Research Article

A 7-nm-Based 5R4W High-Timing Reliability Regfile Circuit

Wanlong Zhao [1], **Yuejun Zhang** [1,2] **Liang Wen,**[3] **and Pengjun Wang**[4]

[1]*Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo, China*
[2]*Zhejiang Key Laboratory of Mobile Network Application Technology, Ningbo University, Ningbo, China*
[3]*Department of Electronic Technology, China Coast Guard Academy, Ningbo, China*
[4]*College of Electrical and Electronic Engineering, Wenzhou University, Wenzhou 325035, China*

Correspondence should be addressed to Yuejun Zhang; zhangyuejun@nbu.edu.cn

Academic Editor: Shashikant Patil

Register file (Regfile), as the bottleneck circuit for processor data interaction, directly determines the computing performance of the system. To address the read/write conflict and timing error problems of register heap, this paper proposes a 5R4W high-timing reliability Regfile circuit design scheme. First, the scheme analyzed the principles of timing errors such as read/write conflicts, write errors, and read errors in the Regfile circuit; then adopted the timing separation method of independent control of the read/write process by clock double edges to solve multiport read/write conflicts, designed a mirror memory check circuit to avoid write errors caused by the word line delays, and used a phase-locked clock feedback structure to eliminate read errors caused by the data timing fluctuations; in the TSMC 7 nm FinFET process, a 64×74-bit 5R4W Regfile circuit was implemented using a fully customized layout. Experimental results show that the Regfile circuit has an area of $0.13 \, \text{mm}^2$ and consumes 5.541 mW. The circuit operates at a maximum frequency of 3.8 GHz at -40 to $-125°C$ and 0.75 V, and is capable of detecting write errors caused by a clock jitter exceeding 30 ps or a frequency above 5 GHz.

1. Introduction

With the rise of intelligent computing and machine learning (ML) technologies, artificial intelligence (AI) [1] is gradually moving from theoretical research in computer science disciplines to real-life practical applications, such as autonomous driving, medical imaging, astronomical observation etc., while placing increasingly high demands on processor performance [2]. The market demand for processors optimized for AI algorithms or with high-speed parallel processing features is growing rapidly. In the face of massive data and large computing power demand, the development of high-performance processors represented by AI processors and graphics processing unit (GPU) [3] is rapidly gaining momentum. The demand for higher performance is increasing, and there are already technologies such as multicore [4–6] and hyperthreading [7–9] to achieve high-performance processors, but higher performance means higher requirements for speed, bandwidth, and power consumption. Especially in recent years, the memory speed has been much lower than the processor's operating speed, and the bandwidth and

power consumption problems brought about by frequent data exchange make memory a major factor limiting arithmetic power, while the register file (Regfile) is becoming more and more prominent as a key circuit for processor data interaction and its bottleneck role [10], and there is an urgent need to carry out high-performance and high-reliability Regfile research.

In the study of high-performance Regfile circuits, the literature by Tsuji et al. [11–13] proposes the use of adiabatic quantum flux parametron (AQFP) components for Regfile circuit design, which is based on superconducting technology with low power consumption and ultrafast switching speed, and therefore has superior potential in terms of computing speed and energy efficiency. AQFP is also considered as one of the alternatives to complementary metal oxide semiconductor (CMOS) technology, so the use of its Regfile design is important for the implementation of high-performance processors in superconducting technology. Liu et al. [14] used SiGe heterojunction bipolar transistor technology for ultrahigh speed optimization of the Regfile circuit to shorten the execution cycle of each instruction in the processor and

increase the computational speed. Nguyen et al. [15, 16] and Yantir et al. [17] used a double pump technology for Regfile optimization to improve the throughput and operation speed under the conventional CMOS technology. However, the high-speed design leads to the degradation of the reliability of the Regfile circuit, especially as the operation speed becomes increasingly faster with faster reliability becoming more and more important, so it is also essential to face the design of the Regfile reliability. The literature by Chen et al. [18] proposes an enhanced Regfile circuit access data reliability scheduling method that detects and judges when accessing data and corrects or rolls back if there is an error. The need to make judgments at each access to the data can consume a lot of energy and generate excessive energy consumption overhead. The literature by Yue et al. [19] proposes an energy-efficient error correction code (ECC) mechanism to address such problems, which is designed to avoid unnecessary ECC generation and verification to reduce energy consumption by exploiting the error-tolerance sensitivity of instructions and the error tolerance sensitivity of data bits, which reduces the energy consumption by 86% compared to the conventional ECC. Ahangari et al. [20] and Jie et al. [21] considered the large amount of power consumption generated by the ECC mechanism with Regfile being designed using narrow width register values to propose a highly reliable Regfile circuit architecture. All the above studies are designed for data reliability, and less research has been done on read/write conflicts and timing reliability issues. In high speed, high-throughput application scenarios, optimization for data reliability alone cannot avoid errors caused by the timing problems, and the simultaneous operability of ports is an inevitable development trend. The processor operating frequency is getting higher and higher, the timing reliability of the Regfile circuit is becoming more and more important, the impact of the read/write conflict problem is also increasing, and the ever-shortening work cycle will lead to a more fragile Regfile circuit timing and more prone to timing problems, with read/write conflict and timing problems gradually becoming the main factors limiting the Regfile performance in high-speed application scenarios.

This paper conducts an in-depth study of timing reliability in Regfile circuits, which first analyzes read/write conflicts and read/write timing errors, and investigates the error generation principle and its impact on the performance of Regfile circuits. Then, to solve the problem of multiport simultaneous read/write conflict, the read/write timing separation technique was used to split the read/write operation process and complete the read/write timing control circuit architecture; to address the write error caused by the word line delay, a single word line error detection circuit was designed to check the circuit for redundant storage of a specific array using mirror memory; a phase-locked clock feedback structure was added to eliminate the data read error caused by the timing fluctuations. To ensure compatibility with the processor's instruction data width, this circuit needs to store 74-bit data for the processor's instruction data processing. Finally, a 64×74-bit 5R4W Regfile circuit with high-timing reliability was designed and fabricated using the TSMC 7-nm process.

The structure of this paper is organized as follows: the second part presents the theoretical analysis of timing errors; the third part describes the design implementation of the 5R4W high-timing reliability Regfile circuit; the fourth part evaluates the performance of the design scheme based on the experimental results; the fifth part gives the final conclusion.

2. Read/Write Conflict and Timing Error Mechanism Analysis

High-performance Regfile circuits, with the need for large throughput bandwidth and ultrahigh operating speeds, continue to pursue the number of multiple ports and high-operating frequencies, and their read/write conflicts and timing errors are becoming increasingly prominent and have become a major factor limiting performance improvements and bandwidth increases in today's high-performance processors. Read/write conflict refers to the read operation in the mixed read/write high-speed operating environment and it conflicts with the resources accessed by the write operation. A timing error refers to a situation in which incorrect data are stored or output during a read/write operation caused by the clocking problems. Regfile circuit serves as the highest speed storage circuit in a high-performance processor, and its timing reliability becomes an issue that must be considered in the design. Next, we analyze the mechanism of the read/write conflict and timing error problem generation.

The read/write conflict problem is a problem that cannot be avoided by generating the multiport technology under the high-bandwidth demand. Based on the sequence of read and write operations, there are three types of read/write dependencies that can be distinguished [22]: (1) write after read (WAR) correlation refers to the data correlation caused by the fact that the result register to be written back by the later executed instruction is the same as the register read by the first executed instruction. Theoretically, the later instruction in the pipeline must not be executed before the previous instruction with WAR correlation; otherwise, the new result will be written before it is read, resulting in data read errors. (2) Write after write (WAW) correlation refers to the data correlation caused by the fact that the result register written back by the later instruction is the same as the register written back by the first instruction. Theoretically, the later instruction in the pipeline must not be executed before the earlier instruction with WAW correlation; otherwise, the result of the later instruction will be overwritten by the result of the earlier instruction after writing, resulting in data writing loss error. (3) Read after write (RAW) correlation refers to the data correlation caused by the fact that the register where the later instruction is executed to read data is the same as the register in which the first instruction is executed to write back data. In theory, the pipeline must not be executed before the instruction that has RAW correlation; otherwise, the read operation will be executed before the writing reads the wrong data.

