

Harald Richter, Christian Siemers:

**Efficient Reprogrammable Architecture for
Boolean Functions and Cellular Automata**

Content

- Basic ideas
- Definition of the architecture
- Summary and outlook

Basic Ideas (1)

- Two sources for this approach:
 - ◆ Looking for a general computing model to be used inside programmable logic devices
 - ◆ Looking for a memory-based programmable logic device architecture

Basic Ideas (2)

- General computing model:
 - ◆ Global cellular automata are a very good candidate
 - ◆ Cellular automaton:
 - ☞ Finite set of finite state machines (FSM) arranged in a k-dimensional array
 - ☞ Communication is defined to nearest neighbours (e.g. 4). Each FSM can read but not write.

Basic Ideas (3)

- General computing model:
 - ◆ Global cellular automaton (GCA):
 - ☞ Communication is defined to all members of the CA.
 - ☞ Avoiding communication time penalties inside CA.
 - ◆ CA and GCA are known as general purpose computer architecture

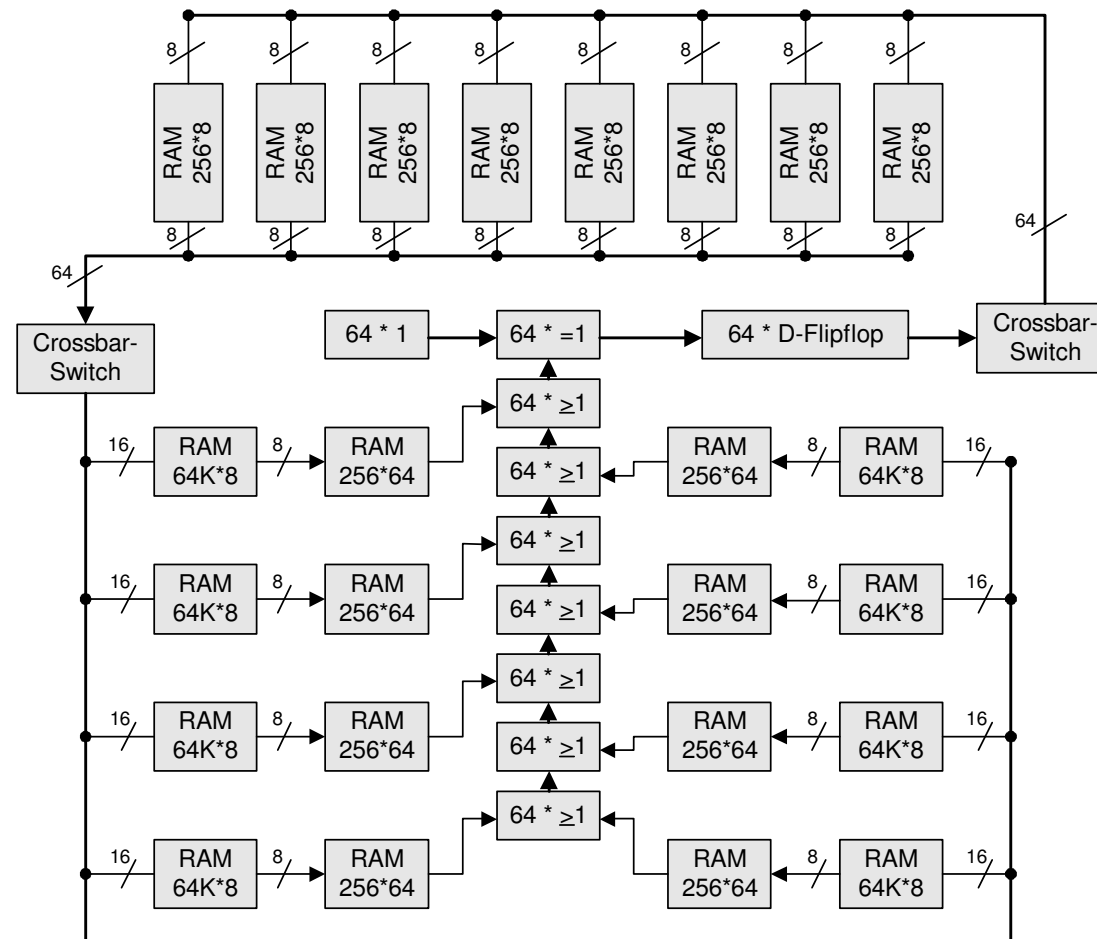
Basic Ideas (4)

- Implementing a GCA inside programmable logic devices
 - ◆ Using N FSMs results in $O(N^2)$ communication effort
 - ◆ The number of states per FSM is not limited
 - ◆ No commercially available device (or architecture) is well-suited for implementing GCAs

Basic Ideas (5)

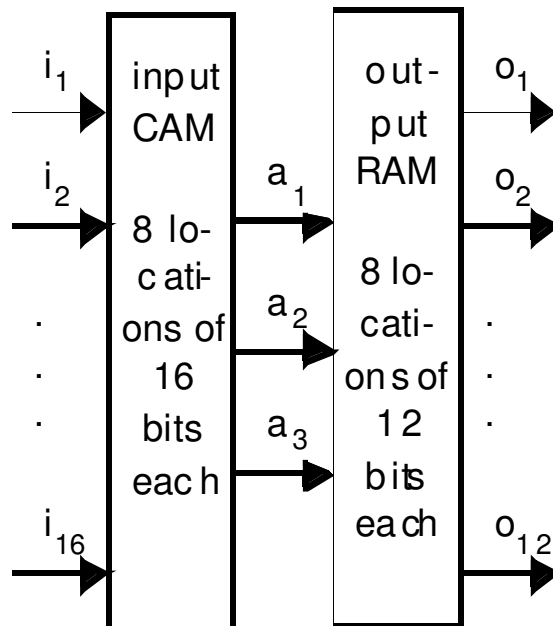
- New approach:
 - ◆ Omitting communication overhead by putting all FSM into one logic block
 - ◆ Dividing the logic block into at least two subblocks for efficient implementation
 - ◆ Using memory arrays (look-up tables) for each of the subblocks

Definition of the Architecture (1)



Definition of the Architecture (2)

input stage output stage



- The most important step is to map the functionality on a CAM/RAM-structure
- The example shows the mapping of 216 locations with 12 bit each on the CAM/RAM network using 8 locations each

Summary and Outlook (1)

- This architecture is very suited for implementation in memory arrays
- It comprises very good efficiency in terms of space and energy consumption but low speed

Summary and Outlook (2)

- Open questions:
 - ◆ Which conditions must be met to map a given function on this architecture?
 - ◆ What is the universal architecture for implementing as much functions as possible with some given parameter (e.g. number of in- and outputs)