Compact Non-Volatile Lookup Table Architecture based on Ferroelectric FET Array through In-Situ Combinatorial One-Hot Encoding for Reconfigurable Computing

Weikai Xu[1], Meng Li[2,1,3], Qianqian Huang[1,3*], and Ru Huang[1,3*]
[1]School of Integrated Circuits, Peking University, Beijing 100871, China; [2]Institute for Artificial Intelligence, Peking University, Beijing 100871, China; [3]Beijing Advanced Innovation Center for Integrated Circuits, Beijing 100871, China.
[*]Corresponding author: ruhuang@pku.edu.cn; hqq@pku.edu.cn

Abstract—**Lookup tables (LUTs) are widely used for reconfigurable computing applications due to the capability of implementing arbitrary logic functions. Various emerging non-volatile memories (eNVMs) have been introduced for LUT designs with reduced hardware cost and power consumption compared with conventional SRAM-based LUT. However, the existing designs still follow the conventional LUT architecture, where the memory cells are only used for storage of configuration bits, requiring dedicated bulky multiplexer (MUX) for computation of each LUT, resulting in inevitable high area, latency, and energy cost. In this work, a compact and efficient non-volatile LUT architecture based on ferroelectric FET (FeFET) array is proposed, where the configuration bit storage and computation can be implemented within the FeFET array through *in-situ* combinatorial one-hot encoding, eliminating the need of costly MUX for each LUT. Moreover, multibit LUTs can be efficiently implemented in the FeFET array using only one shared decoder instead of multiple costly MUXs. Due to the eliminated MUX in the calculation path, the proposed LUT can also achieve enhanced computation speed compared with the conventional LUTs. Based on the proposed LUT architecture, the input expansion of LUT, full adder, and content addressable memory are further implemented and demonstrated with reduced hardware and energy cost. Evaluation results show that the proposed FeFET array-based LUT architecture achieves 51.7×/8.3× reduction in area-energy-delay product compared with conventional SRAM-based/FeFET-based LUT architecture, indicating its great potential for reconfigurable computing applications.**

Index Terms—lookup table, ferroelectric FET, reconfigurable computing

I. Introduction

Lookup tables (LUTs) which can generally perform arbitrary logic functions and support a variety of arithmetic functions, have been widely used for reconfigurable computing applications such as field-programmable gate arrays (FPGAs), finite state machines and digital signal processing et al. [1-6], as well as intelligent applications such as deep neural networks and transformers [7-9]. In a conventional SRAM-based K-input LUT, 2^K static random-access memory (SRAM) cells are employed for storing the configuration bits of logic function, and a 2^K-to-1 multiplexer (MUX) is needed to select the output bit of a logic computation according to the input signals [2]. While SRAM-based LUT has been utilized commercially in various applications such as modern FPGAs [10], it suffers from high hardware cost and energy consumption. Additionally, due to the volatility of SRAM, it requires additional off-chip nonvolatile memory to store the configuration bits or consumes the static standby power [1-2].

In recent years, various emerging non-volatile memories (NVMs) with the potential for high storage density, such as resistive random-access memory (RRAM) [11-14], magnetic tunnel junction (MTJ) [15-17], and ferroelectric FET (FeFET) [18-20], have been actively investigated to construct non-volatile LUTs, which is a promising solution for reducing hardware cost and eliminating standby power dissipation. Generally, the existing NVM-based LUT designs still follow the conventional SRAM-based LUT architecture, where only the SRAM cells are replaced with the dense NVM cells for storing the configuration bits. They suffer from the following challenges: (1) each LUT still requires a dedicated MUX, whose hardware overhead is close to or even exceeds that of NVM-based configuration bits, limiting further improvements in area efficiency and energy efficiency of non-volatile LUTs; (2) multiple transistors in the MUX are connected in series within the calculation path of LUT, resulting in significantly increased computation delay; (3) multibit LUTs are generally required in practical applications, such as input expansion of LUT and implementation of arithmetic functions, where the dedicated bulky MUXs in conventional LUT architecture significantly increase area-energy-delay product (AEDP).

In this work, a compact non-volatile LUT architecture based on FeFET array with *in-situ* combinatorial one-hot encoding is proposed, which can address the challenges mentioned above and is demonstrated with the lowest hardware cost, energy consumption and latency. By utilizing both the storage and computation capabilities of FeFET, multibit LUTs can be implemented in the FeFET array of the proposed LUT architecture, without the need of multiple costly MUXs, thereby facilitating efficient mapping of LUT-based reconfigurable applications. The main contributions of this work are summarized as follows.

- A compact and efficient FeFET array-based non-volatile LUT architecture is proposed. By utilizing the non-volatility of FeFET with a combinatorial one-hot encoding scheme, the storage cells of configurable logic function can be implemented efficiently, along with eliminated standby power consumption. Besides, by further leveraging the gate controllable structure of FeFET for input signals, the logic computation of LUT can be *in-situ* implemented within the FeFET array, without the need of MUX, which leads to higher area efficiency and energy efficiency. Moreover, by eliminating the MUX in the

978-3-9826741-0-0/DATE25/© 2025 EDAA

calculation path, the speed of LUT is also improved.

- The proposed FeFET array-based LUT architecture can efficiently implement multibit LUTs with one shared decoder, instead of multiple costly MUXs in the conventional LUT architecture. It contributes to the efficient mapping of practical applications, where multibit LUTs are generally required, such as input expansion of LUT and implementation of arithmetic functions (e.g., full adder and multiplier). Moreover, the proposed LUT architecture enables content addressable memory (CAM) function for parallel searching without any extra hardware cost, along with enhanced entry compression capability due to the in-situ combinatorial one-hot encoding.
- Experiments and evaluations of the proposed FeFET array-based LUT architecture are carried out based on typical emerging FeFET device. The results show that the proposed LUT can achieve 51.7×/16.7×/8.3× AEDP reduction compared with conventional SRAM-/RRAM-FeFET-based LUT for implementation of 64-bit 6-input LUTs, showing its great potential for area- and energy-efficient reconfigurable systems.

