Ultra-Low Power High Stability 8T SRAM for Application in Object Tracking System

POORAN SINGH, (Student Member, IEEE),
AND SANTOSH KUMAR VISHVAKARMA, (Member, IEEE)
IIT Indore, Indore 452017, India

Corresponding author: Pooran Singh (bishtpooranstar@gmail.com)

ABSTRACT In this paper, an ultra-low power (ULP) 8T static random access memory (SRAM) is proposed. The proposed SRAM shows better results as compared with conventional SRAMs in terms of leakage power, write static noise margin, write-ability, read margin, and I_{ON}/I_{OFF}. It is observed that the leakage power is reduced to $82\times$ (times) and $75\times$ as compared with the conventional 6T SRAM and read decoupled (RD)-8T SRAM, respectively, at 300 mV VDD. In addition, write static noise margin (WSNM), write trip point (WTP), read dynamic noise margin, and I_{ON}/I_{OFF} ratio are also improved by 7.1%, 43%, 7.4%, and $74\times$ than conventional 6T SRAM, respectively, at 0.3 V VDD. Moreover, the WSNM, WTP, and I_{ON}/I_{OFF} values are improved by 6.67%, 7.14%, and $68\times$ as compared with RD-8T SRAM, respectively, at 0.3 V VDD. Furthermore, a fast, reliable, less memory usage object tracking algorithm and implementation of its memory block using ULP 8T SRAM are proposed. A quadtree-based approach is employed to diminish the bounding box and to reduce the computations for fast and low power object tracking. This, in turn, minimizes the complexity of the algorithm and reduces the memory requirement for tracking. The proposed object detection and tracking method are based on macroblock resizing, which demonstrates an accuracy rate of 96.5%. In addition, the average total power consumption for object detection and tracking which includes writing, read and hold power is $1.63\times$ and $1.45\times$ lesser than C6T and RD8T SRAM at 0.3 V VDD.

INDEX TERMS SRAM, leakage power, static noise margin (SNM), object detection and tracking and macroblock resizing.

I. INTRODUCTION

The object detection and tracking device basically uses two types of architectures, one is the proximity sensor and the other one is processing logic with memory system. The processing speed of tracking device can be improved by well-structured algorithm. However, to control the standby power and overall system power, a subthreshold metal oxide semiconductor (MOS) architecture/hardware is required. Considering the fact that tracking system requires a large amount of memory to store information, a high capacity static random access memory (SRAM) cell based cache memory is essential. However, the SRAM is known as a power hungry device due to its high bit-line capacitances. In addition, the SRAM has tendency to fail at subthreshold voltages and shows vulnerability at various process voltage temperature (PVT) conditions. Thus, various subthreshold SRAM cells have been proposed in literature to achieve less standby power with high stability [1]–[4]. The cell stability is considered as one of the major concerns in SRAM cell architectures. Nevertheless, the read static noise margin (RSNM) can be

improved using read decoupled logic [5], which further improves the yield or accuracy of a SRAM cell. Although, it is observed that RSNM is vulnerable at subthreshold regions while reading through full swing sense amplifier (SA) [6]–[8]. Accordingly, researchers have proposed SRAM architectures to work in subthreshold region with better read-write stability [9]–[15]; such as 8T SRAM cell that employ reverse short channel effect to increase write-ability at lower voltages [7], [12]; a 8T SRAM for variability tolerance and low voltage operation in high-performance caches [15]. However, reducing leakage power is still a challenging task for memory devices used in communication networks and wireless sensing devices.

Due to high accuracy and fast speed the SRAM has been used in the signal processing and communications systems. However, the systems designed for object detection and tracking consumes a large amount of static power. The detection and tracking system requires a large amount of memory to store random moving object information and the stored information is used to compare the successive frames to

observe any change in motion. However, the object detection and tracking of moving objects itself is considered as a stimulating task to achieve in real time environment [16]–[19]. Enhancement in methods projected for object detection and tracking from an image or video is the primary concern globally [20]. There are many notable algorithms exist in the literature intended for detection and tracking of an object in the image/video [21], [22]. The commonly used approach for object detection and tracking using monocular camera is to employ sequential information computed from an order of frames to reduce forged detections. This information is typically drawn in the form of difference between the successive frames, which highlights varying regions in frames. Object tracking is an interesting and complex problem to perform in real time. Tracking normally requires the position and silhouette of the object in every frame. Complications in tracking objects can arise due to various reasons such as hasty change in object motion, varying intensity of light and occlusion. Additionally, the change in appearance of the object may occur due to object-to-object occlusion, object-to-scene occlusion, and camera motion. Object tracking is an important aspect in the field of computer vision. There are three major steps in object tracking: detection of object, tracking it from frame to frame, and analyzing the object motion to identify its actions [23]. Substantial amount of work has been done in the past to improve the accuracy and speed of object tracking [21], [24]–[33]. It is noted from literature that the main focus of researchers was on improving the accuracy of object tracking in real-time scenarios, which introduce extra hardware, specifically, static random access memory (SRAM) architectures. Accordingly, in this work, fast, reliable, less memory demanding object detection and tracking algorithm is proposed. The goal is achieved in four steps, namely segmentation & thresholding, object detection, quadtree method to minimize pixelation and tracking.

Furthermore, the proposed 8T SRAM in this paper overcomes the standby power issue with better cell stability, which improves the total power consumption required for object detection and tracking. The rest of the paper is organized as follows: Proposed 8T SRAM cell with 32×64 array is presented in section II. Operation of proposed cell is explained in section III. The simulation results and discussion of proposed 8T SRAM are mentioned in section IV. The summary of experimental results for 8T SRAM 2-kb array is explained in section V. Further, the proposed quadtree based approach for object detection and tracking is detailed in section VI. Section VII demonstrates the simulation results and comparisons of proposed tracking algorithm. Lastly, section VIII concludes our proposed work.

II. POSITIVE FEEDBACK CONTROLLED (PFC) 8T SRAM FOR OBJECT TRACKING

Object tracking requires an ultra-low power memory block to store the information of present and reference macroblock in the form of pixels. Each grey pixel has 8-bit information. Thus, an 8 bit array of memory is required for storing
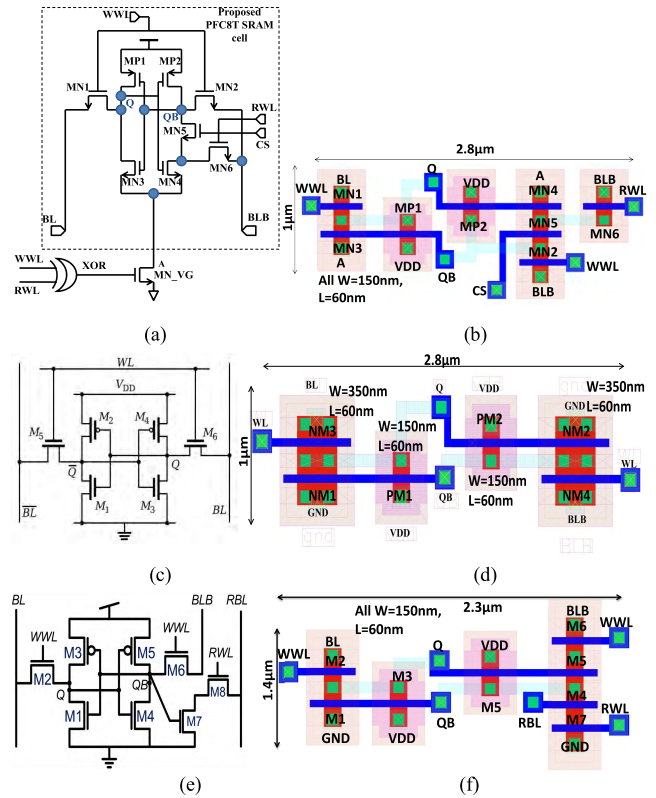

FIGURE 1. (a) Architecture of proposed 8T SRAM cell. (b) Layout of proposed 8T SRAM cell at 65nm standard technology, where all W/L ratios are same W=150nm and L=60nm. (c) Schematic of C6T SRAM cell. (d) Layout of C6T SRAM cell at 65nm standard technology. (e) Schematic of read decoupled 8T SRAM cell. (f) Layout of RD8T SRAM cell at 65nm standard technology.

one pixel. The memory array is designed using SRAM cells due to its fast write and read access time. However, there are some limitations of subthreshold SRAM based memory architecture, specifically, high leakage power, low static noise margins, disparity in stability and susceptibility due to different PVT conditions.