In the case of WAR correlation, for example, the reason for its generation is shown in Figure 1. If read/write operations are performed simultaneously to generate conflicts, it will lead to read and write operations for the same memory
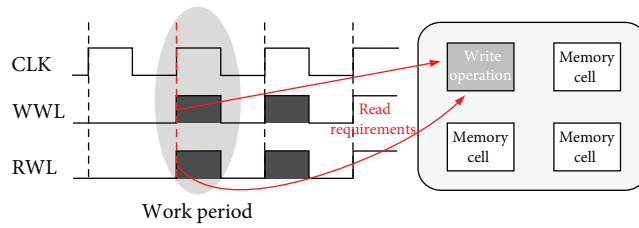
FIGURE 1: Diagram of read/write conflict generation.
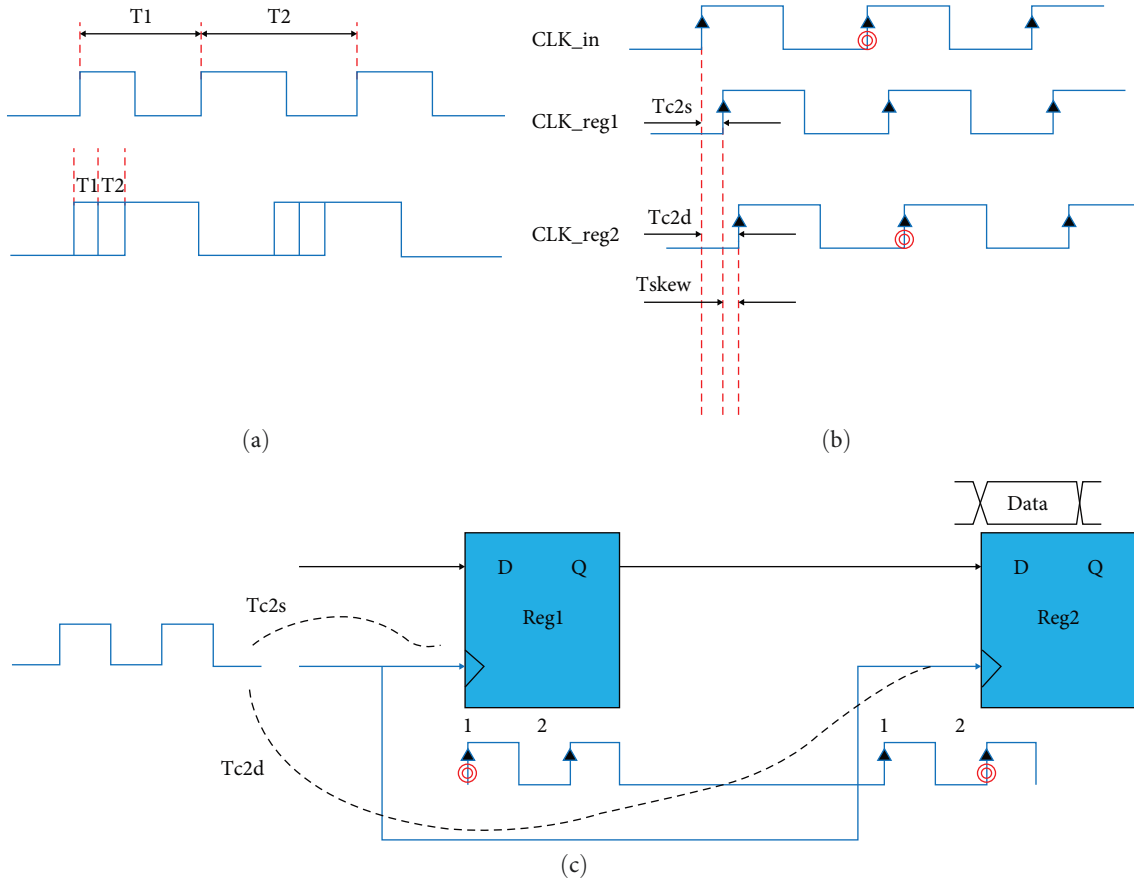

(a)

(b)

(c)

FIGURE 2: Timing error generation: (a) clock jitter, (b) clock skew, and (c) causes of clock skew.

cell having to use two cycles, and computing resources are greatly wasted, which reduces the operational efficiency.

The problem of read/write conflict has been extensively studied in the field of integrated circuits. For instance by Suzuki et al. [23] introduced a novel cell bias technique to mitigate the effect of read/write interference; [24] enhanced the WordLine-Resistance/Capacitor and refined the logic to reduce the read/write conflict issue. Based on the theoretical analysis, this paper proposes an optimized circuit structure design to address the challenge of read and write conflicts.

Timing errors are usually caused by the clocking problems. When the high-speed clock pulse width becomes narrower, the percentage of timing errors increases relatively, leading to a higher possibility of timing errors in the circuit and reduced timing reliability. As process technology advances, chip designers strive for higher performance by lowering the power supply voltage and increasing the operating frequency. However, this leads to timing issues that become critical in high-speed and high-complexity integrated circuits [25, 26]. As a result, timing problems in high-performance processors will become more prominent as the operating speed increases. Clock jitter and clock skew are the two main causes of timing problems in high-speed computing. Clock jitter, which is usually generated by internal causes of the clock generator, is a phenomenon in which the clock period changes briefly at a definite point, thus causing the length of the clock period to vary from cycle to cycle.

Since the clock edge change is not instantaneous in the ideal case, the clock will have a change process from high to low or low to high, which leads to a transient random change of the clock at a given point, with the timing schematic shown in Figure 2(a). Clock skew is caused by wiring length
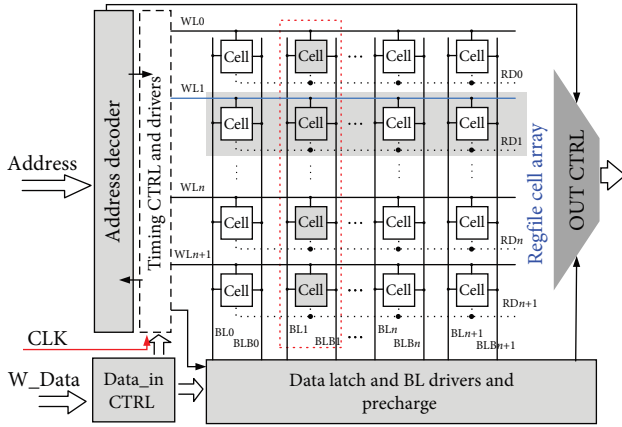
FIGURE 3: Regfile circuit structure.

and load differences, and the same clock generates delay differences between multiple subclocks. The generation process is shown in Figure 2(c), and the timing sequence is shown in Figure 2(b). Clock skew is the uncertainty in the clock phase and clock jitter is the uncertainty in the clock frequency. Both can cause errors when Regfile stores data or reads data, which can degrade processor performance. Therefore, studying the read/write conflict and timing error problems is important in the application of high-speed, high-bandwidth scenarios to processors.

3. 5R4W High-Timing Reliability Regfile Circuit Implementation

The Regfile circuit structure mainly includes the memory array, address module and control module, as shown in Figure 3. The memory array consists of an array of memory cells, as well as a data path composed of bit lines and a switch path composed of word lines; the address module consists of a decoding circuit; the control module contains each module clock control signal and drive buffer, with the module controlling the address signal transfer decoding circuit process, data signal input data latch circuit process and circuit output and other processes. Address information is input from the address module for decoding, and data is passed from the input control module into the data latch circuit, which is controlled by the timing control circuit for transmission to the storage array for storage, and the output is read out from the output circuit after precharging is completed.

3.1. Memory Cell Circuit. With the progress of the process, the leakage current problem is becoming more and more serious, especially for the 7-nm process, as the leakage current power consumption has become a major problem in the power consumption. Therefore, this design improves the circuit structure of the 8T memory cell by adding two transistors to reduce the leakage current, with the cell structure shown in Figure 4. This structure has two more PMOS tubes, P2 and P4, to reduce the leakage current consumption compared to the traditional 8T cell.

The N6 transistor acts as an isolator to completely separate the read circuit from the memory circuit, which reduces "read flip" and causes voltage fluctuations when reading and

does not interfere with the data stored inside the circuit. By choosing the single-port read method, modules such as sensitivity amplifiers can be eliminated compared to the dual-port read, and the area can be greatly reduced by adding the isolation tubes.

Leakage power consumption becomes very important at the 7-nm process node, and the use of MOS tube stacking technology effectively reduces leakage power. The test results compared with the conventional 8T structure are shown in Figure 5. The power supply voltage is scanned from 0.4 to 1.5 V during the test, and the power consumption saves up to 47.99% and down to 38.54%, with a maximum power consumption saving of 532.2 nW. This circuit is rated at 750 mV, with an average power consumption saving of over 40% around the rated operating voltage.

Meanwhile, as the process node increases, the supply voltage becomes lower and lower, and the use of additional circuit structures reduces the noise tolerance of the circuit. This paper compares the static noise margin (SNM) of a conventional circuit with this design in the same state. As shown in Figure 6(a), the largest square between the curves represents the SNM, the solid line represents the SNM curve of this design circuit, and the dashed line represents the SNM curve of the conventional circuit, which has very little effect on the stability when compared to the power saving of more than 40%. Figure 6(b) shows the hold noise margin (HNM), and the memory cell circuit test result is the noise tolerance of the circuit holding state, which is consistent with the SNM test results.

3.2. Read/Write Timing Separation Technology. Typically, Regfile circuits use clock-rising edges as the trigger signal, defining two clock-rising edges as one clock cycle, and performing read and write operations in that time region. However, this will result in the trigger read and write signals being the same clock edge. In the multiport Regfile circuit, the read port decode, read, output operation, and the write port decode, write operation, will simultaneously lead to read and write conflict phenomenon.