The rest of this paper is organized as follows. Section II introduces the conventional LUT architecture and designs, as well as the FeFET device and model. Section III presents the proposed FeFET array-based LUT architecture and functionality expansion. Section IV discusses the evaluations of proposed FeFET array-based LUT architecture and compares it with the state-of-the-art designs in terms of hardware cost, energy, and latency. Finally, the conclusion is drawn in Section V.

II. Background

A. Conventional LUT Architecture and Existing Designs

Fig. 1a shows the general structure of the conventional K-input LUT architecture, which can implement arbitrary Boolean functions of K input signals [2]. For a conventional K-input LUT, 2^K storage cells are required to store the configuration bits of a specific Boolean function, and a 2^K-to-1 MUX is needed to select one of the 2^K storage cells, which stores the Boolean result, according to the K input signals. Typically, to reduce hardware overhead, commercial LUTs almost use n-type pass transistors to implement the MUX instead of complementary metal-oxide-semiconductor (CMOS) transmission gates, as shown in Fig. 1b [21]. Besides, due to the threshold losses in n-type pass transistors, level restorer or sense amplifier (SA) are used to obtain the full-swing output, and it also limits the maximum number of input variables of single conventional LUT [2]. In the conventional SRAM-based K-input LUT, the configuration bits are implemented using 2^K SRAM cells, with a total of 6×2^K transistors, consuming large hardware cost (Fig. 1c). Besides, due to the volatility of SRAM, additional off-chip NVMs are needed to store the configuration bits, or it will suffer from standby power consumption [2].

Recently, by utilizing the non-volatility and high density of emerging NVMs, the non-volatile LUTs are proposed with lower hardware cost and zero standby power. The existing non-volatile LUT designs [11-20] are still based on the conventional

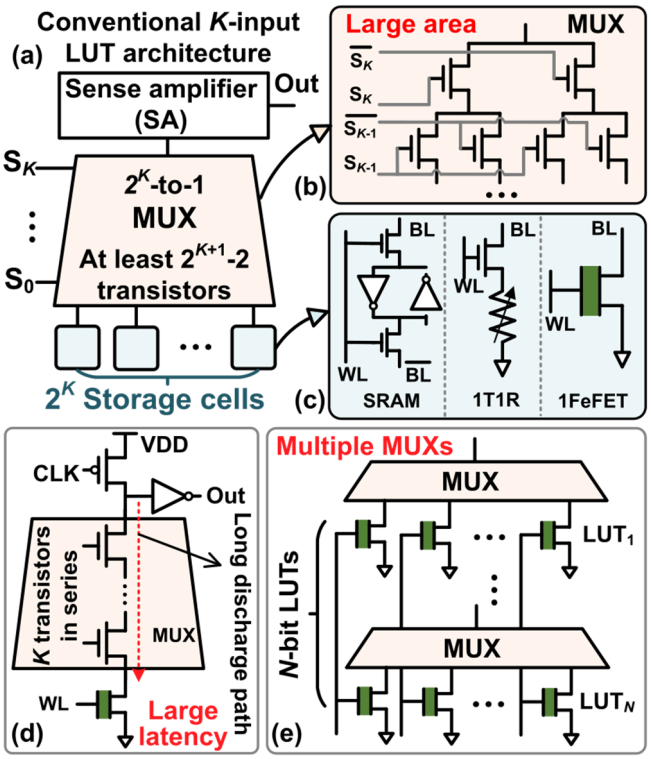


Fig. 1 (a) General structure of the conventional K-input lookup table (LUT) architecture. (b) The structure of n-type pass transistor-based multiplexer (MUX). (c) The configuration bit of LUT based on static random-access memory (SRAM), resistive random-access memory (RRAM), or ferroelectric FET (FeFET). (d) The calculation path of conventional LUT architecture. (e) The implementation of multibit LUTs with multiple MUXs based on conventional LUT architecture, adopting the FeFETs as the storage cells.

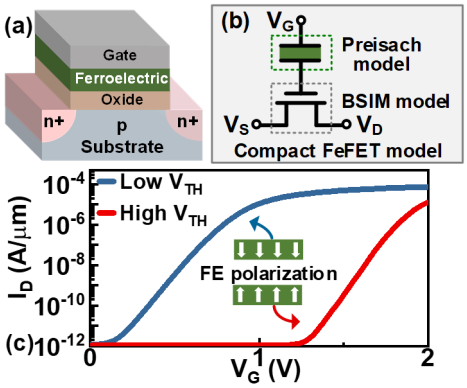


Fig. 2 (a) Device structure of FeFET. (b) The Preisach ferroelectric (FE) switching-based compact FeFET model [22]. (c) The I_D-V_G curves of FeFET modulated by the FE polarization switching.

architecture as depicted in Fig. 1a, by replacing the SRAM storage cells with NVM cells, such as 1-transistor-1-memristor (1T1R) and 1FeFET (Fig. 1c). Therefore, the device number of configuration bits in K-input LUT can be reduced from 6×2^K of SRAM-based LUT to $2^{K+1}/2^K$ of 1T1R/1FeFET-based LUT. Though NVM-based LUTs, especially the FeFET-based LUT, can reduce hardware cost and eliminate the static power consumption compared with conventional SRAM-based LUT, they still suffer from some major drawbacks. On the one hand, for a K-input LUT, there are K transistors of the MUX connected in series in the discharge path (Fig. 1d), resulting in increased