Consequently, a read decoupled positive feedback controlled 8T (PFC8T) SRAM cell is proposed to resolve the ambiguous behavior of subthreshold SRAM architectures at variable PVT conditions in 65-nm standard CMOS technology. The schematic of proposed cell and its layout are shown in Fig. 1(a) and Fig. 1(b), respectively. The schematic and cell layout of conventional 6T and read decoupled (RD)-8T SRAM cell are shown in Fig. 1(c), Fig. 1(d), Fig. 1(e) and Fig. 1(f), respectively. All the metal oxide semiconductor (MOS) transistors in proposed and existing cells are taken as low voltage threshold (LVT) transistor. However, the proposed 8T cell has 1.4× the layout area as compared to the standard C6T SRAM cell, but the improvement in its leakage power and cell stability at different PVT conditions makes it a better choice for portable memories.

A. CELL ARCHITECTURE

The proposed cell has two write access n-MOS transistors, MN1 and MN2. An input bit is written to the SRAM

cell through these transistors. The *BL* contains single bit information to write and at the same time *BLB* contains the complementary of that. However, n-MOS transistors MN4 and MN6 are used as a read decoupled logic to read information from the cell. The *BLB* is precharged to VDD before the read operation is performed. The MP1-MP2-MN3-MN4-MN5 transistors form a latch, where, MP1 and MP2 are the pull-up p-MOS transistors linked to VDD. In addition, MN3 and MN4 are the pull down transistors connected with MN_VG, where MN_VG is an n-MOS transistor connected to ground terminal, which eventually used as a stacking transistor to improve the leakage power in hold operation. The MN_VG is controlled by external *XOR* gate which is shared among the whole row of the SRAM array. Moreover, MN5 is the feedback cutting transistors that is used to disconnect the path between VDD to GND. This transistor is activated by control signal *CS*, which helps to improve, read static noise margins and leakage power of the cell. The operations at different states are shown in Table 1.

TABLE 1. Truth table for various operations in proposed PFC8T SRAM.

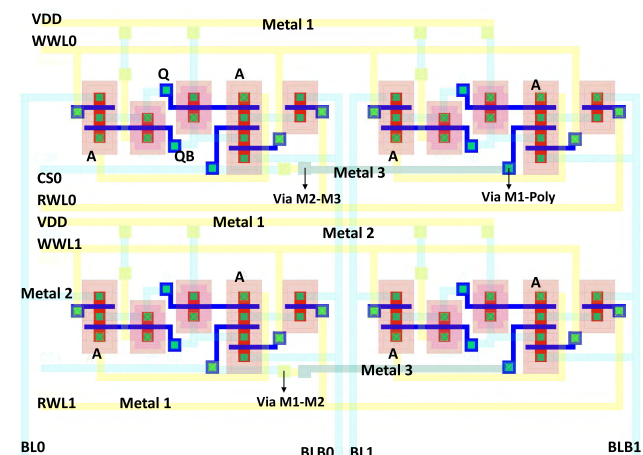| Operation	WWL	RWL	BL	BLB	CS	XOR I/P
Write 1	1	0	1	0	1	1
Write 0	1	0	0	1	1	1
Read 1	0	1	1	1	0	1
Read 0	0	1	1	1	0	1
Hold 1	0	0	0	0	1	0
Hold 0	0	0	0	0	1	0

FIGURE 2. Proposed 8T SRAM cell based 2x2 array layout, where W=150nm and L=60nm. The *RWL, WWL, CS* and VDD shared by each row, while *BL* and *BLB* are shared by each column.

B. SRAM ARRAY LAYOUT

For a better understating of input/output ports and MOS transistors, a 2×2 array layout is shown in Fig. 2. The layout shows the linkage between various input and output ports connected using different metal layers. In proposed cell architecture, three metal layers are used, namely M1, M2 and M3. The design of array is made as compact as possible. These metal layers are connected with each other and poly

through 'via', such as Poly-M1, M1-M2 and M2-M3. There are separate *WWL* and *RWL* for each rows of array to control write and read operation, respectively. The control signal (*CS*) used to make 8T SRAM decoupled during read operation is shared with every column of array. The *BL* and *BLB* signals are also shared by each column of array.

However, the bit-line capacitance associated with the read and write operations for SRAM layout depends upon the number of bits linked with each column. In our case, the bitline in read condition share two transistors MN2 and MN6. Due to two pass transistors coupled with the read path, the bitline capacitance is increased. Though, the capacitance is higher than that of conventional 6T SRAM and RD8T SRAM, the leakage power and static noise margin are improved. In addition, the cell layout area of RD8T is similar to our proposed 8T SRAM. To make simulation results more vigorous, we considered C6T SRAM in iso-area with proposed SRAM cell by making pass transistors' width to 350nm at 65nm standard CMOS Technology.

C. 8T SRAM 32×64 MACRO

The architecture of 8T SRAM Macro is shown in Fig. 3. The Macro consists of 32 rows and 64 columns. The proposed Macro has two row world-line controllers; *WWL* and *RWL* embodied for write and read operations, respectively. The XOR gate is shared with each row of SRAM Macro to offer an input to MN_VG transistor. In addition, the stacking transistor MN_VG is also linked with each row which eventually assists to reduce the leakage power.
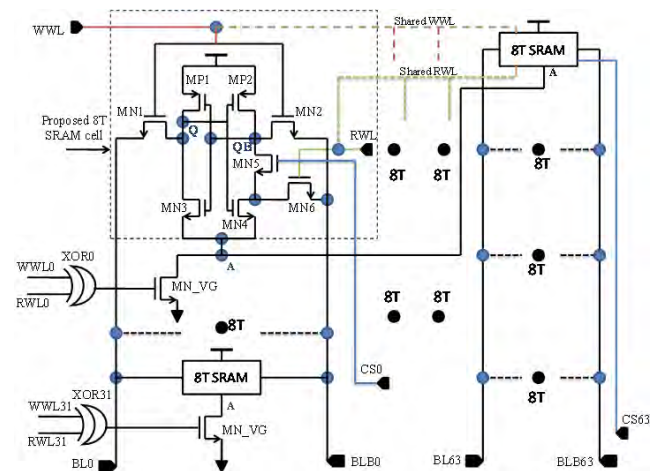

FIGURE 3. Architecture of 8T SRAM cell based 32×64 bit macroblock. The *WWL* and *RWL* are world lines provided for writing and reading the information, respectively. The control signal '*CS*' is generated from controller unit. The MN_VG is the NMOS transistor used for stacking purpose and shared to each row to reduce leakage power associated with bitlines. The input to the MN_VG is controlled by the external XOR gate.

III. OPERATION OF 8T SRAM

This section elaborates the various operations that take place in proposed 8T SRAM and further discusses the leakage power, read-write static noise margin, and behavior of SRAM at different PVT values.

A. READ OPERATION

Before reading information, *BL* and *BLB* are precharged to VDD. The read operation is obtained through ultra-fast current mode sense amplifier [34]. Read operation is performed by keeping read word line (*RWL*) HIGH, write word line (*WWL*) LOW and the control signal (*CS*) and *XOR_I/P* kept at LOW and HIGH, respectively, as shown in Fig. 4(a). Subsequently, for read '1' (Q=1), logic 1 is stored at storage node *Q* and *RWL* is kept HIGH, which eventually turns ON MN4 and MN7. This forms a discharging path across BLB-MN6-MN4-MN_VG and a voltage difference, $\Delta V_{BLB} = \{VDD\text{-}[VDD\text{-}I_{read} \times R_{MN\_VG-MN4-MN6}]\}$ appears between *BL* and *BLB*, which is sensed by the full swing inverter sense amplifier; where, I_{read} is the cell current and $R_{MN\_VG-MN4-MN6}$ is the resistance through MN_VG, MN4 and MN6. The read time is measured as the time the *RWL* is HIGH until the *BLB* is discharged to the required voltage needed by SA to read.

B. WRITE OPERATION

For write operation, *WWL* is kept HIGH and the *RWL* LOW. To write '1,' the control signals *CS* and *XOR_I/P* are kept HIGH. The logic 1 is written to storage node *Q* through *BL-MN1-Q*. The write '1' time is measured as the time when *WWL* signal is HIGH and storage node *Q* reaches to 90% of VDD. Similarly, write '0' time is measured as the time when *WWL* signal is HIGH and storage node *Q* reaches to 10% of VDD. The write '1' operation is shown in Fig. 4(b).
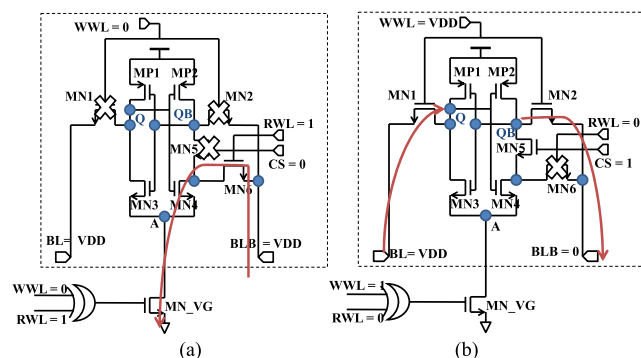

FIGURE 4. (a) Read operation schematic shows discharging of BLB potential from VDD to VDD-V$_T$. Where V$_T$ is the minimum voltage required by SA to sense and read information from cell. (b) Write '1' operation schematic shows charging and discharging of storage nodes 'Q' and 'QB'.