Combined with the double-edge flip–flop operating principle, the multiport Regfile read and write timing separation technology was proposed using the rising/falling edge of the clock to control the read and write operations, respectively, effectively improving the circuit performance, while solving the problem of read and write conflicts. In the read/write timing separation technology, the rising edge of the clock was chosen to be used as the read operation trigger signal and the falling edge as the write operation trigger signal, as shown in Figure 7.

The operational steps of the time-series separation technique are as follows:

The decoding of the write address signal after the arrival of the falling edge causes the corresponding WWL signal of the memory array to become high. The corresponding WBL signal becomes high after the input latching of the data is written.

After the WWL signal and WBL signal are ready, the data are written while the read address is synchronously latched
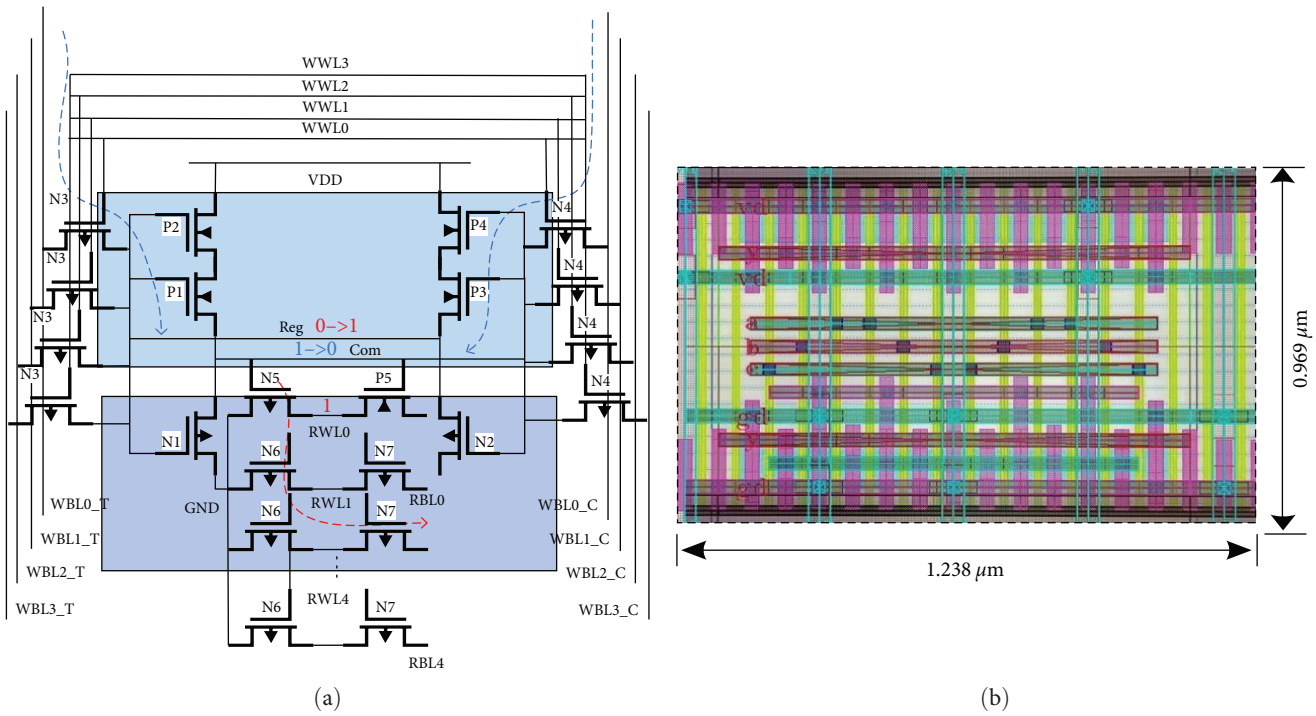
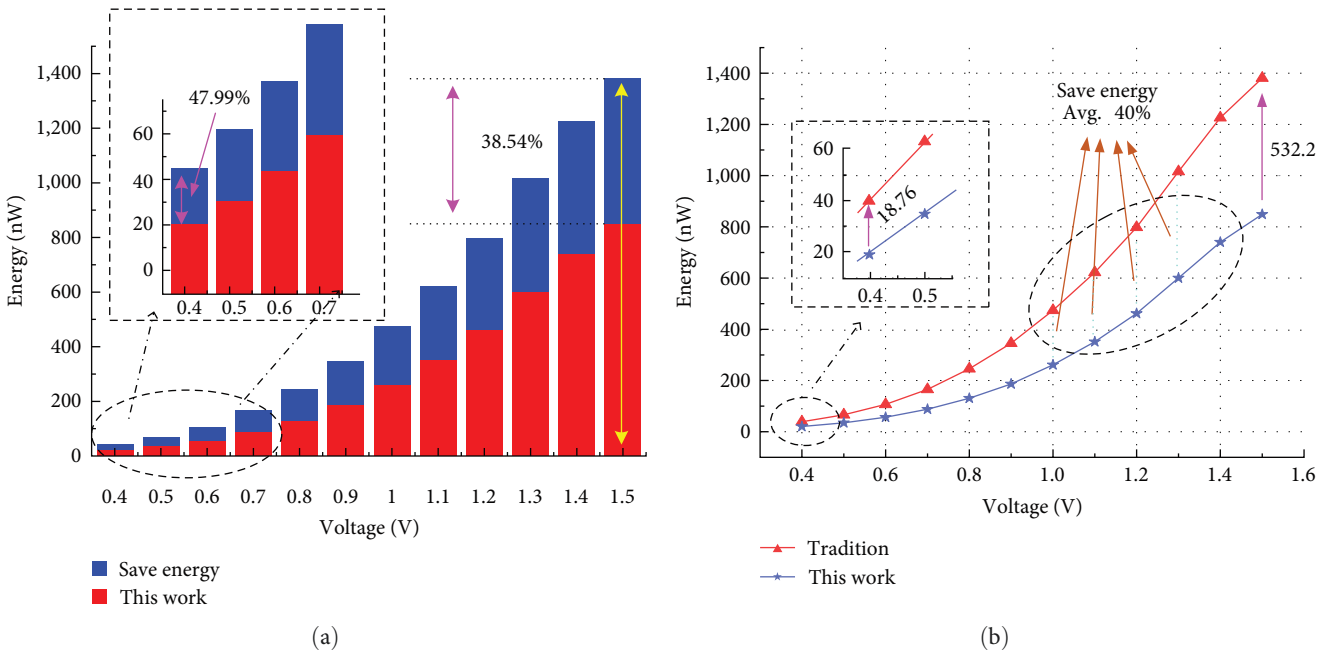FIGURE 4: Storage units: (a) memory cell circuit and (b) memory cell layout.

FIGURE 5: Power consumption analysis: (a) power consumption comparison chart and (b) power consumption percentage chart.

through the input control module, and the precharge module charges the bit line and pulls up the bit line signal in preparation for reading.

The clock signal rising edge comes after the write operation has been completed, and the cell storage data are updated to the latest data. The read address data after latching are transmitted to the decoding module for decoding, and the translated RWL signal becomes high for the read operation, and the voltage condition of the read bit line is judged for data output.