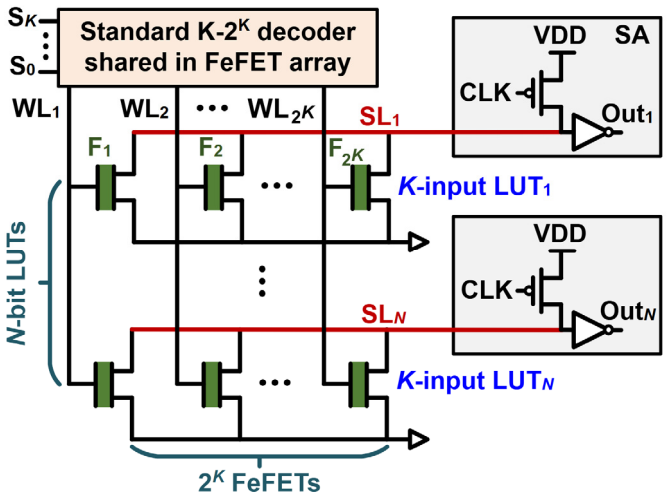


Fig. 3 The structure of proposed FeFET array-based non-volatile LUT architecture for implementing *N*-bit *K*-input LUTs, composed of an FeFET array, standard decoder and sense amplifier (SA).

computation delay. On the other hand, the device count of the simplest n-type pass transistors based 2^K-to-1 MUX is 2^{K+1}-2, which is largely exceeds that of 1FeFET based configuration bits (i.e., 2^K) for one *K*-input LUT, becoming the area bottleneck of NVM-based LUTs. Moreover, multiple MUXs are required in multibit LUTs (Fig. 1e), significantly limiting the area efficiency and energy efficiency of LUT-based applications.

B. FeFET Device and Model

Fig. 2a shows the structure of an FeFET device, where a HfO_2-based ferroelectric (FE) layer is integrated in the gate stack of a MOSFET. By applying a relatively large positive or negative write voltage to the gate, the FE polarization will be switched to the channel or gate direction, and thus the threshold voltage (V_{TH}) of FeFET will be modulated to the low-V_{TH} (V_L) or high-V_{TH} (V_H) state for bit storage, respectively. Then, by applying a relatively small read voltage between V_L and V_H to the gate, the drain current of I_{ON} or I_{OFF} can be readout to represent the stored data of "1" or "0". Compared with two-terminal NVMs, where the write operation is current-driven with high write power consumption, the write operation of FeFET is electric-field-driven without large direct current, and thus more energy efficient. Moreover, the I_{ON}/I_{OFF} ratio of three-terminal FeFET is higher than the resistance ratio of two-terminal NVMs, which is beneficial to enhancing the sense margin of LUT.

In this work, we utilize the calibrated FeFET device model [22] which is established based on Preisach FE switching model and BSIM MOSFET mode for circuit simulations and evaluations (Fig. 2b). Fig. 2c shows the I_D-V_G curves of FeFETs with V_L and V_H states after writing, indicating the non-volatile storage based on FE polarization switching.

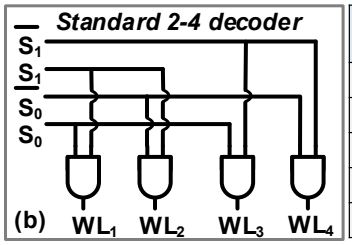
III. Design and Operation of Proposed FeFET Array-based Non-Volatile LUT Architecture

A. FeFET array-based Non-Volatile LUT Architecture Design and Operation

In conventional FeFET-based LUT designs [18-20], the gate voltages of all FeFETs are fixed (e.g., 1V in the case of Fig. 2c)

(a) Combinatorial one-hot encoding scheme of FeFETs

$f(S_1,S_0)$	Logic	F_1	F_2	F_3	F_4
$\overline{S_1}\,\overline{S_0} = \overline{S_1 + S_0}$	NOR	V_H	V_H	V_H	V_L
$\overline{S_1}\,S_0$	-	V_H	V_H	V_L	V_H
$S_1\,\overline{S_0}$		V_H	V_L	V_H	V_H
$S_1\,S_0$	AND	V_L	V_H	V_H	V_H
$\overline{S_1}\,\overline{S_0} + \overline{S_1}\,S_0 = \overline{S_1}$	NOT	V_H	V_H	V_L	V_L
$\overline{S_1}\,\overline{S_0} + S_1\,\overline{S_0} = \overline{S_0}$		V_H	V_L	V_H	V_L
$\overline{S_1}\,S_0 + S_1\,S_0$	XNOR	V_L	V_H	V_H	V_L
$\overline{S_1}\,S_0 + S_1\,\overline{S_0}$	XOR	V_H	V_L	V_L	V_H
$\overline{S_1}\,S_0 + S_1\,S_0 = S_0$	-	V_L	V_H	V_L	V_H
$S_1\,\overline{S_0} + S_1\,S_0 = S_1$		V_L	V_L	V_H	V_H
$\overline{S_1}\,\overline{S_0} + \overline{S_1}\,S_0 + S_1\,\overline{S_0} = \overline{S_1}\overline{S_0}$	NAND	V_H	V_L	V_L	V_L
$\overline{S_1}\,\overline{S_0} + \overline{S_1}\,S_0 + S_1\,S_0 = \overline{S_1} + S_0$	-	V_L	V_H	V_L	V_L
$\overline{S_1}\,\overline{S_0} + S_1\,\overline{S_0} + S_1\,S_0 = S_1 + \overline{S_0}$		V_L	V_L	V_H	V_L
$\overline{S_1}\,S_0 + S_1\,\overline{S_0} + S_1\,S_0 = S_1 + S_0$	OR	V_L	V_L	V_L	V_H
$\overline{S_1}\,\overline{S_0} + \overline{S_1}\,S_0 + S_1\,\overline{S_0} + S_1\,S_0 = 1$	-	V_L	V_L	V_L	V_L