C. STANDBY POWER ESTIMATION

Generally, the memory cell remains in static or hold state for most of time. Therefore, there would be very high possibility of increase in leakage power in SRAM cell at different process-voltage-temperature (PVT) values. In proposed PFC8T SRAM cell, the control signal *XOR_I/P* turns OFF MN_VG which helps to reduce the leakage current by disconnecting the path of the latch from VDD to GND. The leakage power in C6T SRAM cell introduces due to non-availability of virtual ground transistor such as MN_VG. In proposed 8T SRAM cell, MN3/MN4 and MN_VG transistors forms a

stack, where one node of the stack is linked to VDD and other end is connected to ground. In hold mode, the tail transistor (MN_VG) is turned OFF as output of the XOR-gate is low (due to both *WWL* and *RWL* are low). Therefore, the cross-coupled inverters are decoupled from the ground and a stack is formed between MN3/MN4 and MN_VG. Because of this effect, the intermediate node A rises to some positive voltage. This positive voltage (V$_A$) reduces leakage and hold power during standby mode.

D. READ STATIC NOISE MARGIN (RSNM)

A key figure of merit for an SRAM cell is its read static noise margin (RSNM). It can be extracted by plotting the largest possible square in the two voltage transfer curves (VTC) of the involved CMOS inverters [35]. The RSNM is defined as the length of side of the square, given in volts. When an external DC noise is larger than the RSNM, the state of the SRAM cell can change and data is lost. In read '0' operation, node *Q* and *QB* are at logic 0 and logic 1 state, respectively. In conventional 6T SRAM, if a DC noise is added at *Q* or *QB*, it flips the state of opposite storage node and causes a reduction in RSNM value. Moreover, in proposed PFC8T SRAM read operation, *CS* is LOW, which eventually turns OFF MN5, which disconnects the path from *QB* to GND. This, subsequently, improves the RSNM value. Consequently, *QB* will remain at logic 1 value, besides, a positive noise added at *Q*. However, in conventional 6T SRAM cell, if a positive noise, ΔV_{noise} is added at $Q=0+\Delta V_{noise}$ and reaches to the threshold voltage of opposite n-MOS transistor, which turns it ON and makes a discharging path from *QB* to GND, which would flips the state of *QB* from logic 1 to logic 0 and degrades the RSNM value.

IV. SIMULATION RESULTS OF PFC8T SRAM

The proposed 32×64 bit SRAM array is simulated using standard 65-nm CMOS technology. The post layout simulations in iso-area condition is carried out to determine various constraints like leakage current, power, read-write delay and power, power delay product (PDP), read static noise margin (RSNM), write static noise margin (WSNM), dynamic noise margin (DNM), half select issues, and write trip point (WTP). Further, all the constraints are observed at different temperature values ranging from 0°C to 100°C and at different process corners namely fast-fast (FF), slow-slow (SS), typical-typical (TT), slow-fast (SF) and fast-slow (FS) using Monte Carlo (MC) simulations at standard 6σ (sigma) variations.

A. LEAKAGE CURRENT AND POWER CONSUMPTION

Leakage current in SRAM is measured as the current drawn from VDD to GND while the SRAM cell is at static or hold condition. The static power or leakage power is the amount of power dissipated at hold operation. Fig. 5(a) shows the leakage power variations for proposed 8T SRAM w.r.t. the temperature conditions ranging from 0 to 100°C. It is observed that the spread of the distribution curves of leakage power of
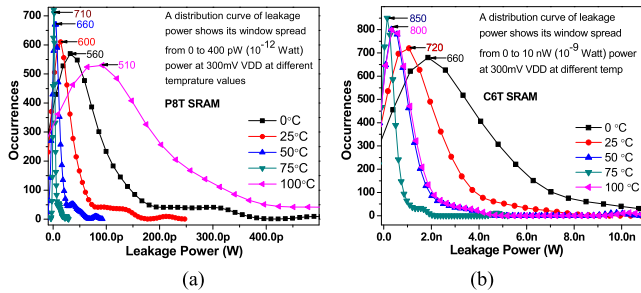
FIGURE 5. (a) Distribution curves of leakage power of 8T SRAM (b) Leakage power of C6T SRAM at various temperature values.
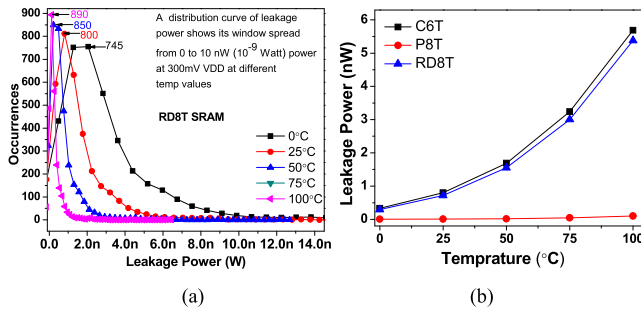

FIGURE 6. (a) Distribution curves of leakage power of RD8T SRAM cell in 10,000 Monte Carlo simulations. (b) Leakage power of proposed 8T SRAM compared with C6T and RD8T SRAM at various temperature values.

C6T SRAM (in Fig. 5(b)) and RD8T SRAM (in Fig. 6(a)) at various temperature values is wider than that of proposed 8T SRAM (Fig. 5(a)). The leakage power of proposed 8T SRAM ranges from 0pW to 200pW at various temperature values. Further, the mean (μ) values of leakage power are observed at different temperatures as shown Fig. 6(b). It is observed that the proposed SRAM has low leakage power variations at higher temperature values. It shows a remarkable improvement as the proposed cell has negligible leakage power as compared to C6T and RD8T SRAM at various process-voltage-temperature (PVT) values.
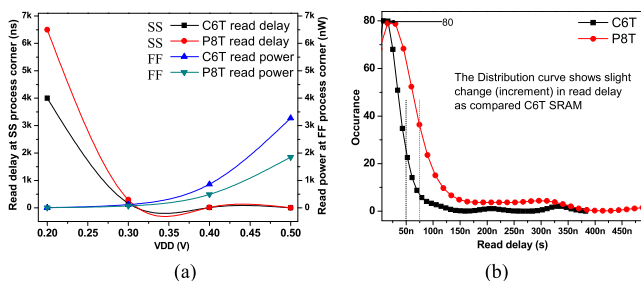

FIGURE 7. (a) Read delay and power consumption at different supply voltages. (b) Distribution curve of read delay at 300mV VDD.

B. READ DELAY AND POWER ANALYSIS

Read delay is measured when *RWL* is activated and *BLB* discharges and reaches to the minimum sensing voltage required by SA [34]. It is obtained from a 32-bit SRAM cell column architecture having a bit-line capacitance of 320fF at worst case process corner (SS). The power measured till the read access time is defined as a read access power of SRAM. Fig. 7(a) shows the plot of read delay and power in SS and

FF process corners. It is noticed that proposed PFC8T SRAM have similar read access time as compared to C6T SRAM with less read power consumption as shown in Fig. 7(b).

C. WRITE DELAY AND POWER ANALYSIS

The write '1' access time is measured as the time when *WWL* signal is triggered and storage node *Q* reaches to 90% of VDD value. Similarly, write 0 access time is defined as the time when *WWL* signal is activated and storage node *Q* reaches to 10% of VDD value. The write dynamic power is measured as the product of average current flow and the source voltage at the write access time. Fig. 8(a) show the mean (μ) and sigma (σ) values of write '1' delay and power at various supply voltages. The delay time and power are compared with C6T and RD8T SRAM, which show the proposed PFC8T SRAM has similar write access time. The write power of proposed PFC8T SRAM shows reduction of 2.4% and 7% as compared to C6T and RD8T, respectively at 300mV VDD as shown in Fig. 8(b). Due to profound reduction in leakage power with slight improvement in read-write power, the proposed cell shows better alternative for power hungry wireless devices such as mobile, laptops and medical portable devices.
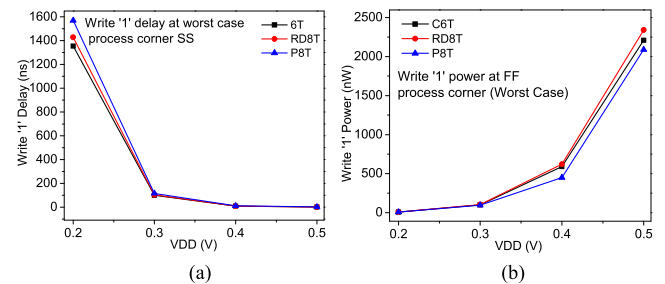

FIGURE 8. (a) Write '1' delay, power and the power delay product in worst case process corners at different VDD. (b) Distribution curve of write '1' delay at worst case process corner (SS).
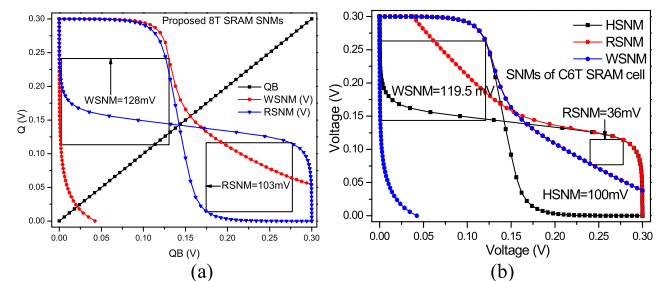

FIGURE 9. (a) Read static noise margin (RSNM) and write static noise margin (WSNM) of proposed 8T SRAM is observed at 300mV VDD. (b) RSNM, HSNM and WSNM of C6T SRAM 300mV VDD.

