ns	2 ns
Ret. time	175 μ s	<i>N.A.</i>	10 μ s
Static power	173 μ W	457 μ W	0.768 W
Static power per bit	1.35 nW/bit	3.57 nW/bit	384 nW/bit
Area per bit	4.04 μ m ²	4.9 μ m ²	1.09 μ m ²

8. Conclusion

We demonstrated a dynamic memory, that shows lower static power than the equivalent SRAM matrix, for the same size and technological node. The access time is slightly higher but comparable. This shows, that it is possible to use dynamic cells for moderately high speed low power memories.

Several techniques were designed to accommodate the use of 2T cells in this context. Shorter read bitlines and double phase writing were used to bring the retention time to 175 μ s.

While the area needed for the peripherals allowing such performance makes this matrix less dense than typical eDRAM macros, it should be noted that the density is still higher than its SRAM counterpart. It has been shown that the refresh can be handled internally by the memory, requiring only a dedicated refresh clock from the system. Moreover it has been proven, that the occurrence of having an access delayed can be as low as 6.5% even in the worst case of a permanently active memory. This probability is less than 1% for more realistic activity scenarios.

Figure 11: Chip microphotograph.

It has been demonstrated that it is possible to reduce static power consumption for medium speed, low power memories, by using capacitance based memory cells. An implementation of such memory has been described, that shows a lower static power, when compared to an equivalent implementation using SRAM cells. A static leakage of 1.35 nW/bit for a 2 ns access time is achieved by this implementation.

- [] Cosemans, S., Dehaene, W., Catthoor, F.. 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. *IEEE Journal of Solid-State Circuits* 2009;44(7):2065–2077.
- [] Verma, N., Chandrakasan, A.. A 65nm 8T Sub-Vt SRAM employing sense-amplifier redundancy. *IEEE International Solid-State Circuits Conference, Digest of Technical Papers* 11-15 Feb. 2007;:328–606.
- [] Jain, S., Agarwal, P.. A low leakage and SNM free SRAM cell design in deep sub micron CMOS technology. *VLSI Design, 19th International Conference on* 3-7 Jan. 2006;:4.
- [] Noguchi, , et al. A 10T non-precharge two-port SRAM for 74% power reduction in video processing. *VLSI, IEEE Computer Society Annual Symposium on* 9-11 March 2007;:107–112.
- [] Rooseleer, B., Cosemans, S., Dehaene, W.. A 65 nm, 850 MHz, 256 kbit, 4.3 pJ/access, ultra low leakage power memory using dynamic cell stability and a dual swing data link. In: *Solid-State Circuits Conference, European*. 2011, p. 519–522.
- [] Kawano, M.. A 3D packaging technology for high-density stacked DRAM. *VLSI Technology, International Symposium on* 23-25 April 2007;:1–2.
- [] Facchini, M., Marchal, P., Catthoor, F., Dehaene, W.. An RDL-configurable 3D memory tier to replace on-chip SRAM. In: *DATE*. 2010, p. 291–294.
- [] Facchini, M., Carlson, T., Vignon, A., Palkovic, M., Catthoor, F., Dehaene, W., et al. System-level power/performance evaluation of 3D stacked DRAMs for mobile applications. In: *DATE Conference Exhibition*. 2009, p. 923–928.
- [] Barth, J., et al. A 500 MHz random cycle, 1.5 ns latency, SOI embedded DRAM macro featuring a Three-Transistor micro sense amplifier. *IEEE Journal of Solid-State Circuits* 2008;43(1):86–95.
- [] Vignon, A., Cosemans, S., Dehaene, W., Marchal, P., Facchini, M.. A novel DRAM architecture as a low leakage alternative for SRAM caches in a 3D interconnect context. In: *Design, Automation Test in Europe Conference Exhibition*. 2009, p. 929–933.
- [] Somasekhar, D., et al. 2GHz 2Mb 2T gain-cell memory macro with 128GB/s bandwidth in a 65nm logic process. In: *Solid-State Circuits Conference, Digest of Technical Papers. IEEE International*. 2008, p. 274–613.

- [] Vignon, A., Cosemans, S., Dehaene, W.. A low leakage 500MHz 2T embedded dynamic memory with integrated semi-transparent refresh. In: Solid-State Circuits Conference, European. 2011, p. 523–526.
- [] Ekanayake, V.N., Manohar, R.. Asynchronous DRAM design and synthesis. In: Ninth International Symposium on Asynchronous Circuits and Systems, 2003. Proceedings. IEEE; 2003, p. 174– 183.
- [] Fossum, J., Lu, Z., Zhang, W., Trivedi, V., Mathew, L., Sadd, M.. A novel two-transistor floating-body memory cell. In: SOI Conference, 2007 IEEE International. 2007, p. 105 –106.
- [] Heller, L., Spampinato, D., Yao, Y.. High sensitivity charge-transfer sense amplifier. Solid-State Circuits, IEEE Journal of 1976;11(5):596–601.
- [] Kirihata, T., Parries, P., Hanson, D.R., Kim, H., Golz, J., Fredeman, G., et al. An 800-MHz embedded DRAM with a concurrent refresh mode. IEEE Journal of Solid-State Circuits 2005;40(6):1377– 1387.
- [] Cosemans, S., Dehaene, W., Catthoor, F.. A low power embedded SRAM for wireless applications. Solid-State Circuits Conference, European Sept. 2006;;291–294.