(c) One-hot encoded input signals

S_1S_0	WL_1	WL_2	WL_3	WL_4
00	0	0	0	1V
01	0	0	1V	0
10	0	1V	0	0
11	1V	0	0	0

Fig. 4 (a) The combinatorial one-hot encoding scheme of FeFETs for configuration bits of the proposed LUT architecture, taking 2-input LUT as an example. (b) The structure of standard 2-4 decoder for (c) decoding the binary variables into one-hot encoded input signals.

in the logic computation phase, to obtain two values of I_{DS}, i.e., I_{ON} or I_{OFF}. Therefore, the MUX is indispensable to select an FeFET according to the input signals and obtain the output result based on the different I_{DS} of FeFET. In this work, by combining the storage and computation capabilities of FeFET along with the *in-situ* combinatorial one-hot encoding, a novel FeFET array-based non-volatile LUT architecture is proposed, which can implement multibit LUTs within the compact FeFET array, eliminating the need of multiple expensive MUXs.

Fig. 3 shows the circuit structure of proposed FeFET array-based non-volatile LUT architecture, which includes the FeFET array, input decoder and SA. The proposed *in-situ* combinatorial one-hot encoding scheme is further adopted to efficiently store logic functions and implement logic computations. On the one hand, the configuration bits of each *K*-input logic function are stored in a row of 2^K FeFET cells with a combinatorial one-hot encoding scheme, and *N* rows of the FeFET array can implement *N*-bit LUTs with different logic functions. On the other hand, the input signals is also decoded into one-hot encoded signals through one input decoder shared in the *N*-bit LUTs of FeFET array, which are applied to the gates of each column (i.e., WL) of FeFET array, and thus *in-situ* perform the logic computation. For *K*-input LUTs, a standard *K*-to-2^K decoder can be utilized, which has hardware overhead comparable to that of a single 2^K-to-1 MUX. Besides, Each row employs a straightforward SA, which consists a precharge transistor and an inverter for restoring the output to full swing.

Take 2-input LUT as an example to explain the principle of the proposed LUT architecture. For any logic function, it can be represented in the form of a Sum of Products (SOP), where

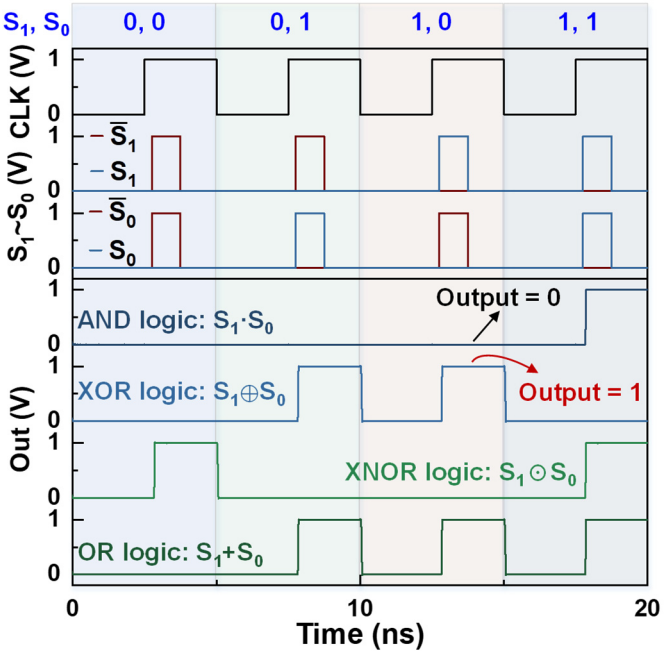


Fig. 5 The transient waveforms of proposed FeFET array-based 2-input LUTs, which are configured as 2-input AND, XOR, XNOR and OR logic functions with different input signals.

multiple AND operations are connected by OR operations, with each AND operation being a sub-expression that includes all input variables (or their negations). Therefore, for two input variables (S_1, S_0), there are a total of 2^2 (i.e., 4) different AND operations (i.e., $\overline{S_1}\cdot\overline{S_0}$, $\overline{S_1}\cdot S_0$, $S_1\cdot\overline{S_0}$ and $S_1\cdot S_0$), which are represented by 4 FeFETs (F_1~F_4) using basic one-hot encoding, where one FeFET is programmed to V_L state and another three FeFETs are programmed to V_H states (Fig. 4a). To configure other logic functions, the corresponding FeFETs are programmed to V_L states according to the form of SOP, called as combinatorial one-hot encoding. For example, when the required logic is XOR (i.e., $\overline{S_1}\cdot S_0 + S_1\cdot\overline{S_0}$), the F_2 and F_3 are programmed to V_L states, and others are programmed to V_H states. In the logic computation phase, the input variables are encoded as one-hot signals based on the standard 2-4 decoder (Fig. 4bc). Only when the input variables make the configured logic true, the high voltage will apply to the gate of FeFET with V_L state, and the FeFET will turn on with high drain current of sense line (SL), indicating the calculation result is true (i.e., "1"). Otherwise, the high voltage will apply to the gate of FeFET with V_H state, and all FeFETs will turn off with low current of SL for the false output (i.e., "0"). Therefore, the computation of logic function can be carried by further utilizing the gate terminal of three-terminal FeFET as input, eliminating the need of MUX in the conventional LUT architecture.