D. RSNM AND DYNAMIC READ MARGIN

Read static noise margin (RSNM) is measured by applying a DC noise voltage source at one of the storage node *Q* or *QB* and examining the effect on other. The RSNM is examined in the read operation when *RWL* is HIGH and *WWL* is LOW. The control signal (*CS*) is kept LOW which disconnects the path between node *Q* and *GND* which makes a decoupled logic from *BLB* to *GND*. The RSNM of proposed 8T is shown in Fig. 9(a), which shows an improvement by 2.86× and

1.03× than that of C6T and RD8T SRAM, respectively as shown in Fig. 9(b) and Fig. 10(b) respectively, at 0.3V VDD.
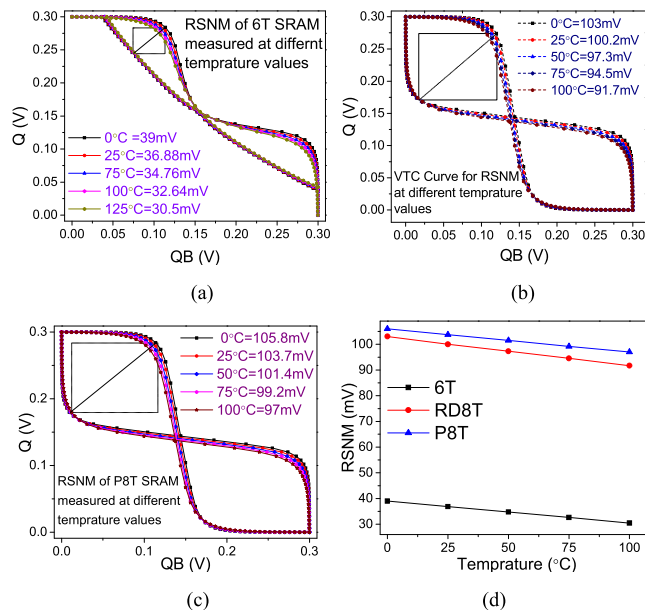

FIGURE 10. (a) RSNM of C6T (b) RSNM of RD8T (c) RSNM of P8T is observed at 300mV VDD in different temperature values. (d) Comparison of RSNM values of C6T, RD8T and P8T at different temperatures.
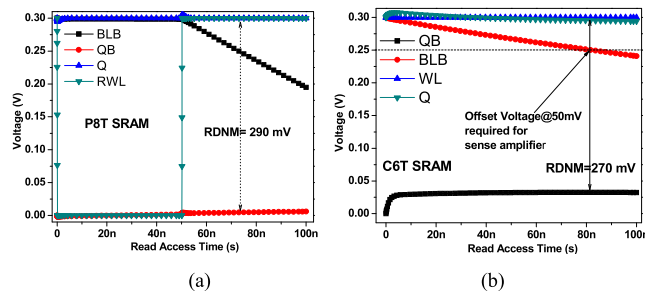

FIGURE 11. (a) Read dynamic noise margin (RDNM) of proposed 8T SRAM (b) RDNM of C6T SRAM at 300mV VDD.

Simultaneously, the RSNM values for C6T, RD8T and P8T SRAM are observed at different temperature values as shown in Fig. 10(a), (b) and (c), respectively. The comparison of RSNM values are also shown in Fig. 10(d). It shows that the proposed cell has better read stability than C6T and RD8T SRAM. On the other hand, while measuring dynamic read margin, *RWL* is activated HIGH and *WWL* is kept at LOW. Further, when the *BLB* reaches to the minimum offset voltage required for sensing the stored information, the difference between the values of storage node *Q* and *QB* is defined as read dynamic noise margin (RDNM). Fig. 11(a) and 11(b) shows the observation of RDNM of proposed 8T SRAM. The RDNM of proposed SRAM comes out to be 290mV at worst case process corner, which is 7.4% better as compared to C6T SRAM. The distribution curve in Fig. 12(a) shows that the proposed 8T SRAM has RDNM distributed near to 300mV at 0.3V VDD and has narrow variations at different PVT values as compared to C6T as shown in Fig. 12(b).
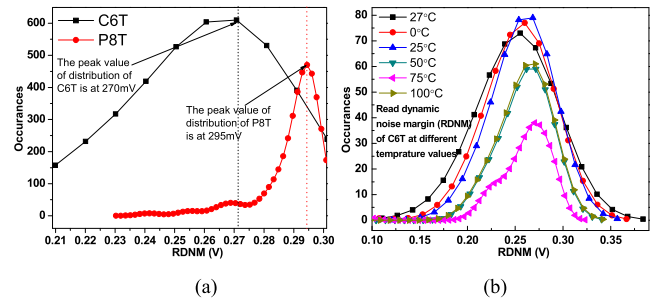

FIGURE 12. (a) Comparison of distribution curves of RDNM of PFC8T and C6T SRAM at 300mV VDD. (b) Distribution curve of RDNM of C6T SRAM at different temperature values at 300mV VDD.

E. WSNM

The write static noise margin (WSNM) is considered at the time of write operation by initiating a linear DC noise at one of the storage node and observing the effect of the noise at other. To achieve it the *WWL* is kept HIGH, *RWL* kept LOW and the *CS* is kept HIGH. Fig. 9(a) shows that the WSNM of PFC8T as 128mV at 300mV VDD, which is 7.1% (119.5mV) better than that of C6T SRAM as shown in Fig. 9(b).

F. WRITE TRIP POINT (WTP)

Write trip point (WTP) is measured at the time of write operation while *WWL* asserted HIGH and *RWL* is LOW. It can be observed by two methods one by adding a linearly variable DC voltage source at *BL* and observing its effect on the *BLB* and other by varying *WWL* and writing through *BL* and *BLB* [36]. The WTP when *WWL* varies comes out to be 30mV which is 43% better than C6T SRAM (21mV) at 300mV VDD in worst case process corner (SF). Fig. 13 shows the comparisons of simulation results of WTP observed for PFC8T, C6T and RD8T at different supply voltages. The write margin (WTP) is the important factor to improve the writing ability of SRAM.
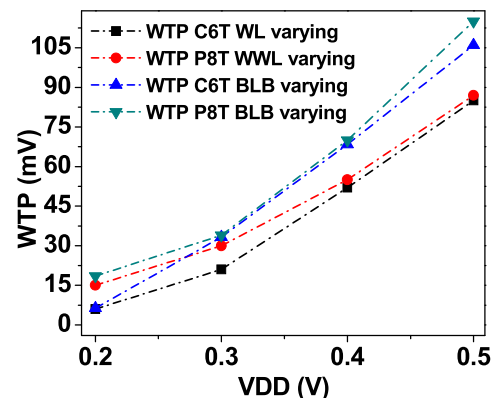

FIGURE 13. Write trip point (WTP) at worst case (SF) process corner.