When a write operation is performed in the same cycle, only the read operation is prepared and no data are read.
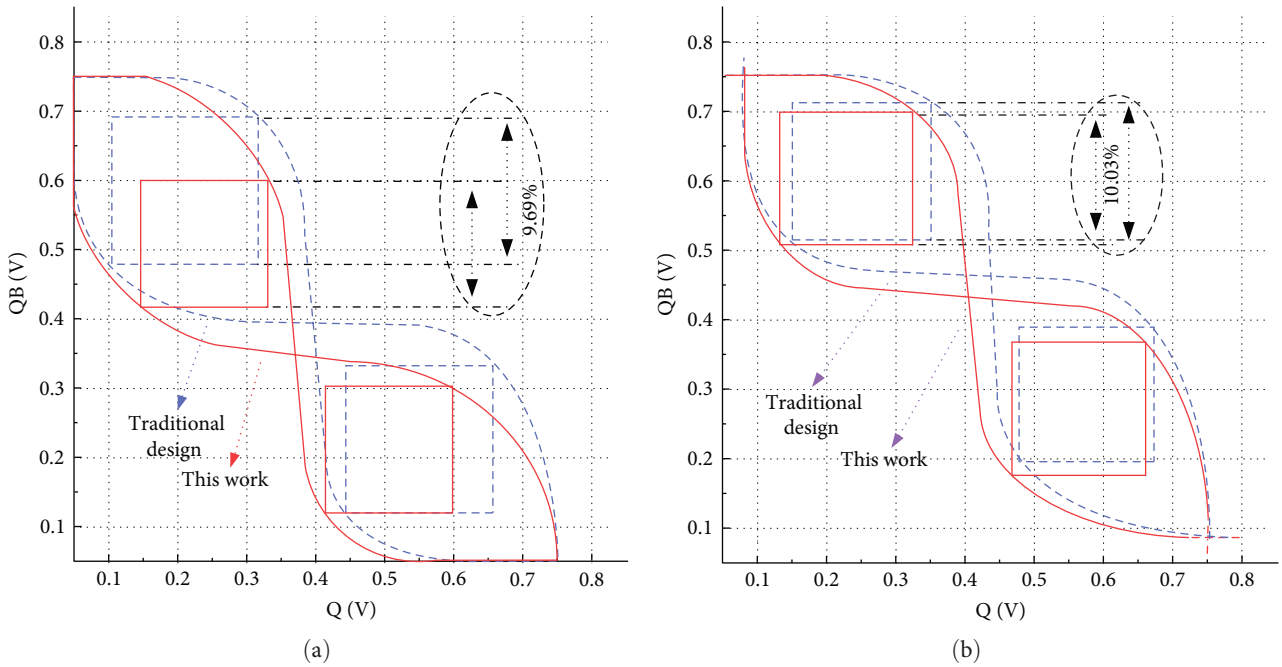
FIGURE 6: Noise tolerance comparison chart: (a) SNM comparison chart and (b) HNM comparison chart.
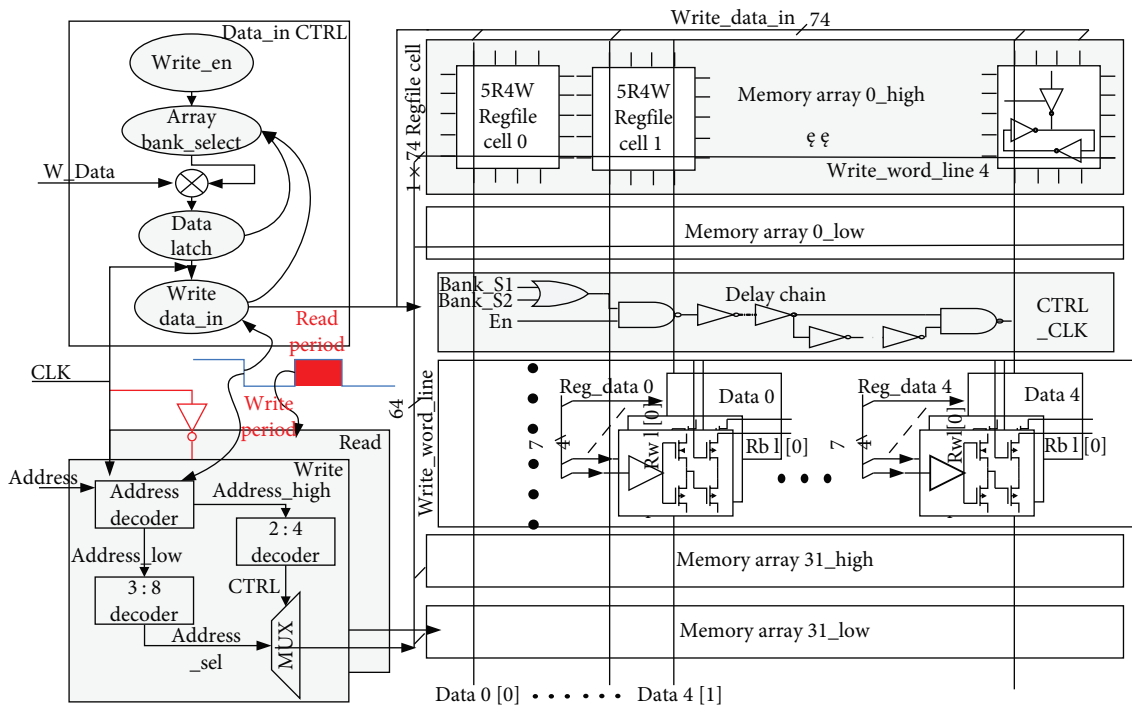

FIGURE 7: Schematic of read/write separation technology.

When performing the read operation, the first half cycle of the write operation does not begin, and the second half cycle begins writing after the process of reading from the storage unit has been completed; meanwhile, the read operation is the subsequent output process that does not involve the storage unit. Therefore, read and write are carried out in one cycle, in fact, the timing of the two are independent and will not produce read and write conflict problems.

3.3. Single-Word Line Error Detection Technology.

During the Regfile layout full custom design process, it was found that differences in the design layout resulted in different

signal arrival times for different word lines, which led to writing errors due to excessive delay in the metal lines. In high-speed operation, the same data written to different word lines "long word line" delay were greater to make the written data confusing, resulting in reading errors. Therefore, this paper proposes a single-word line error detection technique for the above write error problem. Usually, the "farthest" word line has a much higher chance of writing errors than other word lines, and if the writing error caused by the delay on the line occurs at the near end of the word line, then the far end of the word line must also occur, which is why only the "farthest" word line error is checked for the "farthest" word line to alert whether the write is correct, and the area overhead of replicating the array for only one-word line is within acceptable limits. This technique uses the replicated array method to check only for data errors on the "farthest" word line.

The 64×74-bit Regfile cell arrays are divided into two arrays <0 : 31> and <32 : 63> using the split storage technology, and each storage array has 32-word lines. According to the layout design, the input data control module is located in the center of the layout, and the word lines of the left memory cell array are arranged in <63 : 32> order, while the word lines of the right memory cell array are arranged in <0 : 31> order, and the two memory arrays are symmetrical from left to right. Therefore, WL <31> and WL <63> are the two longest word lines with the lowest timing reliability. Two error-checking memory arrays are added nearest to each side of the data control module to which WL <31> and WL <63> are connected. The memory array copies are located closest to the center of the layout to ensure the correctness of the data stored in them, and the structure is shown in Figure 8(a).

The same data are written to both WL <31> and WL <63>, while the same data are written to the memory array copy, and the data are compared to the two when read out; however, if they are the same, the data are output and the error detection output port is low. Conversely, the output level of the error detection output port is high and the data at the farthest end of the word line is incorrect, which means that at least one-word line is incorrect in the writing process, and other operations such as writing back need to be judged. The function implementation is illustrated in Figure 8(b), with the line delay causing an error to occur in the write data when the two storage arrays store different content. Therefore, when reading out the error port, a high level will be output to indicate that this data write a read error, so as to complete the error detection process.

3.4. Precharge Phase-Locked Clock Control Circuit. The control module is mainly responsible for controlling the signal input and output to meet the functional and timing requirements of the circuit by controlling the sequence of arrival of different signals, so that the circuit can work correctly according to the expected output. Figure 9 shows the precharge module, which precharges the voltage on the output line; that is, "charging" the change of the bit line voltage during the reading process. Figure 9 shows the precharge module, which prebuilds up the voltage on the output line, i.e., "charging" to prepare for the change of the bit line voltage during the reading process.

Compared with the traditional precharge circuit, the precharge module of this design adds a phase-locked clock signal CTRL_CLK to the circuit to keep the output signal stable and prevent the "0" signal from being eaten up by the "1" signal. The feedback module consisting of MOS tubes is added at the upper and lower ends to speed up the charging speed when charging the line, reduce the leakage current by stacking multiple MOS tubes, and reduce the output error caused by voltage fluctuations on the LBL_T and LBL_B lines. The output is guaranteed to be correct by judging the voltage on the D line feeding back to the T5 tube.

In the initial state, PCH_T and PCH_B two port signals are low, opening T1 and T2 tubes to charge LBL_T and LBL_B lines, when the voltage on both lines are VDD. Control CTRL_CLK was made low to open the T15 tube and T5 tube via the control module, and D line voltage dropped rapidly to GND via T3, T4, and T5 three transistors. The output OUT of the module keeps the output data "1" in the default state.

In the read state, when the read data is "1", the LBL_T and LBL_B lines remain high and the output does not change at this time. If one of the upper and lower memory arrays reads "0," the voltage on the corresponding line will be pulled down. After the voltage is pulled down to the PMOS tube turn-on voltage, the MOS tube turns on and charges D.

The output OUT becomes "0" and the feedback circuit uses the T14 tube to maintain the voltage on the D line. After the reading is completed, the read port is closed and the precharge signal is turned on to pull the D line voltage down again to prepare for the next reading. The phase-locked clock signal is generated by the enable signal and the block selection signal. The role of the clock is to maintain the n clock cycle output signal to the $n + 1$ CLK signal rising edge of the change arrival, and the output signal will be maintained long enough to prevent misreading by the other modules. If the fluctuation or precharge signal are wrongly turned on, then reading data "0" when reading the word line will lead to the arrival of a high-level momentarily opening of the T4 tube, so that the D line voltage is pulled down to "0," and the output generates a flip read error. The phase-locked clock structure can control the circuit to maintain the output within the specified time, and the D line cannot pass through to GND output error data, so this structure can greatly improve the Regfile circuit timing reliability.