Fig. 5 shows the simulated results of the proposed FeFET array-based 2-input LUTs taking four logic functions (i.e., AND, XOR, XONR and OR) as examples, where the configuration bits are shown in Fig. 4. The computation process can be divided into two stages. In the first stage, the p-type precharge transistor is turn on by a low-voltage CLK signal, precharging the SLs to high voltages. In the second stage, all

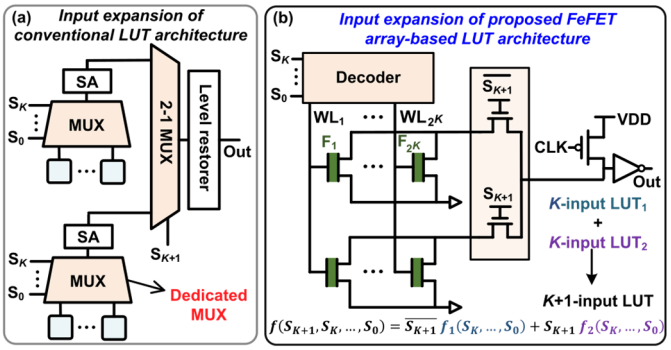


Fig. 6 (a) Input expansion of conventional LUT architecture. (b) Input expansion of proposed FeFET array-based LUT architecture.

SLs are floating with a high-voltage CLK signal, and the decoder sets the corresponding WL to a high voltage for computation based on the input variables. Only when the result of the logic computation is "1", one FeFET in the LUT will turn on, discharging the SL to GND. As a result, the final output (Out) is high voltage with full swing through the inverter. Otherwise, the SL will remain at a high level, and the Out will be GND, indicating the computation result of "0". It should be noted that the number of LUTs can be expanded by increasing the number of rows in the FeFET array without the need of additional MUXs, which will significantly reduce the hardware cost. Furthermore, benefiting from the utilization of three-terminal FeFET for input, the discharge current of proposed LUT is larger than that of conventional LUTs with multiple transistors connected in series, leading to the higher computation speed, which will be discussed in the next session.

B. Multibit LUTs based on proposed LUT Architecture for Pratical Applications

Input expansion of proposed LUT architecture: In the conventional LUT architecture, due to the threshold losses in n-type pass transistors which are in series in the MUX, the number of inputs for one LUT is constrained. As a result, the LUT with larger number of inputs must be expanded with multiple LUTs with smaller number of inputs, resulting in substantial hardware overhead due to the multiple dedicated MUXs (Fig. 6a). Based on the proposed LUT architecture, the input expansion of LUT can be implemented with higher area efficiency and energy efficiency (Fig. 6b). Taking the expansion from K-input LUT to K+1-input LUT as an example, a K+1-input logic function $f(K+1)$ can be represented in the form of $\overline{S_{K+1}}\cdot f_1(K) + S_{K+1}\cdot f_2(K)$. The $f_1(K)$ and $f_2(K)$ are logic functions with K inputs, which can be implemented with two FeFET rows. The S_{K+1} and $\overline{S_{K+1}}$ are the (K+1)th input variable and its negation, which apply to the gates of two transistors connected to the two SLs for input expansion, without the need of two 2^K-to-1 MUXs. Moreover, benefiting from the elimination of MUX in LUT, only one SA is required for input expansion. For the expansion of more inputs, the reduction of hardware overhead is more significant based on the proposed FeFET array-based non-volatile LUT architecture.

Arithmetic functions based on proposed LUT architecture: In practical reconfigurable computing applications, in addition to the simple logic functions, it is necessary to implement

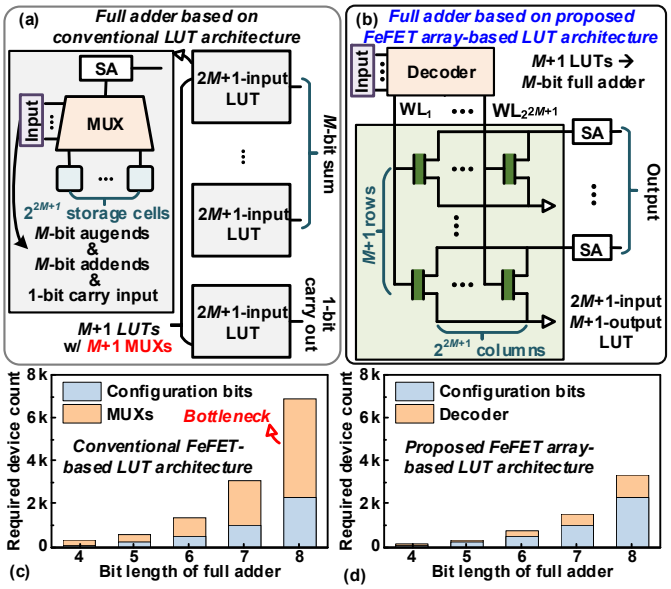


Fig. 7 (a) Implementation of full adder based on conventional LUT architecture. (b) Implementation of full adder based on proposed FeFET array-based LUT architecture. Required device count of full adder based on (c) conventional FeFET array-based LUT architecture and (d) proposed FeFET array-based LUT architecture as the bit length increases.

(a) Mapping to the entry

F_1	F_2	F_3	F_4	Entry
V_L	V_L	V_L	V_H	00
V_L	V_L	V_H	V_L	01
V_L	V_H	V_L	V_L	10
V_H	V_L	V_L	V_L	11
V_L	V_L	V_H	V_H	0X
V_L	V_H	V_L	V_H	X0
V_H	V_L	V_L	V_H	00&11
V_L	V_H	V_H	V_L	01&10
V_H	V_L	V_H	V_L	X1
V_H	V_H	V_L	V_L	1X
V_L	V_H	V_H	V_H	0X&10
V_H	V_L	V_H	V_H	0X&11
V_H	V_H	V_L	V_L	00&1X
V_H	V_H	V_H	V_L	01&1X
V_H	V_H	V_H	V_H	XX