G. I_on AND I_off CURRENT ANALYSIS

Another key metric for the read state is the read current or on current. The on current is the current across the pass transistor that discharges the bit-line during the read state. The differential sense amplifiers can be used in a single ended read

configuration using a reference voltage at the other input [37] for structures other than 6T. In the proposed 8T SRAM, the on current is lower due to the stacking effect caused by inclusion of MN7. While the stacking effect decreases the read current, it also reduces the leakage current of the access transistors. This causes a higher I_{on}/I_{off} for the 8T cell as shown in Fig. 14.
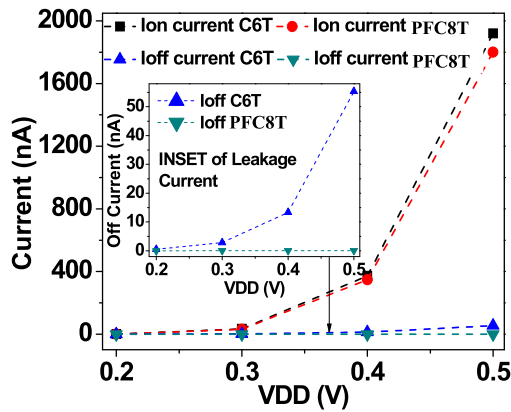

FIGURE 14. I_{on} and I_{off} ratio measured at different supply voltages. Inset: Off current of proposed 8T SRAM as compared to C6T SRAM.
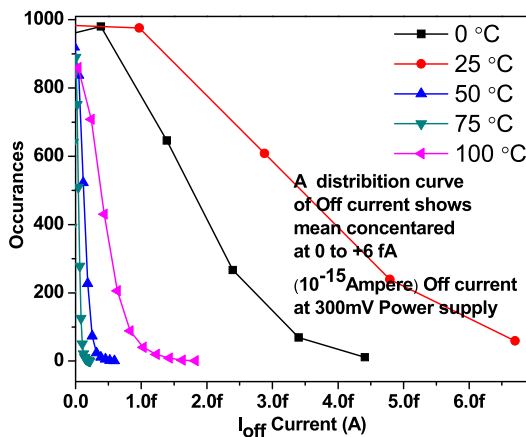

FIGURE 15. Off-Current distribution curve analysis of proposed 8T SRAM measured at different temperature values.

In addition, the higher I_{on}/I_{off} ratio increases the sensing margin, the maximum cells per bit-line, and the sensing timing window. The proposed cell shows much better I_{on}/I_{off} current ratio as compared to conventional SRAM architectures. The technique to reduce the OFF current evolves as a major factor to improve ON-OFF current ratio. The distribution curve for OFF current is shown in Fig. 15. The 6σ process variation shows deviations from 0 to +6 fA (Femto Ampere).

V. RESULTS SUMMARY FOR 2-kb ARRAY

A read decoupled positive feedback controlled (PFC) 8T SRAM cell for fast and low power object tracking is proposed in this work. Furthermore, the proposed cell is used to implement a 2-kb macro. The proposed cell is designed in 65-nm standard CMOS technology. It is observed from

the simulation results that the proposed cell shows a superior performance in terms of cell stability, write trip point (WTP) and leakage power as compared to existing SRAM cells. The proposed cell also shows better ON to OFF current ratio (I_{on}/I_{off}) and energy for read and write operations. Further, the leakage power and energy is observed for 2-kb macroblock and it shows improved outcomes at different supply voltages in subthreshold regime.

The proposed 8T SRAM power consumption is compared with the total power consumed by conventional 6T and RD8T SRAM cell. It is observed that the total power is reduced to $0.33\times$ as compared to C6T and RD8T SRAM. Table 2 shows the summary of post-layout simulation results of proposed PFC8T SRAM cell based 2-kb SRAM at 0.3V power supply. Further, it is observed that the WSNM is improved by 7.1%, RSNM by $2.86\times$; WTP by 43% as compared to C6T at worst case process corners. The power delay product (PDP) for write 1 operation is reduced by 32.85% as compared to C6T SRAM cell. The leakage power is also reduced by $82\times$ as compared C6T SRAM cell based 2-kb array at 300mV VDD.

TABLE 2. Summary of mean (μ) value of simulation results of proposed PFC8T SRAM based 2-kb macro in UMC 65-nm Technology using 1,000 Monte Carlo simulations at standard 6-sigma process variance.

SRAM	Leakage Power (pW)	Write '1' Delay (ns)	Write '1' Power (nW)	Read Delay (ns)	Read Power (nW)	RSNM (mV)	WSNM (mV)	WTP (mV)
C6T	850	99.43	98.4	9.17	860	36	119	21
RD8T	786	104	103.1	10.8	720	100	120	30
P8T	10.5	115.8	96	17	492	104	128	28

VI. OBJECT DETECTION AND TRACKING

An image is an array represented by a numbers of bits. An image is defined as a two-dimensional function $f(X, Y)$, where X and Y are the coordinates of image, and the amplitude of $f(X, Y)$ defines the intensity of the image at that point. The object detection and tracking are achieved using various steps, which are mentioned in the block diagram shown in Fig. 16. In proposed object detection and tracking algorithm, rectangular shape macroblocks are placed at every entrance points in the FOV. Further, the object is detected by considering the difference of root mean square (RMS) values of reference and current frame. After detecting object, quadtree based approach is employed to minimize the bounding box. This is the one of the major contributions of the proposed work to reduce the processing time, logical comparators, memory utilization and henceforth power consumption. Thereafter, object tracking is achieved by using macroblock resizing. Further, implementation of memory required for the proposed approach is performed using ultra low power (ULP) 8T SRAM cell.

A. SEGMENTATION

Image segmentation is a basic step in image processing and is also a significant part of image analysis. The macroblocks are placed at various locations of the FOV of camera, where
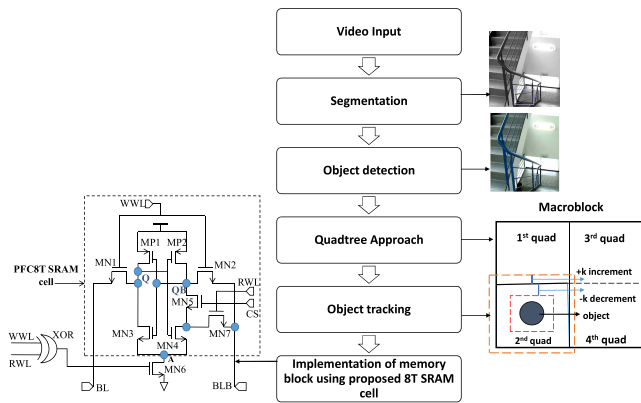
FIGURE 16. Block diagram of proposed object tracking algorithm and its memory cache implementation using proposed 8T SRAM.

the probability of entering the object is high. These entry blocks, which are in rectangular shape, are denoted as initial macroblocks. The captured RGB frame is converted into gray image and the initial macroblocks are considered as reference macroblocks. Fig. 17 shows the most probable entry locations and corresponding macroblocks.
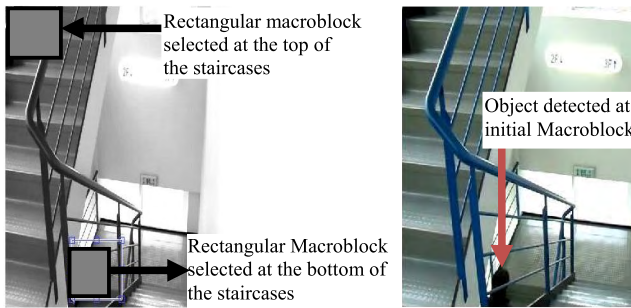

FIGURE 17. Rectangular macroblocks are selected at the entry points in the FOV where the possibility of entering the object is high and detected at the bottom.

B. OBJECT DETECTION

To detect the object, RMS values of macroblocks are considered. The RMS value of macroblock is calculated using equation (1).

$$rms = \sqrt{\frac{\sum_{k=1}^{N} p_k^2}{N}} \qquad (1)$$

where, rms denotes root mean square (RMS) value of macroblock, N is number of pixels in macroblock and p_k is gray intensity value at k^{th} pixel. The difference of RMS value between reference macroblock and current macroblock is determined and compared with an adaptive threshold value for object detection. When the difference of RMS value becomes greater than or equals to the threshold, in that case the object is detected in the FOV.

C. QUADTREE METHOD

To determine the exact location of object in the macroblock and to reduce the size of bounding box, quadtree decomposition approach is used [38]. The macroblocks of

reference and current frames are divided into four quadrants. The RMS values of corresponding quadrants of reference and current frames are compared. If the difference of RMS values of at least three quadrants is greater than the threshold, then minimum bounding box is achieved. Otherwise, the quadrants with difference of RMS value greater than threshold value will be further divided into four parts. The procedure is repeated until the minimum bounding box is achieved. It should be noted that if we divide the macroblock into four parts, then the new threshold of each quarter is changed to 2× of the threshold value of initial macroblock, according to the equation (1). The number of pixels after dividing the frame into 4 parts will be N/4, thus RMS becomes twice of the original value.