3.5. Predriven Decoding Circuit. Regfile circuits typically add enable drivers to word lines after decoding is complete to control their output timing and ensure proper circuit function. This requires adding additional drivers to each word line to allow the enable signal to control it, which generates a significant additional area overhead. In this paper, we propose a new decoding architecture that adds enable and clock information to the address signal before decoding, and performs static decoding of the address signal at the backend only to complete word line decoding and timing control. The
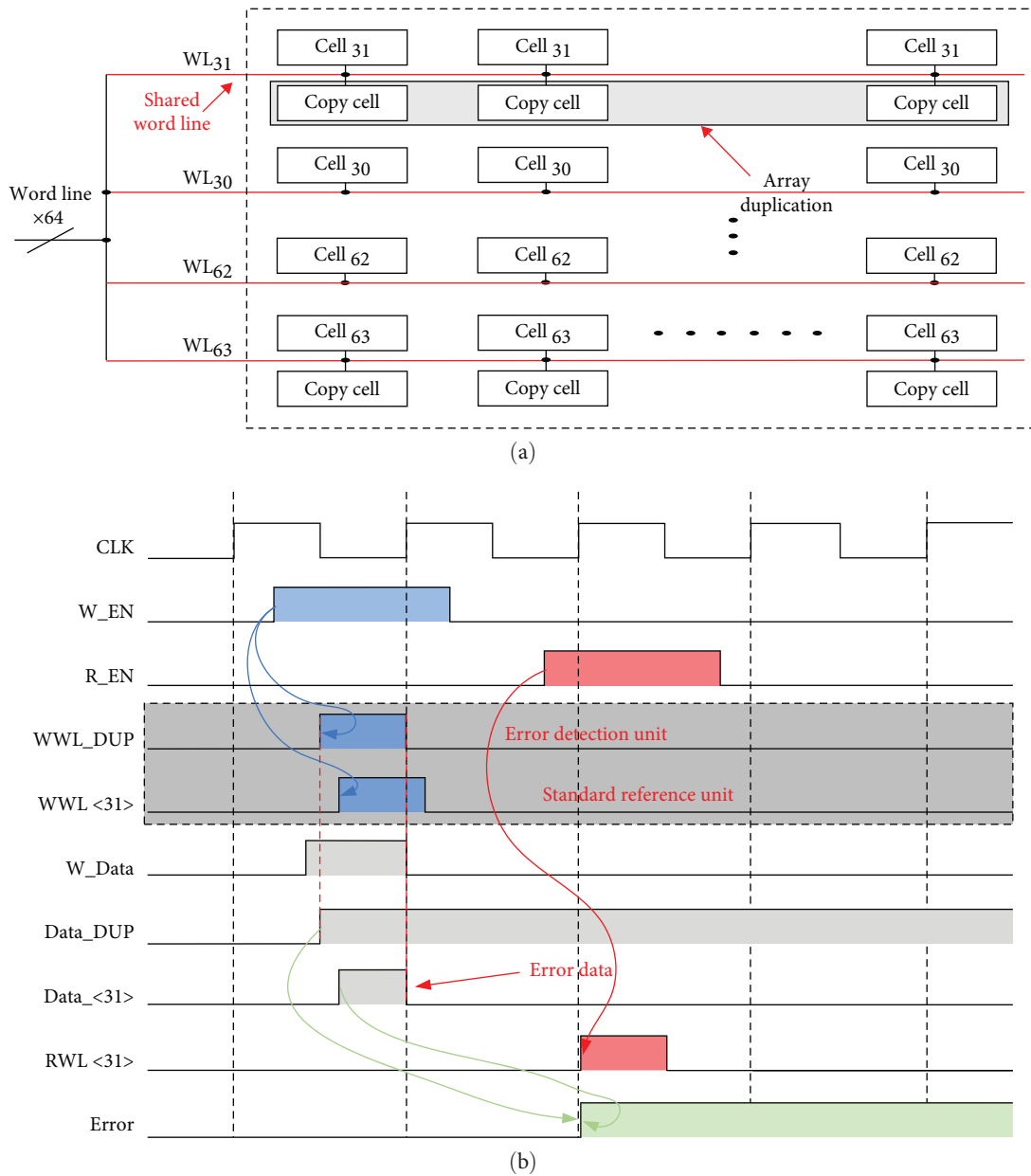
FIGURE 8: Single-word line error detection technique: (a) single word line error detection structure and (b) single word line error detection implementation.

entire decoding process is divided into four parts, first, the address information is latched to ensure the address information is stable and prevent the address information from changing during the decoding process, resulting in a word line selection error. Alternatively, the data information changes during the writing process, resulting in the bit line information confusion problem. After data latching, the clock is waited for and the selected clock edge arrives with the clock signal, enable signal and address signal fuzed through the predecoding circuit to output in-phase address information and inverse address information. This step allows the subsequent static decoding process to include both the timing information and the automatic shutdown of the decoding process based on the enable information contained in the signal, thereby eliminating the

need for an additional enable driver unit to control the word line output. After completing the precoding, the information is processed by the main decoding circuit to obtain the selected word line. The two-stage static decoding in the main decoding circuit is chosen to balance performance and power consumption, which can reduce the high dynamic power consumption caused by high-flip–flop rate and avoid the large area overhead problem caused by single-stage decoding.

The decoding function is illustrated in Figure 10(b), where the address information changes when the clock is high and does not affect the output result of the word line. Since the address information is only sampled and latched at the clock edge and held until the end of that CLK, the problem of false flip–flops caused by voltage fluctuations or
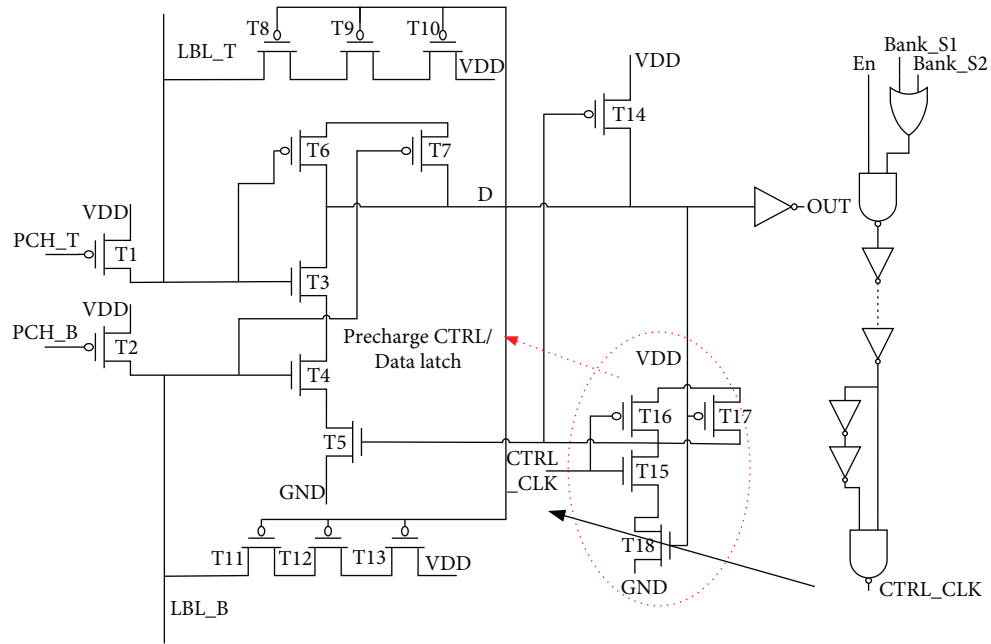
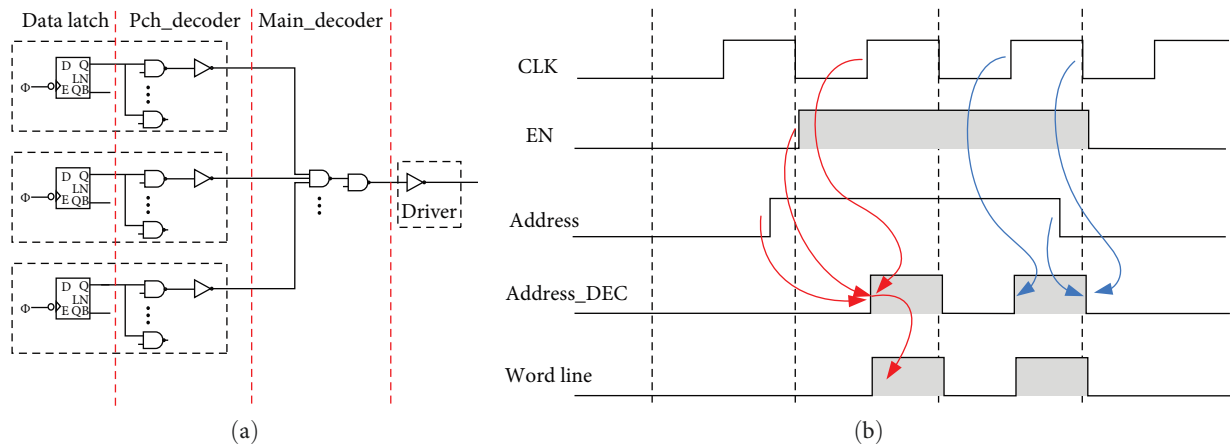FIGURE 9: Precharge design with phase-locked clock structure.