(b) Mapping to the query

WL_1	WL_2	WL_3	WL_4	query
0	0	0	1V	00
0	0	1V	0	01
0	1V	0	0	10
1V	0	0	0	11

(c) Simplified entry range: 1~14

Entry	Binary CAM	Ternary CAM	2-input LUT based CAM
0001	0001	0001	00_01&1X
0010	0010	001X	
0011	0011		
0100	0100	01XX	01&10_XX
0101	0101		
0110	0110		
0111	0111		
1000	1000	10XX	
1001	1001		
1010	1010		
1011	1011		
1100	1100	110X	11_0X&10
1101	1101		
1110	1110	1110	
Count	14	6	3

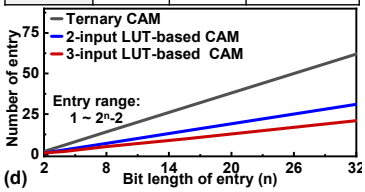


Fig. 8 (a) The combinatorial one-hot encoding scheme of FeFETs for mapping to the entry of content addressable memory (CAM). (b) Mapping the binary query to the one-hot encoded signals. (c) Comparison of the worst-case combination of simplified 4-bit entries based on different CAM modes. (d) Comparison of the entry count as the entry length increases.

various of arithmetic functions, which generally feature multiple input and output signals. Taking a M-bit full adder as an example, it has M-bit addend, M-bit augend and 1-bit carry-in, along with M-bit sum and 1-bit carry-out. The implementation of M-bit full adder based on conventional LUT architecture requires M+1 LUTs with M+1 MUXs, resulting in high hardware cost (Fig. 7a), which will further enlarge as M increases. Based on the proposed LUT architecture, the arithmetic functions can be effectively mapped (Fig. 7b). The M+1 logic functions of the M-bit full adder can be implemented with M+1 rows of the FeFET array and one shared decoder, eliminating the need of additional multiple MUXs. As shown in Fig. 7cd, the proposed FeFET array-based LUT architecture can significantly reduce the hardware overhead, especially as the bit number increases or implementation of more complex arithmetic functions with a greater number of output bits (e.g., multiplier), indicating its great potential for reconfigurable computing.

CAM based on Proposed LUT Architecture: CAM can perform parallel comparison between an input query and entire stored entries, which is widely used for efficient searching and matching operations in reconfigurable computing applications [23-24]. The proposed FeFET array-based LUT architecture can also operate in CAM mode, where the entries are stored in FeFET array and the query is inputted through the same decoder. Fig. 8ab shows the mapping method from 2-input LUT to 2-bit CAM adopting the combinatorial one-hot encoding scheme, where the programmed V_H and V_L states in FeFETs are the opposite of the configuration in LUT, and thus the high voltage will apply to the gate of FeFET with V_H state only when the query matches the entry, resulting the SL to maintain high for the entry vector matching operation. As shown in Fig. 8c, the proposed LUT-based CAM enables compression of any number of entries, which is largely reduced the number of entries for the most challenging range to compress, i.e., 1~(2^n-2) [25], where n is the bit length of entry compared with conventional CAM, such as binary CAM and ternary CAM [26]. Moreover, the reduction is more significant as n or K increases (Fig. 8d), indicating the enhanced entry compression capability of the proposed LUT-based CAM applications.

IV. Evaluations

In this section, the evaluation of proposed FeFET array-based LUT architecture is carried out with circuit simulation in HSPICE. The MOSFETs and FeFETs are based on BSIM model of 45nm technology node [27] and Preisach FE switching model [22], respectively. We mainly evaluate the device count, compute energy and delay of the typically used 6-input LUTs with different bit lengths based on proposed LUT architecture and conventional LUT architecture.

Fig. 9 shows the comparison of required device count for different bit lengths of LUTs. Compared with conventional SRAM-based LUT, the RRAM- and FeFET-based LUTs of conventional architecture reduce the device count about 50% and 62% due to the dense non-volatile devices. Moreover, benefiting from the proposed LUT architecture which eliminates the dedicated MUXs, the required device count can be further reduced compared with conventional LUT architecture. The FeFET array-based LUTs of proposed architecture achieve reduction of device count from 68% to 85% as the bit length of LUTs increase from 8 to 64, indicating the higher area efficiency.

Fig. 10 compares the compute energy consumption between conventional LUT designs and proposed FeFET array-based LUT design. When the computation result is "1", the precharged-high SL will discharge to GND, resulting in the dominant dynamic energy consumption, and the LUT will only consume leakage energy consumption when the computation

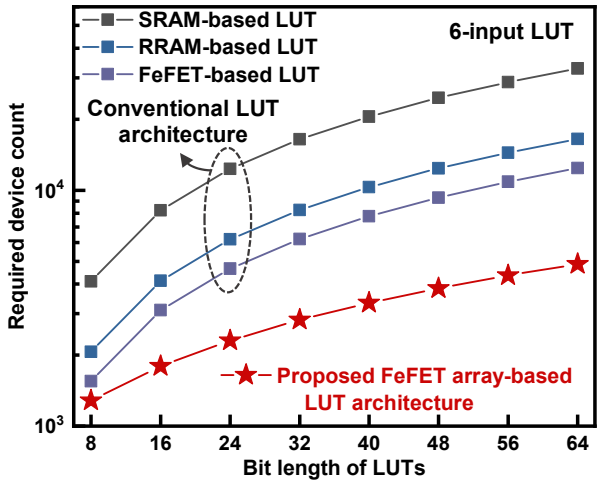


Fig. 9 The required device count of SRAM-/RRAM-/FeFET-based LUTs of conventional LUT architecture and proposed FeFET array-based LUT architecture as the bit length of 6-input LUTs increases.