In Fig. 18(a), the quadtree approach is defined by using the flow diagram. Fig. 18(b) shows the initial macroblock which is further divided into 4 equal quarters. As shown in Fig.18(b), if difference of RMS value of 2nd quad become greater than or equal to 2×threshold value of initial macroblock then it would be further divided into 4 parts. It will go on until at least three quarters reaches to $\sqrt{2^{(i+1)}} \times$ threshold of the initial macroblock. Where, 'i' represents number of divisions performed for each quadrant.
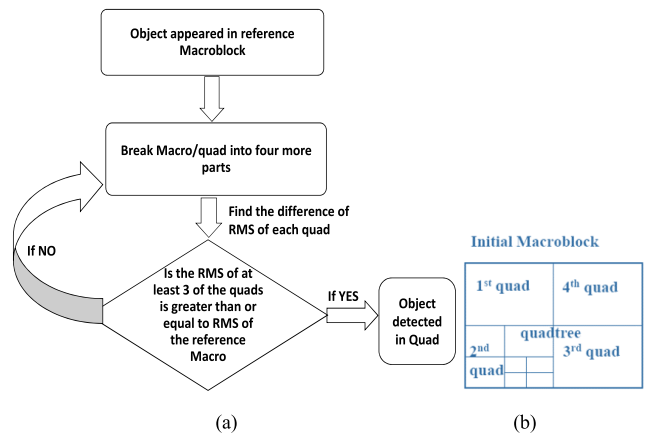

FIGURE 18. (a) Quadtree method for reducing the macroblock size of detected object. (b) Macroblock divided into four quarters to reduce the bounding box.

D. TRACKING USING MACROBLOCK RESIZING

After minimizing the bounding box around the object, our next objective is to improve the tracking speed and the area utilization in terms of memory. The less memory usage consequently results in minimizing the power consumption of memory blocks. Fig. 19 shows the macroblock where object is detected in 2nd quad. After applying quadtree method, the size of quad is reduced to $1/4\times$ the size of original macroblock. Therefore, to improve tracking speed with high accuracy rate and less memory utilization, it is intended to have motion vectors in all four directions as shown in Fig. 19(a) and 19(b). In Fig. 19, k is the directional vector pixels (DVPs) taken outside and inside of the rectangular macroblock or the sub-blocks reckoned from the quadtree approach.
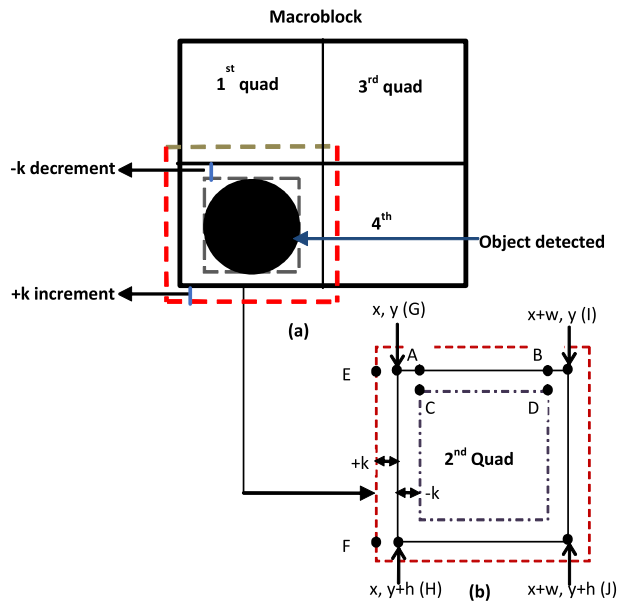
FIGURE 19. (a) Selected quad is incremented and decremented by a factor 'k' in all directions to trail the trajectory. (b) The selected 2nd quad where object is detected and further the object tracking algorithm is implemented using the eight rectangular sections just outside and inside to the selected 2nd quad.

The algorithm for object detection and tracking is described as a pseudo code in algorithm 1 and 2, respectively. It takes a video input and extracts it into numbers of frames. Further, the quadtree approach minimizes the bounding box and thereafter, the object tracking algorithm is implemented on the selected quad as shown in Fig 19(a) and 19(b).
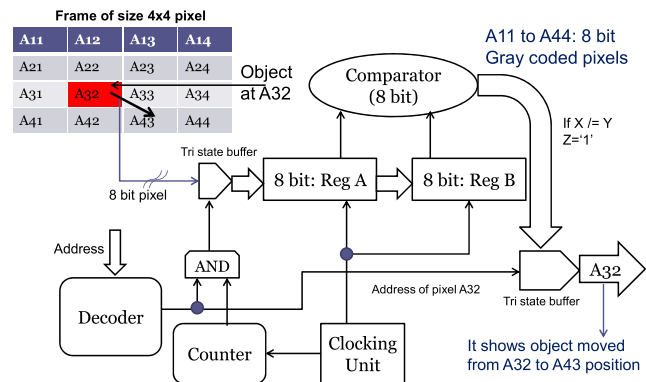

FIGURE 20. The object tracking memory block using 8T SRAM cell based 8-bit register for storing and comparing the Grey-coded frames.

Further, for tracking an object or group of objects in a scene, an object detection/tracking system is required. The processing unit comprises a memory block, comparators, counters, decoders and clocking unit, as shown in Fig. 20. The tracking system shown in the figure detects the motion vectors of consecutive frames. The memory block is required to store current and reference macroblocks. Further, an 8-bit comparator is used to compare the change in information

---

Algorithm 1 Object Detection

Input: video, macroblock_coordinates
Output: object_boundbox, objdet_frame
    thresh = calc_threshold (video);
    img= read_frame(video(1));
    ref_macroblock= macroblock(img,
                    macroblock_coordinates);
    ref_rms= rms(Ref_macroblock);
for i=1 to nframes **do**
      img = read_frame(video(i));
      cur_macroblock = macroblock(img,
                    macroblock_coordinates);
      cur_rms = rms(cur_macroblock);
      if (ref_rms-cur_rms)>=thresh **then**
        object_detect=1;
        object_boundbox=quadtree(cur_macroblock);
        objdet_frame=i;
      end if
.**if** mod(i,5)=0**then**
ref_rms= cur_rms;
      end if
end for
Definitions:

- video is a input video
- macroblock_coordinates are the initial macroblock coordinates.
- object_boundbox is the final coordinates where the object is detected after applying quadtree approach.
- objdet_frame is the frame in which object is detected.
- calc_threshold() takes video as an argument and calculates the threshold value for object detection.
- read_frame() takes video as an argument and returns frames.
- macroblock() selects macroblock in the FOV and returns coordinates of macroblock.
- rms() takes coordinates of block as an argument and calculates RMS value of the block.
- quadtree() takes input as macroblock where is object is detected and returns the minimum bounding box around the object.

---

of the macroblocks. With the aim of fast and low power tracking system, SRAM is used as a memory device due to its benefits of high performance and low power consumption. Evidently, the cache memory block for object detection/tracking requires arrays of SRAM cell to write, store, compare and read binary information [39]–[43]. In [39], a micro-programmable real-time video signal processor (VSP) -large scale integration (LSI) has been developed for constructing a parallel video signal processing system.

The VSP-LSI utilizes a multistage pipelined architecture and can handle complex image and signal processing applications such as high-speed edge detection, face

---

Algorithm 2 Object Tracking
Input: video, object_boundbox, objdet_frame, k, thresh
Output: new_boundbox, objdet_frame
    [x, y, h, w]=coord(object_boundbox);
    ref_rms = rms(object_boundbox);
      for i= objdet_frame+1 to nframes do
        img = read_frame(video(i));
        leftout_block=img(x-k: x, y: y+h);
        cur_rms=rms(leftout_block);
        if (ref_rms-cur_rms)>=thresh and
                     other_condition=0 **then**
          new_boundbox=img(x-k: x+w-k, y: y+h);
        else
          new_boundbox = img(x: x+w, y: y+h);
        end if
        if mod(i,5)=0**then**
          ref_rms = cur_rms;
        end if
      end for

Note: The **other_condtion** is defined as the conditions for the other 7 blocks just outside and inside of the 2nd quad. In our algorithm we took the left out rectangular block "EGHF" as shown in Fig. 19(b), where the co-ordinates are (x-k: x, y: y+h). Similarly, all the other 7 rectangular blocks are observed and if the **diff_rms** becomes greater than or equal to the threshold of initial macroblock than we expect that the object moves towards that direction.

Definitions:

- 'k' is the incremented or decremented pixels taken within and outside the 2nd quad.
- coord() takes block as an argument and returns x, y coordinates, height h and width w of the block.
- leftout_block is the left side outer rectangular block "EGHF" taken in our example. The movement of object is observed at the left side of the image.
- new_boundbox is the new moved bounding box according to the movement of object towards left side of the image.

detection and motion compensation. The LSI contains SRAM based cache for the realization. In [40], real-time face detection system is fabricated in a $0.13\mu m$ CMOS technology. It consists of 75,000 gate logic, 58,000 bit SRAM, and an ARM-Advanced Microcontroller Bus Architecture (AMBA) bus interface. In [41], a dynamically reconfigurable SRAM array for low-power mobile multimedia application is presented which shows how a SRAM array can significantly changes the overall multimedia power. However, a large number of memory arrays in portable tracking devices consume a huge amount of processing and leakage power. Therefore, a subthreshold 8T SRAM cell is proposed to reduce leakage power consumption, detailed in section III, which further utilized to implement SRAM arrays for tracking device.