(a) (b)

FIGURE 10: Schematic of the preenablement decoding circuit: (a) architecture of the decoding circuit and (b) schematic diagram of the decoding function.

incorrect inputs is reduced. The word line output has both clock information and enable information to ensure correct results.

4. Results and Analysis

The circuit was designed in TSMC's 7-nm CMOS process using Cadence Virtuoso ICADVM20 and then verified using Specter for functional simulation, Layout XL platform for layout design, and Mentor's Caliber tool for Design Rule. The design was verified by the Layout XL platform for layout design and Mentor's Caliber tool for Design Rule Check (DRC) and Layout Versus Schematics (LVS). After passing the verification, the circuit was analyzed for electromigration (EM) and supply voltage drop (IR drop) using the Cadence Voltus-Fi tool. The design was tested using different process

voltage temperature (PVT) environments and postsimulated using different RC extraction models during simulation verification, with the specific PVT settings shown in the results of Table 1.

The layout design was implemented based on the 7-nm FinFET process, and the layout was designed using a fully customized approach, as shown in Figure 11(a). The circuit diagram corresponds to two memory arrays, each consisting of 37 vertically arranged memory cells. The two memory cell arrays were connected in the middle with metal wires and drive circuits etc. The read/write control circuit was placed in the middle of the layout to ensure that the control signals reach the memory cells at an equal distance to avoid problems such as line lengths that cause excessive delays and circuit operation errors. The decoding module was placed in the middle of the memory array to ensure that the signal

TABLE 1: Simulation PVT environment table.

Process	Voltage	Temperature	RC condition
FF	0.825	125	CC best
SS	0.675	−40	CC worst
TT	0.75	85	Typical
FF	0.935	125	CC best
SS	0.765	−40	CC worst
TT	0.85	85	Typical
FF	0.825	−40	CC best
SS	0.675	125	CC worst
FF	0.935	−40	CC best
SS	0.765	125	CC worst
FF	1.02	125	CC best
FF	1.02	−40	CC best
FF	1.02	125	CC worst
FF	1.02	−40	CC worst
SS	1.02	125	CC best
SS	1.02	−40	CC best
SS	1.02	125	CC worst
SS	1.02	−40	CC worst
FF	0.585	125	CC best
FF	0.585	−40	CC best
FF	0.585	125	CC worst
FF	0.585	−40	CC worst
SS	0.585	125	CC best
SS	0.585	−40	CC best
SS	0.585	125	CC worst
SS	0.585	−40	CC worst

Technology	7-nmFinFET CMOS
Bitcell read/write ports	5R4W
Bitcell area	9051.42 μm^2
Register file area	13118 μm^2
Fmax	3.8 GHz
Power	5.541 mW
Read band width	19G Words/s
Write band width	15.2G Words/s

(a)   (b)
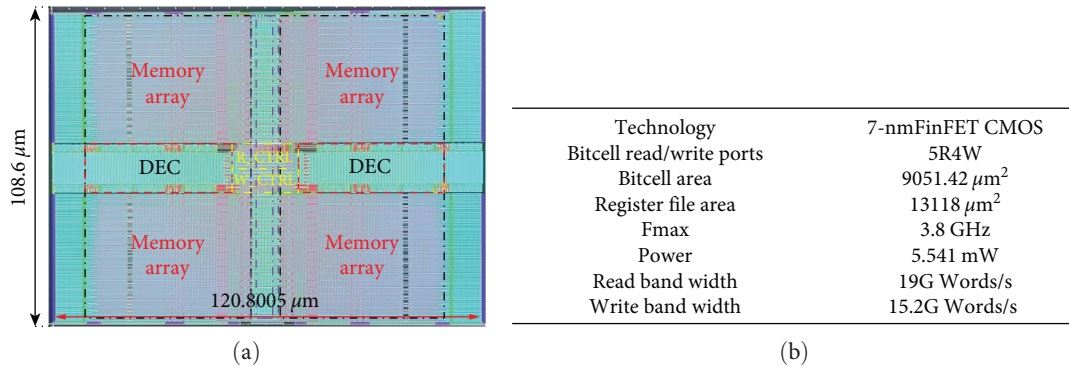
FIGURE 11: Regfile circuit implementation: (a) Regfile circuit layout and (b) feature parameter map.

reaches each memory cell in the shortest possible time and to decode the input and output data. The output input port control module was placed in the middle of the layout to connect to each cell port.

The Regfile layout area is 13,118 μm^2. Key parameters are shown in Figure 11(b) with five read ports and four write ports, a memory cell area of 9051.42 μm^2, and a power consumption of 5.541 mW to operate normally at frequencies up to 3.8 GHz. The single-channel throughput of the circuit proposed in this paper can reach 35.15 G Bytes/s. According to the calculation, this design has a read bandwidth of 19 G Words/s and a write bandwidth of 15.2 G Words/s.

The bandwidth calculation equation is as follows.

$$BW = Work\ freq \times Data\ bits. \quad (1)$$

The area and power consumption shares are shown in Figure 12, where the memory cell array accounts for 69% of the total area, the decoding module accounts for 16%, and the other module parts account for a total of 15%. The power consumption of the memory circuit is 47%, the decoding module consumption is 10%, and the control module accounts for 28% of the power consumption overhead due to its high
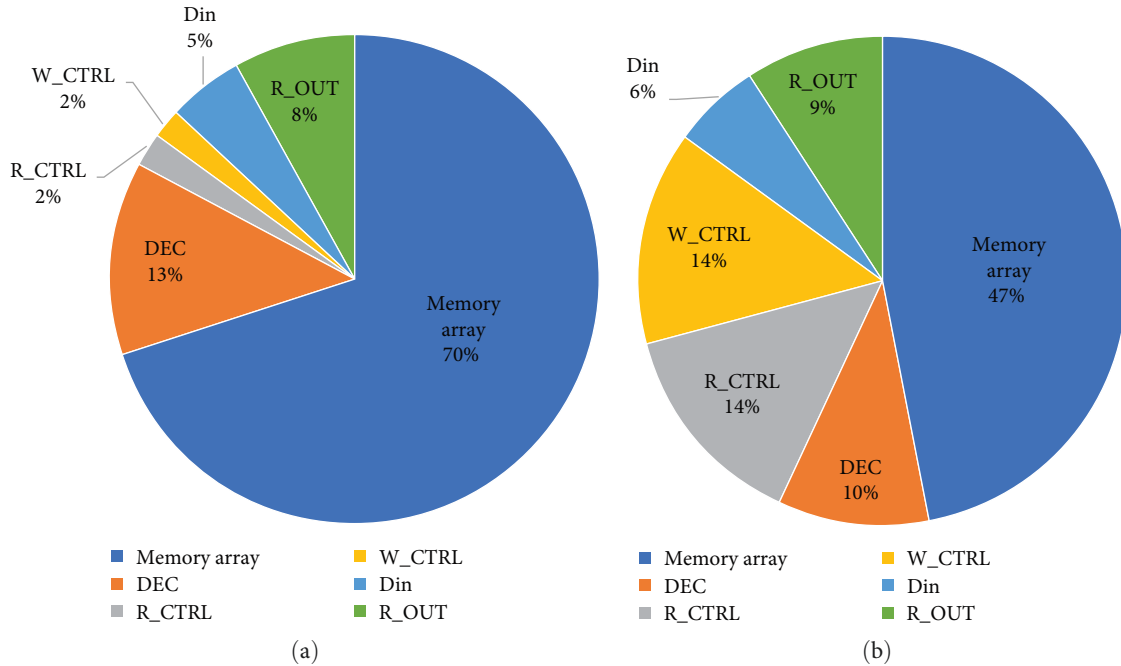
FIGURE 12: Area vs. power consumption ratio chart: (a) area share diagram and (b) power consumption share diagram.
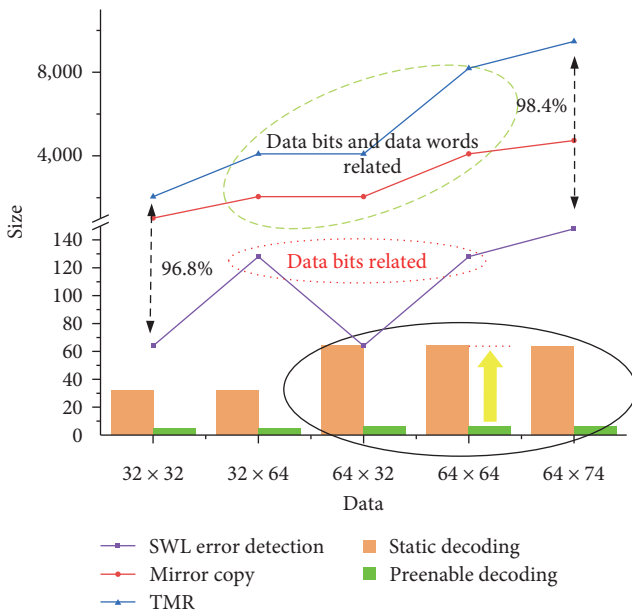

FIGURE 13: Preenabled decoding and single-word line error detection area saving.

activity in the case of 2% of the area, for a total circuit power consumption of 5.541 mW.