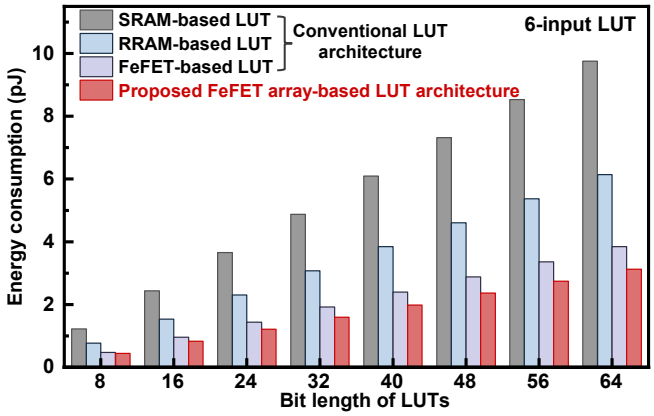


Fig. 10 The comparison of energy consumption between SRAM-/RRAM-/FeFET-based LUTs of conventional LUT architecture and proposed FeFET array-based LUT architecture.

result is "0". Considering that the probabilities of computation results being "1" and "0" are the equal, the power consumption comparison here is the average of the two results. The proposed FeFET array-based LUT architecture eliminates the MUX connected to SL, which not only reduces the leakage power consumption but also decreases the parasitic capacitance of SL, further reducing dynamic power consumption. The proposed FeFET array-based 64-bit length LUTs achieve 3.1×/1.9×/1.3× energy consumption reduction compared with the conventional CMOS-/RRAM-/FeFET-based 64-bit length LUTs, indicating the higher energy efficiency.

The compute latency of LUT is determined by both the parasitic capacitance and the discharge speed of SL. In conventional LUT architecture, the multiple transistors in MUX which are serially connected in the discharge path of the SL not only increases the parasitic capacitance of the SL but also reduces the discharge current of the SL, resulting in increased computation latency. The proposed FeFET array-based LUT architecture eliminates the MUX in the computation path, thereby leading to 5.1×/2.7×/1.6× lower compute latency compared with CMOS-/RRAM-/FeFET-based LUTs of conventional architecture (Fig. 11).

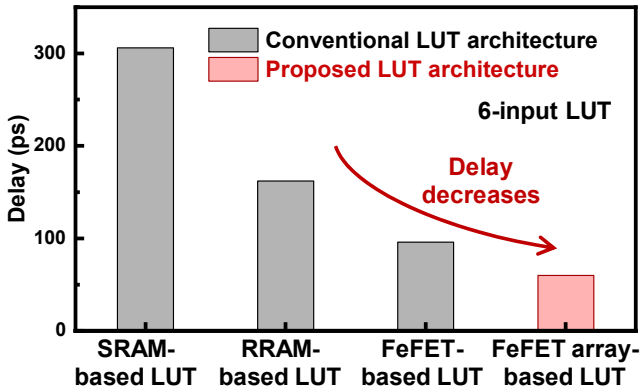


Fig. 11 The comparison of computation delay between SRAM-/RRAM-/FeFET-based LUTs of conventional LUT architecture and proposed FeFET array-based LUT architecture.

Table I. Comparison between conventional LUT architecture and proposed FeFET array-based LUT architecture.

LUT type		Conventional LUT architecture				Proposed FeFET array-based LUT architecture
		SRAM-based LUT [2]	RRAM-based LUT [11]	MTJ-based LUT [15]	FeFET-based LUT [18]	
Compute AEDP reduction*	8-bit	1×	3.1×	3.2×	6.2×	14.3×
	32-bit	1×	3.1×	3.2×	6.2×	34.7×
	64-bit	1×	3.1×	3.2×	6.2×	51.7×
Write EDP**		1×	~1000 ×	~1000 ×	~10 ×	~10 ×

*The compute AEDP is evaluated for 6-input LUTs with different bit lengths
**The write EDP is evaluated using the typical values of SRAM, MTJ, RRAM and FeFET [31-32]

For more comprehensive comparison, the normalized area-energy-delay product (AEDP) of computation is further analyzed, where the area refers to the transistor area, as MTJs, RRAMs and FE layers can be fabricated on the top of MOSFETs through the back-end-of-line process [28-30]. As shown in Table. I, the proposed FeFET array-based LUT architecture achieves 51.7×/16.7×/16.1×/8.3× AEDP reduction compared with conventional SRAM-/RRAM-/MTJ-/FeFET-based LUT for implementation of 64-bit 6-input LUTs, and the reduction will become more significant as the number of LUTs increases, due to the elimination of non-reusable MUXs in the conventional LUT architecture. Moreover, due to the electric-field-driven write mechanism of FeFETs, the write energy-delay product (EDP) exhibits a significant advantage over other NVMs, indicating its great potential for LUT-based reconfigurable computing applications.

V. Conclusion

In this work, a compact and efficient LUT architecture based on FeFET array is proposed and demonstrated with the lowest compute AEDP for implementing multiple LUTs. Benefiting from the elimination of MUX in LUT, the area, energy and latency are all largely reduced. Moreover, based on the proposed design, the input expansion of LUT, full adder for arithmetic function, and CAM for searching function are further implemented and demonstrated with lower hardware cost, showing its great potential for reconfigurable computing.

Acknowledgements

This work was supported by NSFC (61927901, 62374009) and 111 Project (B18001).

REFERENCE

[1] R. Tessier et al., "Reconfigurable Computing Architectures," *Proceedings of the IEEE*, vol. 103, no. 3, pp. 332-354, 2015.

[2] I. Kuon, R. Tessier and J. Rose, "FPGA Architecture: Survey and Challenges," *Foundations and Trends in Electronic Design Automation*, vol. 2: no. 2, pp. 135-253, April 2008.

[3] A. Barkalov, et al., "Using Codes of Output Collections for Hardware Reduction in Circuits of LUT-Based Finite State Machines," *Electronics*, vol. *11*, no. 3, p. 2050, 2022.

[4] R. Fuchikami et al., "Fast and Light-weight Binarized Neural Network Implemented in an FPGA using LUT-based Signal Processing and its Time-domain Extension for Multi-bit Processing," in *IEEE International Conference on Consumer Electronics (ICCE)*, 2019, pp. 120-121.