VII. TRACKING SYSTEM ANALYSIS AND SRAM UTILIZATION: RESULTS AND COMPARISONS

The proposed object tracking algorithm uses less number of memory blocks to compare two different frames as shown in Table 3. Instead of comparing whole frames of size 704×576 in given example, the proposed algorithm uses a macroblock resizing approach. The average memory utilized in the proposed work can be determined from the information of mean of total numbers of pixels used to track the object. To measure the average power consumption of memory block, the amount of power consumption by memory bit cell is required. The average power of bit cell is measured in the supply range of interest, assuming equal probability of performing read and write operations. To authenticate the proposed algorithm in MATLAB tool, a monocular camera based surveillance video is considered. To support high-level video surveillance task and other processing algorithms, the server is equipped with an INTEL Core i7 4710HQ 2.5GHz CPU and 8GB RAM.

TABLE 3. Average memory size required/frame for detection and tracking with object detection and tracking delay.

Macro-block size (pixels)	Memory required for object detection/ frame (KB) without quadtree pixelation	Memory size required after quadtree pixelation (KB)	Average Memory required for tracking/ frame (KB)	Delay in detection (s)	Average Delay in Tracking/ frame (s)
13×16	0.20	0.06	37.15	2.08	0.15
23×18	0.41	0.12	39.85	1.73	0.15
73×56	4.08	1.00	39.65	1.71	0.15
57×73	4.10	2.20	37.62	1.68	0.18
64×79	5.00	1.30	36.84	2.12	0.18
75×74	5.55	5.55	39.00	1.84	0.16
100×112	11.2	2.85	38.20	1.80	0.17
115×119	13.6	3.40	42.37	1.70	0.15
149×148	22.0	10.0	41.40	2.00	0.16
215×137	29.4	7.45	37.00	1.74	0.16

The average leakage power/frame for C6T, RD8T and proposed PFC8T SRAM is equivalent to 272nW, 251.5nW and 3.2nW, respectively as shown in Table 4. The average total power consumption is the sum of power consumed while writing, reading and holding (leakage) a data into SRAM for tracking an object. The total power consumed is equivalent

$$\text{Tracking Accuracy Rate (TAR in \%)} = \sum_{i=k}^{N} \frac{\text{Area of proposed bounding box} - \text{Area of actual bounding box}}{\text{Area of actual bounding box}} \quad (2)$$

TABLE 4. Average leakage power observed in tracking an object at 0.3V VDD.

SRAM	Average leakage power (nW)	Average total power (mW) at worst case process corner (FF)
C6T	272nW	38.33mW
RD8T	251.5nW	33mW
PFC8T	3.2nW	23.5mW

to 38.33mW, 33mW and 23.5mW for C6T SRAM, RD8T and PFC8T SRAM based memory array required for tracking an object. Fig. 21 shows the set of frames extracted from a video. In the very first frame of the video the macroblock of size 64×79 is selected. To detect the object in the scene, a threshold value is determined which is equal to the *peak* of the difference of RMS value of the selected macroblocks. In Fig. 21, it is shown that the object enters the scene at frame 129 and after that a quadtree approach is used to reduce the bounding box size. Consequently, by using quadtree approach, the object can be tracked by using 1/4th or 1/8th numbers of pixels as compared to original macroblock. The object tracking output using macroblock resizing is shown in Fig. 21. From this observation, we calculated the accuracy of our proposed work. Table 5 shows the accuracy rate of the proposed work as compared to other works in the past.
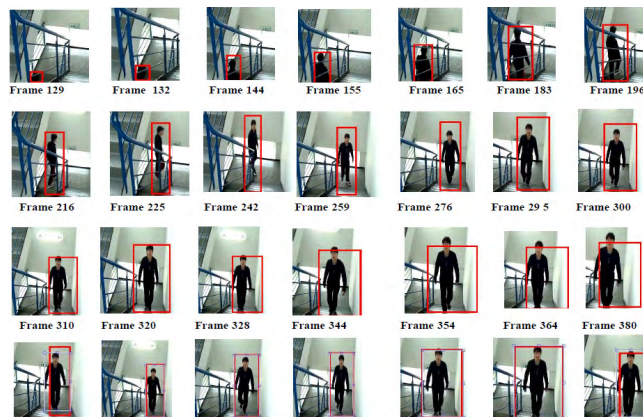

FIGURE 21. Object tracking using quadtree pixelation where the bounding box is selected at the entry points of the FOV.

TABLE 5. Comparison of tracking accuracy rate w.r.t the actual bounding box.

Tracking Accuracy Rate (%)	Classical background difference [27]	Back-ground difference [28]	Fusion algorithm [28]	PETS 2009 dataset [29]	iLIDS medium sequence [29]	Proposed
	88.5%	87.4%	91.0%	96.2%	95.2%	96.5%

The accuracy rate is calculated by using the equation (2), where the percentage of difference area of actual bounding box to proposed bounding box is measured. In the present work the accuracy rate comes out to be 96.5%. The accuracy rate could be different for different macroblock size. Equation (2), as shown at the bottom of previous page, where, k is the number of frame where the object is detected and N is the total number of frames present in the video.

VIII. CONCLUSION

The proposed work presents an ULP 8T SRAM cell with better leakage power, write static noise margin, write-ability and read margin as compared to C6T and 8T SRAMs. Further a fast, reliable, less memory usage object tracking algorithm and implementation of its memory block using ULP 8T SRAM is proposed. A quadtree based approach is employed to diminish the bounding box and to reduce the computations for fast and low power object tracking. The proposed object detection and tracking method is based on macroblock resizing, which demonstrates an accuracy rate of 96.5%.

Further, the leakage and total power consumption of required memory block for tracking algorithm is observed and implemented using a proposed 8T SRAM. The memory block implementation of tracking system using proposed 8T, C6T and RD8T has been observed at worst case process corner using Monte Carlo simulations.

It is observed that the average leakage power/frame for tracking system using proposed 8T SRAM is 82× and 75× better than that of C6T and RD8T SRAM, respectively. The average total power consumption is the sum of power consumed while writing, reading and holding (leakage) a data into SRAM for tracking an object. The total power consumed is equivalent to 38.33mW, 33mW and 23.5mW for C6T SRAM, RD8T and PFC8T SRAM based memory array required for tracking an object. The total power of proposed SRAM block is 1.64× and 1.4× better than that of C6T SRAM and RD8T SRAM based object tracking memory block.

REFERENCES

[1] S.-L. Wu et al., "A 0.35 V, 375 kHz, 5.43 μW, 40 nm, 128 kb, symmetrical 10T subthreshold SRAM with tri-state bit-line," *Microelectron. J.*, vol. 51, pp. 89–98, May 2016.

[2] J. Chen, K.-S. Chong, and B.-H. Gwee, "Ultra-low power read-decoupled SRAMs with ultra-low write-bitline voltage swing," *Circuits, Syst., Signal Process.*, vol. 33, no. 10, pp. 3317–3329, 2015.

[3] C. B. Kushwah, S. K. Vishvakarma, and D. Dwivedi, "Single-ended boostless (SE-BL) 7T process tolerant SRAM design in sub-threshold regime for ultra-low-power applications," *Circuits, Syst., Signal Process.*, vol. 35, no. 2, pp. 385–407, 2016.

[4] M. Moghaddam, S. Timarchi, M. H. Moaiyeri, and M. Eshghi, "An ultra-low-power 9T SRAM cell based on threshold voltage techniques," *Circuits, Syst., Signal Process.*, vol. 35, no. 5, pp. 1437–1455, 2016.

[5] B. H. Calhoun and A. Chandrakasan, "A 256 kb sub-threshold SRAM in 65 nm CMOS," in *IEEE Int. Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2006, pp. 2592–2601.

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

[7] Y. C. Lai and S. Y. Huang, "A resilient and power-efficient automatic-power-down sense amplifier for SRAM design," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 55, no. 10, pp. 1031–1035, Oct. 2008.

[8] D. Anh-Tuan, K. Zhi-Hui, and Y. Kiat-Seng, "Hybrid-mode SRAM sense amplifiers: New approach on transistor sizing," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 55, no. 10, pp. 986–990, Oct. 2008.

[9] B. Zhai, S. Hanson, D. Blaauw, and D. Sylvester, "A variation-tolerant sub-200 mV 6-T subthreshold SRAM," *IEEE J. Solid-State Circuits*, vol. 43, no. 10, pp. 2338–2348, Oct. 2008.

