A Novel Quantum-Dot Cellular Automata X-bit × 32-bit SRAM

Abstract:
Application of quantum-dot cellular automata (QCA) technology as an alternative to CMOS technology on the nanoscale has a promising future; QCA is an interesting technology for building memory. The proposed design and simulation of a new memory cell structure based on QCA with a minimum delay, area, and complexity is presented to implement a static random access memory (SRAM). This paper presents the design and simulation of a 16-bit × 32-bit SRAM with a new structure in QCA. Since QCA is a pipeline, this SRAM has a high operating speed. The 16-bit × 32-bit SRAM has a new structure with a 32-bit width designed and implemented in QCA. It has the ability of a conventional logic SRAM that can provide read/write operations frequently with minimum delay. The 16-bit × 32-bit SRAM is generalized and an n × 16-bit × 32-bit SRAM is implemented in QCA. Novel 16-bit decoders and multiplexers (MUXs) in QCA are presented that have been designed with a minimum number of majority gates and cells. The new SRAM, decoders, and MUXs are designed, implemented, and simulated in QCA using a signal distribution network to avoid the coplanar problem of crossing wires. The QCA-based SRAM cell was compared with the SRAM cell based on CMOS. Results show that the proposed SRAM is more efficient in terms of area, complexity, clock frequency, latency, throughput, and power consumption. The proposed architecture of this paper analysis the logic size, area and power consumption using Xilinx 14.2.

Enhancement of the project:

Existing System:
The structure of 16-bit × 32-bit SRAM elements and their layouts in QCA and depicts their logic block diagram.

4-to-16 Decoder in QCA
The 4-to-16 decoder in the first stage uses eight two-input AND gates. The inputs of the AND gates are connected to input wires A, B, C, and D. The 4-to-16 decoder in the second stage uses eight two-input AND gates to enable or disable the output. The 16 outputs are generated by a $4 \times 4$ two-input AND gate network. Fig. 1 shows the schematic layout and its implementation in QCA.

Fig. 1. (a) 4-to-16 decoder implemented in QCA and (b) 4-to-16 decoder layout in QCA with SDN method.
16-to-1 Multiplexer

This section presents an implementation of the 16-to-1 multiplexer (MUX), which is necessary to the design of the 16-bit × 32-bit SRAM. The 16-to-1 MUX is composed of 24 three-input majority gates to implement two-input AND and OR gates in the first stage and 12 three-input majority gates to implement the two-input AND and OR gates in the second stage and nine three-input majority gates to implement the two-input AND and OR gates in the third and fourth stages.

![Diagram of 16-to-1 MUX in QCA and layout in QCA with SDN method.]

Fig. 2. (a) 16-to-1 MUX implemented in QCA and (b) 16-to-1 MUX layout in QCA with SDN method.

Memory Cell

SRAM continues to be a fundamental and vitally important memory technology. They are fast, robust, and easily manufactured in standard logic processes, and nearly universally found on the same die with microcontrollers and microprocessors. The demand for ever larger and ever faster
microprocessors has aggressively driven the scaling of transistor geometries down and the density and speed of the SRAM cells up.

Fig. 3. (a) Memory cell logic and (b) memory cell layout in QCA with SDN method.

In the proposed structure (Fig. 3), when EN = 1, output is enabled and when EN = 0, output = 0. When R/W = 1, the write state is enabled and the D value is saved in the memory loop.

Disadvantages:

- coplanar problem of crossing wires
- delay is high

Proposed System:

16-bit × 32-bit SRAM

In this section, memory cells are located in the 16-bit × 32-bit arrays, as shown in Fig. 4. As shown, the 4-to-16 decoder addresses the 16 SRAM lanes. This SRAM has four addressing lines that can address up to 16 SRAM lanes. After selecting the desired SRAM lane, 32 bits of data can be written or read in accordance with the R/W line. The read or write operation from one 32-bit SRAM lane is applied in accordance with the path delays and occurs at the same clock. Data from the SRAM lanes are transferred to the output by a 16-to-1 MUX in read mode. The four
lines of the addressing decoder are connected to four switch lines of the MUX jointly and the output line of the MUX is selected by the addressing decoder.

Decoders select each unit of memory cell having minimum delay. In the memory, addressing occurs by decoder within 45 clock cycles. Fig. 5(a) shows the layout of one lane of 32-bit SRAM and Fig. 5(b) shows the implementation of 16-bit × 32-bit SRAM in QCA.
Fig. 5. (a) One lane of 16-bit × 32-bit SRAM layout in QCA. (b) 16-bit × 32-bit SRAM layout in QCA with SDN method.

n-bit × 16-bit × 32-bit SRAM

The 16-bit × 32-bit SRAM presented in Section IV-A can be considered a module with a constant structure and correct clocking. First, combine eight modules with suitable connections to produce a 128-bit × 32-bit SRAM. This is used to indicate that the 32-bit data bus is connected to the 32-bit data bus of the 16-bit × 32-bit SRAMs. Addressing eight 16-bit × 32-bit SRAMs is performed by the 3-to-8 decoder.

Data from the 16-bit × 32-bit SRAMs are transferred to the output by an 8-to-1 MUX in read mode. The three lines of the addressing decoder are connected to three switch lines of the MUX jointly and the output line of the MUX is selected by the addressing decoder. When EN1 = Sd1 =
1, the first 16-bit × 32-bit SRAM becomes active; when EN8 = Sd8 = 1, the eighth 16-bit × 32-bit SRAM becomes active. Fig. 6 shows implementation of a 128-bit × 32-bit SRAM by eight 16-bit × 32-bit SRAM modules.

Advantages:

- Avoid the coplanar problem of crossing wires
- Increase the speed

Software implementation:
The Master of IEEE Projects

- Modelsim
- Xilinx ISE