The single-word line error detection structure was chosen to compare with the three-mode redundancy structure and the mirror copy structure, and the preenablement decoding circuit was compared with the conventional static decoding circuit, and the results are shown in Figure 13. Five types of data bits were selected: 32×32, 32×64, 64×32, 64×64, and 64×74. The first four are commonly used

data formats, and the fifth is the data format used in this circuit. The single-word line error detection structure can save 96.8%–98.4% area compared to triple-mode redundancy. The area of this design circuit remains the same after storing data from 32×64 to 64×64, but the area of the triple-mode redundancy increases by 100%. The preenabled decoding circuit can reduce the area overhead by 87.5% compared with the traditional static decoding circuit, and the area of this structure grows linearly with the number of data words, while the area of the traditional static decoding circuit grows exponentially, and the area of the traditional static decoding circuit increases by 100% after the stored data grow from 32×64 to 64×64, while the area of the preenabled decoding circuit increases by only 20%.

Comparing the same circuits implemented in TSMC's 65-nm process, the results are shown in the Figure 14. The memory cell saves over 60% power, the precharge circuit saves 24.75% power, and the control and decoder modules save 50.81% power. The 7-nm process is a huge improvement in power saving.

The relevant performance comparison is shown in Figure 15, where this design operates at a minimum speed improvement of 0.25 times compared to other literature, as shown in Figure 15(a). The area saving is at most 71.5%, and the present design also saves 30.9% of area compared to the low speed, low-area overhead design as shown in Figure 15(b).

The comparative analysis with related literature is shown in Table 2. From the table, it can be seen that the present design has a very significant advantage in terms of speed area compared to the low-process node Regfile. The power consumption saving is 58% compared to the literature by Wu et al. [22], and literature by Suzuki et al. [23] and Li et al. [30] significantly reduces the power consumption, but
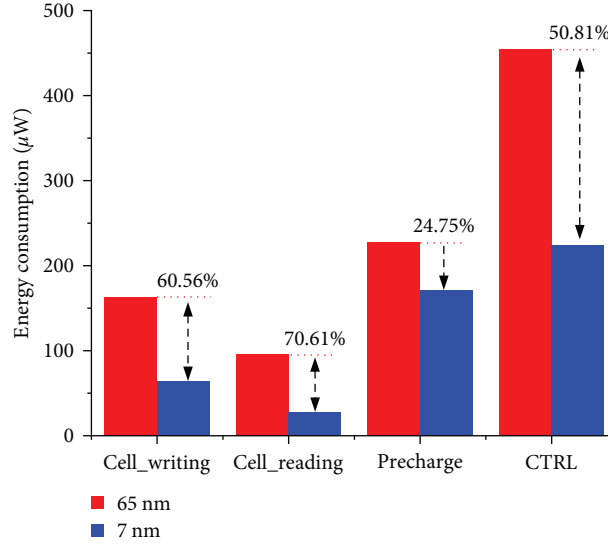
FIGURE 14: Comparison of power consumption of different processes.
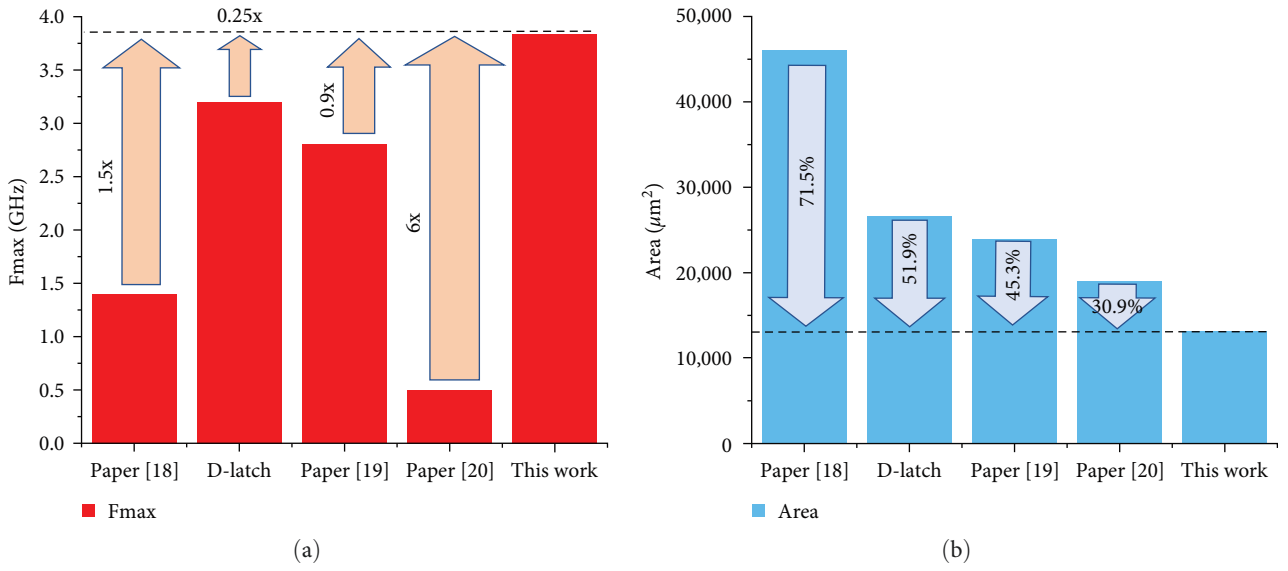

(a)

(b)

FIGURE 15: Speed and area comparison: (a) operating speed comparison and (b) circuit area comparison.

TABLE 2: Comparative analysis table with related literature.

Paper	Tech (nm)	Array size	Throughput (G Bytes/s)	Frequency (GHz)	Energy (mW)	Area (mm²)	Port
[27]	180	16 × 2	0.125	0.5	46	1.9	10R6W
[28]	180	32 × 8	—	—	4.538	—	4R2W
[29]	180	32 × 32	—	—	—	—	6R2W
[10]	130	32 × 32	2.664	0.666	26.9	—	2R1W
[30]	45	64 × 64	8	1	13.2	—	8R4W
[31]	40	16 × 36	4.5	1	1.17	0.25160	7R6W
[32]	40	32 × 64	8	1	1.50	0.08821	3R2W
[33]	28	64 × 32	13.2	3.3	20.94	0.18900	6R4W
[34]	28	32 × 128	44.8	2.8		0.23980	4R2W
[16]	7	128 × 32	10.36	2.59	117	8.052	3R3W
This work	**7**	**64 × 74**	**35.15**	**3.8**	**5.541**	**0.13**	**5R4W**

Bold value represents the work indicators.

the present design increases the operating frequency to 3.8 times of the original and has a larger data capacity. The literature by Manivannan and Srinivasan [28] uses a 28-nm process design, which is similar to the present design in terms of array size, number of ports, and operating speed, and the present design saves 81.91% in terms of power consumption. The design in the literature by Nguyen et al. [16] uses a 7-nm process design with dual-pumping technology to adjust the operating frequency and bandwidth to the machine learning chip, and the design saves 99% energy overhead and 99% area overhead in terms of power consumption and area. We calculated the single-channel throughput of this circuit as 35.15 G Bytes/s, demonstrating superior performance over other designs. Although paper by Kadomoto et al. [34] achieves slightly higher single-channel throughput, our design provides more bandwidth and ports, leading to higher overall throughput.

5. Conclusion

This paper implements the 5R4W high-timing reliability Regfile circuit. The read/write separation technology was used to solve the read/write conflict problem; the single-word line error detection technology was used to solve the line delay write error; the phase-locked clock feedback structure was used to maintain the read output stability. The circuit was designed and tested under TSMC's 7-nm process. The circuit area is $0.13 \, mm^2$ and the power consumption is 5.541 mW. The maximum operating frequency is 3.8 GHz, and it can detect write errors caused by a clock jitter exceeding 30 ps or a frequency exceeding 5 GHz. The single-word line error detection technique saves up to 98.4% area than the triple-mode redundancy technique, and the preenable decoding circuit saves 87.5% area than the conventional static decoding circuit. Compared with related literature its timing reliability is high and its advantages in terms of power consumption and area are obvious for high-performance processors such as machine learning and AI chips.

Data Availability

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy restrictions.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Wanlong Zhao contributed in writing the original draft, resources, data curation. Yuejun Zhang contributed in the conceptualization, validation, and writing review. Liang Wen contributed in the methodology and investigation. Pengjun Wang contributed in the software and supervision.