[10] B. H. Calhoun and A. P. Chandrakasan, "A 256-kb 65-nm sub-threshold SRAM design for ultra-low-voltage operation," *IEEE J. Solid-State Circuits*, vol. 42, no. 3, pp. 680–688, Mar. 2007.

[11] S.-W. Sun and P. G. Y. Tsui, "Limitation of CMOS supply-voltage scaling by MOSFET threshold-voltage variation," *IEEE J. Solid-State Circuits*, vol. 30, no. 8, pp. 947–949, Aug. 1995.

[12] T.-H. Kim, J. Liu, and C. H. Kim, "An 8T subthreshold SRAM cell utilizing reverse short channel effect for write margin and read performance improvement," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, vol. 7, Sep. 2007, pp. 241–244.

[13] K. Zhang *et al.*, "SRAM design on 65 nm CMOS technology with integrated leakage reduction scheme," in *Symp. Dig. Tech. Papers VLSI Circuits*, Jun. 2004, pp. 294–295.

[14] J. Chen, L. T. Clark, and T.-H. Chen, "An ultra-low-power memory with a subthreshold power supply voltage," *IEEE J. Solid-State Circuits*, vol. 41, no. 10, pp. 2344–2353, Oct. 2006.

[15] L. Chang *et al.*, "An 8T-SRAM for variability tolerance and low-voltage operation in high-performance caches," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 956–963, Apr. 2008.

[16] A. L. Bovic, *Image and Video Processing*. New York, NY, USA: Academic, 2000.

[17] A. M. Tekalp, *Digital Video Processing*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1995.

[18] Y. Amit, *2D Object Detection and Recognition*. Cambridge, MA, USA: MIT Press, 2002.

[19] R. F. Gonzalez and R. E. Wood, *Digital Image Processing*. Singapore: Pearson Education, 2001.

[20] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, pp. 1–45, 2006.

[21] S. Zhang, C. Wang, S. C. Chan, X. Wei, and C. H. Ho, "New object detection, tracking, and recognition approaches for video surveillance over camera network," *IEEE Sensors J.*, vol. 15, no. 5, pp. 2679–2691, May 2015.

[22] M. M. Azab, H. A. Shedeed, and A. S. Hussein, "New technique for online object tracking-by-detection in video," *IET Image Process.*, vol. 8, no. 12, pp. 794–803, 2014.

[23] R. K. Behera, P. Kharade, S. Yerva, P. Dhane, A. Jain, and K. Kutty, "Multi-camera based surveillance system," in *Proc. World Congr. Inf. Commun. Technol. (WICT)*, Trivandrum, India, Oct./Nov. 2012, pp. 102–108.

[24] X. Lei and A. Fu, "2-D maximum-entropy thresholding image segmentation method based on second-order oscillating PSO," in *Proc. 5th Int. Conf. Natural Comput.*, Tianjin, China, Aug. 2009, pp. 161–165.

[25] S. Kumar, M. Pant, and A. K. Ray, "2D maximum entropy method for image thresholding converge with differential evolution," *Adv. Mech. Eng. Appl.*, vol. 2, no. 3, pp. 289–292, 2012.

[26] J. Jeon and R. Manmatha, "Using Maximum Entropy for Automatic Image Annotation," in *Proc. Int. Conf. Image Video Retr. (CIVR)*, vol. 3115. Berlin, Germany, 2004.

[27] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 747–757, Aug. 2000.

[28] G. Han, X. Li, N. Sun, and J. Liu, "A robust object detection algorithm based on background difference and LK optical flow," in *Proc. 11th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, Xiamen, China, Aug. 2014, pp. 554–559.

[29] W. Wang, R. Nevatia, and B. Yang, "Beyond pedestrians: A hybrid approach of tracking multiple articulating humans," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Waikoloa, HI, USA, Jan. 2015, pp. 132–139.

[30] V. Reddy, C. Sanderson, and B. C. Lovell, "Improved foreground detection via block-based classifier cascade with probabilistic decision integration," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 1, pp. 83–93, Jan. 2013.

[31] E. López-Rubio and R. M. Luque-Baena, "Stochastic approximation for background modelling," *Comput. Vis. Image Understand.*, vol. 115, no. 6, pp. 735–749, 2011.

[32] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recognit. Lett.*, vol. 27, no. 7, pp. 773–780, 2006.

[33] S. Zhang, S.-C. Chan, B. Liao, and K. M. Tsui, "A new visual object tracking algorithm using Bayesian Kalman filter," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Melbourne, VIC, USA, Jun. 2014, pp. 522–525.

[34] B. S. Reniwal, P. Bhatia, and S. K. Vishvakarma, "Design and investigation of variability aware sense amplifier for low power, high speed SRAM," *Microelectron. J.*, vol. 59, pp. 22–32, Jan. 2017.

[35] E. Grossar, M. Stucchi, K. Maex, and W. Dehaene, "Read stability and write-ability analysis of SRAM cells for nanometer technologies," *IEEE J. Solid-State Circuits*, vol. 41, no. 11, pp. 2577–2588, Nov. 2006.

[36] J. Wang, S. Nalam, and B. H. Calhoun, "Analyzing static and dynamic write margin for nanometer SRAMs," in *Proc. ACM/IEEE Int. Symp. Low Power Electron. Design (ISLPED)*, Bangalore, India, 2008, pp. 129–134.

[37] M. E. Sinangil, N. Verma, and A. P. Chandrakasan, "A reconfigurable 8T ultra-dynamic voltage scalable (U-DVS) SRAM in 65 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 44, no. 11, pp. 3163–3173, Nov. 2009.

[38] E. Shusterman and M. Feder, "Image compression via improved quadtree decomposition algorithms," *IEEE Trans. Image Process.*, vol. 3, no. 2, pp. 207–215, Mar. 1994.

[39] M. Yamashina *et al.*, "A microprogrammable real-time video signal processor (VSP) LSI," *IEEE J. Solid-State Circuits*, vol. SSC-22, no. 6, pp. 1117–1123, Dec. 1987.

[40] Y. Hori and T. Kuroda, "A 0.79-mm^2 29-mW real-time face detection core," *IEEE J. Solid-State Circuits*, vol. 42, no. 4, pp. 790–797, Apr. 2007.

[41] M. Cho, J. Schlessman, W. Wolf, and S. Mukhopadhyay, "Reconfigurable SRAM architecture with spatial voltage scaling for low power mobile multimedia applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 1, pp. 161–165, Jan. 2011.

[42] M. Cho, J. Schlessman, H. Mahmoodi, M. Wolf, and S. Mukhopadhyay, "Postsilicon adaptation for low-power SRAM under process variation," *IEEE Des. Test Comput.*, vol. 27, no. 6, pp. 26–35, Nov. 2010.

[43] S. Kim, B. J. Lee, J. W. Jeong, and M. J. Lee, "Multi-object tracking coprocessor for multi-channel embedded DVR systems," *IEEE Trans. Consum. Electron.*, vol. 58, no. 4, pp. 1366–1374, Nov. 2012.

POORAN SINGH received the M.Tech. degree from ABV-IIITM, Gwalior, in 2010. He is currently a Ph.D. Scholar with the Electrical Engineering Department (VLSI Design), IIT Indore. He is researching with Dr. S. K. Vishvakarma at the Nanoscale Devices, VLSI Circuit and System Design Research and Development Laboratory with a focus on Ultra Low power, high-stability feedback controlled SRAM cell for FPGA and DSP Memories. He received the prestigious Fulbright-Nehru Doctoral Research Fellowship from 2014 to 2015, a very highly competitive Fellowship Award to carry on his research in USA for nine months. He was with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, USA, under the Fulbright-Nehru Fellowship.

SANTOSH KUMAR VISHVAKARMA received the M.Tech. degree in microelectronics from Punjab University, Chandigarh, India, in 2003, and the Ph.D. degree from IIT Roorkee, Roorkee, India, in 2010. From 2009 to 2010, he was with the University Graduate Center, Kjeller, Norway, as a Post-Doctoral Fellow with Prof. T. A. Fjeldly under the European Union Project COMON on compact modeling development and parameter extraction of multi-gate MOSFETs. He is currently with the School of Engineering, IIT Indore, Indore, as an Assistant Professor, where he is leading the Nanoscale Devices, VLSI Circuit and System Design Laboratory. His current research includes nanoscale devices and circuits, ultralow-power digital and analog circuit design and their technology, FPGA-based design, power reduction techniques in FPGA-based system design, multi-gate and multi-fin MOSFET, and tunnel FET and their circuit applications in memories. Recently, he is involved in high speed transceiver design and graphene-based digital standard cell design. He is also interested in Internet of Things for healthcare and defence applications.

• • •