[5] M. Andronic et al., "PolyLUT: Learning Piecewise Polynomials for Ultra-Low Latency FPGA LUT-based Inference," in *International Conference on Field Programmable Technology (ICFPT)*, 2023, pp. 60-68.

[6] J. Zhang et al., "A comprehensive analysis of DAC-SDC FPGA low power object detection challenge," *Science China Information Sciences*, vol. 67, no. 8, p. 182401, 2024.

[7] S. Bavikadi et al., "ReApprox-PIM: Reconfigurable Approximate Lookup-Table (LUT)-Based Processing-in-Memory (PIM) Machine Learning Accelerator," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 43, no. 8, pp. 2288-2300, 2024.

[8] J. Cong et al., "Energy-efficient computing using adaptive table lookup based on nonvolatile memories," in *International Symposium on Low Power Electronics and Design (ISLPED)*, 2013, pp. 280-285.

[9] Y. He et al., "A 28nm 2.4Mb/mm2 6.9 - 16.3TOPS/mm2 eDRAM-LUT-Based Digital-Computing-in-Memory Macro with In-Memory Encoding and Refreshing," in *IEEE International Solid-State Circuits Conference (ISSCC)*, 2024, pp. 578-580.

[10] I. Kuon et al., "Measuring the Gap Between FPGAs and ASICs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 2, pp. 203-215, 2007.

[11] X. Xue et al., "Low-Power Variation-Tolerant Nonvolatile Lookup Table Design," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 3, pp. 1174-1178, 2016.

[12] X. Cui et al., "An Area-Efficient and Robust Memristive LUT Based on the Enhanced Scouting Logic Cells," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2022, pp. 2571-2575.

[13] Y. Chen et al., "A novel peripheral circuit for RRAM-based LUT," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2012, pp. 1811-1814.

[14] H. Chee et al., "Analysis of a Novel Non-Volatile Look-Up Table (NV LUT) Controller Design with Resistive Random-Access Memories (RRAM) for Field-Programmable Gate Arrays (FPGA)," in *IEEE Regional Symposium on Micro and Nanoelectronics (RSM),* 2019, pp. 87-90.

[15] D. Suzuki et al., "Design of an MTJ-based nonvolatile lookup table circuit using an energy-efficient single-ended logic-in-memory structure," in *IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2015, pp. 1-4.

[16] R. Kuttappa et al., "Comparative analysis of robustness of spin transfer torque based look up tables under process variations," *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2016, pp. 606-609.

[17] D. Kim et al., "SOT-MRAM Based LUT Cell Design for Area and Energy Efficient FPGA," *IEEE Transactions on Circuits and Systems II: Express Briefs*.

[18] X. Chen et al., "Nonvolatile Lookup Table Design Based on Ferroelectric Field-Effect Transistors," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2018, pp. 1-5.

[19] Y. Xu et al., "Ferroelectric FET-based context-switching FPGA enabling dynamic reconfiguration for adaptive deep learning machines," *Science Advances*, vol. 10, no. 3, 2024.

[20] X. Chen et al., "Power and Area Efficient FPGA Building Blocks Based on Ferroelectric FETs," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 5, pp. 1780-1793, 2019.

[21] C. Chiasson et al., "Should FPGAS abandon the pass-gate?," in *International Conference on Field programmable Logic and Applications*, 2013, pp. 1-8.

[22] Z. Fu et al., "Device Modeling and Application Simulation of Ferroelectric-FETS with Dynamic Multi-Domain Behavior," in *China Semiconductor Technology International Conference (CSTIC)*, 2020, pp. 1-4.

[23] R. Karam et al., "Emerging Trends in Design and Applications of Memory-Based Computing and Content-Addressable Memories," *Proceedings of the IEEE*, vol. 103, no. 8, pp. 1311-1330, 2015.

[24] W. Xu, et al. "A Novel Ferroelectric FET based Universal Content Addressable Memory with Reconfigurability for Area-and Energy-Efficient In-Memory-Searching System," *IEEE Electron Device Letters*, 2024.

[25] W. Xu et al., "A Novel Complementary Ferroelectric FET based Compressed Multibit Content Addressable Memory with High Area- and Energy-Efficiency," in *IEEE Electron Devices Technology & Manufacturing Conference (EDTM)*, 2024, pp. 1-3.

[26] K. Pagiamtzis et al., "Content-Addressable Memory (CAM) Circuits and Architectures: A Tutorial and Survey," *IEEE Journal of Solid-State Circuit*, vol. 41, no. 3, pp. 712-727, 2006.

[27] W. Zhao et al., "New Generation of Predictive Technology Model for Sub-45 nm Early Design Exploration," *IEEE Transactions on Electron Devices*, vol. 53, no. 11, pp. 2816-2823, 2006.

[28] M. Chang et al., "Area-efficient embedded resistive RAM (ReRAM) macros using logic-process vertical-parasitic-BJT (VPBJT) switches and read-disturb-free temperature-aware current-mode read scheme," *IEEE Journal of Solid-State Circuits*, 49.4 (2014): 908-916.

[29] Y. Song et al., "Highly functional and reliable 8Mb STT-MRAM embedded in 28nm logic," in *IEEE International Electron Devices Meeting (IEDM)*, 2016.

[30] F. Müller et al., "Multi-level operation of ferroelectric FET memory arrays for compute-in-memory applications," in *IEEE International Memory Workshop (IMW)*, 2023.

[31] S. Yu et al., "Emerging memory technologies: Recent trends and prospects," *IEEE Solid-State Circuits Magazine*, 8.2 (2016): 43-56.

[32] K. Ni et al., "A circuit compatible accurate compact model for ferroelectric-FETs," in *IEEE symposium on VLSI technology*, 2018.