Acknowledgments

References

[1] H. Momose, T. Kaneko, and T. Asai, "Systems and circuits for AI chips and their trends," *Japanese Journal of Applied Physics*, vol. 59, Article ID 050502, 2020.

[2] B. Li, J. Gu, and W. Jiang, "Artificial intelligence (AI) chip technology review," in *2019 IEEE International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, pp. 114–117, IEEE, Taiyuan, China, November 2019.

[3] T. Ze, Z. Jun, R. Xianglong, F. Feihu, and C. Yue, "Survey of shared register file design for unified shader array in GPUs," in *2022 IEEE 9th International Conference on Cyber Security and Cloud Computing (CSCloud)/IEEE 8th International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pp. 201–206, IEEE, Xi'an, China, June 2022.

[4] S. Borkar and A. A. Chien, "The future of microprocessors," *Communications of the ACM*, vol. 54, no. 5, pp. 67–77, 2011.

[5] G. Blake, R. G. Dreslinski, and T. Mudge, "A survey of multicore processors," *IEEE Signal Processing Magazine*, vol. 26, no. 6, pp. 26–37, 2009.

[6] L. Lin-Dong, Q. De-Yu, C. Qiang, and R. Jin-Xin, "Efficient scheduling mechanism for performance-heterogeneous multi-core processor," in *2014 5th International Conference on Digital Home*, pp. 342–346, IEEE, Guangzhou, China, November 2014.

[7] S. Saini, H. Jin, R. Hood, D. Barker, P. Mehrotra, and R. Biswas, "The impact of hyper-threading on processor resource utilization in production applications," in *2011 IEEE 18th International Conference on High Performance Computing*, pp. 1–10, IEEE, Bangalore, India, December 2011.

[8] D. T. Marr, F. Binns, D. L. Hill et al., "Hyper-threading technology architecture and microarchitecture," *Intel Technology Journal*, vol. 6, no. 1, pp. 1–12, 2002.

[9] T. Leng, R. Ali, J. Hsieh, V. Mashayekhi, and R. Rooholamini, "An empirical study of hyper-threading in high performance computing clusters," *Linux HPC Revolution*, Article ID 45, 2002.

[10] T.-S. Jau, W.-B. Yang, and C.-Y. Chang, "Analysis and design of high performance, low power multiple ports register files," in *IEEE Asia Pacific Conference on Circuits and Systems*, pp. 1453–1456, IEEE, Singapore, December 2006.

[11] N. Tsuji, Y. Yamanashi, N. Takeuchi, C. Ayala, and N. Yoshikawa, "Design and implementation of scalable register files using adiabatic quantum flux parametron logic," in *2017 IEEE 16th International Superconductive Electronics Conference (ISEC)*, pp. 1–3, IEEE, Naples, Italy, June 2017.

[12] O. Chen, R. Saito, T. Tanaka, C. L. Ayala, N. Takeuchi, and N. Yoshikawa, "Design of adiabatic quantum-flux-parametron register files using a top–down design flow," *IEEE Transactions on Applied Superconductivity*, vol. 29, no. 5, pp. 1–5, 2019.

[13] N. Tsuji, C. L. Ayala, N. Takeuchi, T. Ortlepp, Y. Yamanashi, and N. Yoshikawa, "Design and implementation of a 16-word by 1-bit register file using adiabatic quantum flux parametron logic," *IEEE Transactions on Applied Superconductivity*, vol. 27, no. 4, pp. 1–4, 2017.

[14] X. Liu, S. Raman, R. Clarke et al., "Design of high-speed register files using SiGe HBT BiCMOS technology," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 61, no. 3, pp. 178–182, 2014.

[15] H. Nguyen, J. Jeong, F. Atallah, D. Yingling, and K. Bowman, "A 7-nm 6R6W register file with double-pumped read and write operations for high-bandwidth memory in machine learning and CPU processors," *IEEE Solid-State Circuits Letters*, vol. 1, no. 12, pp. 225–228, 2018.

[16] H. Nguyen, J. Jeong, F. Atallah et al., "A 7NM double-pumped 6R6W register file for machine learning memory," in *2018 IEEE Symposium on VLSI Circuits*, pp. 1-2, IEEE, Honolulu, HI, USA, June 2018.

[17] H. E. Yantir, S. Bayar, and A. Yurdakul, "Efficient implementations of multi-pumped multi-port register files in FPGAs," in *2013 IEEE Euromicro Conference on Digital System Design*, pp. 185–192, IEEE, Los Alamitos, CA, USA, September 2013.

[18] Q. Chen, L. Wu, L. Li, X. Ma, and X. A. Wang, "Enhanced reliability scheduling method for the data in register file," in *2015 IEEE 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, pp. 188–193, IEEE, Krakow, Poland, November 2015.

[19] H. Yue, X. Wei, J. Tan, N. Jiang, and M. Qiu, "Eff-ECC: protecting GPGPUs register file with a unified energy-efficient ECC mechanism," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 7, pp. 2080–2093, 2022.

[20] H. Ahangari, I. Alouani, O. Ozturk, S. Niar, and A. Rivenq, "Register file reliability enhancement through adjacent narrow-width exploitation," in *2016 IEEE International Conference on Design and Technology of Integrated Systems in Nanoscale Era (DTIS)*, pp. 1–4, IEEE, Istanbul, Turkey, April 2016.

[21] H. Jie, W. Shuai, and S. G. Ziavras, "In-register duplication: exploiting narrow-width value for improving register file reliability," in *IEEE International Conference on Dependable Systems and Networks (DSN'06)*, pp. 281–290, IEEE, Philadelphia, PA, USA, June 2006.

[22] J.-J. Wu, M.-F. Chang, S.-W. Lu, R. Lo, and Q. Li, "A 45-nm dual-port SRAM utilizing write-assist cells against simultaneous access disturbances," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 59, no. 11, pp. 790–794, 2012.

[23] T. Suzuki, H. Yamauchi, Y. Yamagami, K. Satomi, and H. Akamatsu, "A stable 2-Port SRAM cell design against simultaneously read/write-disturbed accesses," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 9, pp. 2109–2119, 2008.

[24] H. Fujiwara, C.-Y. Lin, H.-Y. Pan et al., "24.2 A 7nm 2.1GHz dual-port SRAM with WL-RC optimization and dummy-read-recovery circuitry to mitigate read- disturb-write issue," in *2019 IEEE International Solid-State Circuits Conference(-ISSCC)*, pp. 390–392, IEEE, San Francisco, CA, USA, February 2019.

[25] C. Constantinescu, "Trends and challenges in VLSI circuit reliability," *IEEE Micro*, vol. 23, no. 4, pp. 14–19, 2003.

[26] S. Valadimas, Y. Tsiatouhas, and A. Arapoyanni, "Timing error tolerance in nanometer ICs," in *2010 IEEE 16th International On-Line Testing Symposium, Corfu*, pp. 283–288, IEEE, Corfu, Greece, July 2010.

[27] Q. Yu, D.-H. Wang, T.-J. Zhang, and C.-H. Hou, "A design of 500MHz 10-read 6-write register file," in *2005 IEEE 6th International Conference on ASIC*, pp. 266–269, IEEE, Shanghai, China, October 2005.

[28] T. S. Manivannan and M. Srinivasan, "A 4-READ 2-WRITE multi-port register file design using pulsed-latches," in *2018 IEEE Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pp. 262–267, IEEE, Coimbatore, India, March 2018.

[29] X. Zhang and Z.-L. Li, "Full-custom implementation of 6-read 2-write multi-port register file," *Computer Engineering*, vol. 288, no. 20, pp. 248–250, 2007.

[30] J. Li, L. H. Wang, Z. Bi, and P. Liu, "A 1 GHz multi-port low-power register file design," *Computer Engineering & Science*, vol. 37, no. 12, pp. 2222–2227, 2015.

[31] Q. P. Zhang, Z. T. Li, and Y. Liu, "40nm process multi-port register file design," in *Proceedings of the 21th National Conference on Computer Engineering and Technology and the 7th Microprocessor Forum*, pp. 174–179, Xiamen, China, 2017.

[32] G. J. Yuan, H. Shen, E. Shao, and D. W. Zang, "Low-latency register file based on adaptive timing matching," *Chinese High Technology Letters*, vol. 28, no. 2, pp. 91–99, 2018.

[33] H. Hsieh, S. H. Dhong, C.-C. Lin et al., "Custom 6-R, 2- or 4-W multi-port register files in an ASIC SOC with a DVFS window of 0.5 V, 130 MHz to 0.96 V, 3.2 GHz in a 28-nm HKMG CMOS technology," in *2015 IEEE Custom Integrated Circuits Conference (CICC)*, pp. 1–3, IEEE, San Jose, CA, USA, September 2015.

[34] J. Kadomoto, H. Irie, and S. Sakai, "Multiport register file design for high-performance embedded cores," in *2021 IEEE 14th International Symposium on Embedded Multicore/Many-Core Systems-on-Chip (MCSoC)*, pp. 281–286, IEEE, Singapore, Singapore, December 2021.