Synthesis and implementation of the ordered access memory in programmable logic devices

Nataliia Lishchyna

PhD in Technical Sciences
Lutsk National Technical University
Lutsk, Ukraine
E-mail: lischyna@gmail.com

Valerii Lishchyna

PhD in Technical Sciences
Lutsk National Technical University
Lutsk, Ukraine
E-mail: lvaleriy@gmail.com

Oleh Herasymchuk

PhD in Technical Sciences
Lutsk National Technical University
Lutsk, Ukraine
E-mail: lutsk@ukr.net

Andrii Yashchuk

PhD in Technical Sciences
Lutsk National Technical University
Lutsk, Ukraine
E-mail: xxxxandyxxxx@gmail.com

Abstract
The description of the program memory models with the ordered access is presented in the article. Memory is described in VHDL language. The synthesis of memory was implemented in programmable logic devices using modern technologies and design tools.
Keywords: ORDERED ACCESS MEMORY, ADJUSTABLE SORTING NETWORK, VHDL LANGUAGE

Introduction

Three types of structures of the ordered access memory (OAM) were studied: OAM based on adjustable sorting networks (ASN-based OAM), variable ordered access memory (VOAM) and Fixed ordered access memory (FOAM) based on commutating network (CN). In functional terms ASN-based OAM exceeds the other two types, because it allows to store data matrixes and organize them in an arbitrary order. The order of the data in the original matrix is specified by corresponding indexes.

VOAM allows storing data matrixes. The ordering operator Q must be preliminary calculated. It specifies the order of arrangement of data in the initial matrix. FOAM allows storing data matrixes and provides their ordering according to the rule, specified once on the stage of the design of the memory. The increasing of required resources of equipment, caused by increasing of the capacity of OAM (namely the amount of incoming data), is more considerable for VOAM, and especially for ASN-based OAM, than for FOAM. It should be mentioned that for large values of N on ASN-based OAM the required equipment resources are unacceptably high, so this type of memory is advisable to apply for small values of N.

Objective

Carrying out the investigation of the synthesis of program memory models with the ordered access based on an adjustable sorting network and commutating network with a limited and fixed ordered access with certain characteristics.

Materials and methods

The software for the synthesis of programmable integrated logical micro-schemas were used on the stage of the logical synthesis of the device. The leading companies on the global market of manufacturers of programmable integrated logical schemas are Altera, Xilinx, Actel, Lattice, Atmel, Lucent Technologies etc.

Special hardware was used for programming of programmable logical integrated schemas. This hardware consists of a printed plate with a crystal of programmable logical integrated scheme placed on it and tools for programming of the plate.

The cores of computer devices are developed with the language of hardware description. Special integrated environments with built-in tools of compilation and simulation are used for developing and adjusting of cores of computer devices. The examples of these environments are ModelSIM by Mentor Graphics, Active-VHDL and Active-HDL by Aldec.

Results and discussion

Perform synthesis of programming models of ASN-based OAM and CN-based VOAM and FOAM with the following characteristics: N=8, l= m = 4, n= k =2. The choice of such a small capacity allows us to fully show the designing process and the engineering solutions of OAM.

The data storage environment was synthesized on the base of the registers, which were used as OAM cells. The description of the interface of this data storage environment on VHDL language is shown in fig.1.

After carrying out of modelling by the means of ISE Xilinx Inc. the model of the element of storage environment was obtained. Graphical interpretation of its interface is represented on the fig. 2 [1].

The internal structure of the data storage environment is represented in fig.3. It consists of four registers, 2 elements and their connections.

```
ENTITY reg_gr_0 IS
    PORT(
    D_0 : in std_logic_vector (7 downto 0);
    D_1 : in std_logic_vector (7 downto 0);
    D_2 : in std_logic_vector (7 downto 0);
    D_3 : in std_logic_vector (7 downto 0);

    W : in std_logic;
    Count : in std_logic_vector (1 downto 0);
    CLK : in std_logic;

    O_0 : out std_logic_vector (7 downto 0);
    O_1 : out std_logic_vector (7 downto 0);
    O_2 : out std_logic_vector (7 downto 0);
    O_3 : out std_logic_vector (7 downto 0)

    );
end reg_gr_0;
```

Figure 1. The description of the storage environment by VHDL language

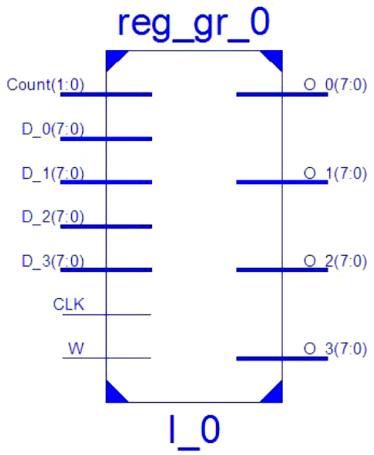

Figure 2. Graphical interpretation of the interface of data storage environment

VHDL description of the interface of the basic element of the adjustable sorting network is represented in fig.4. Its internal structure is represented in the fig.5 [2].

Respectively, internal structure of the adjustable sorting network are presented in fig.6 .

VHDL descriptions of interfaces of OAM ASN, VOAM and FOAM are represented in fig. 7 and 8 respectively.

On the basis of these descriptions using Xilinx ISE 13.2 designing environment the synthesis of current memory types was carried out. Graphical interpretation of OAM, OAM ASN, VOAM and FOAM interfaces is represented in fig.9.

Here D_in_0 – D_in_3 is four input ports of 8-rank data, CLK - the signal of sync pulses, Q – the input of ordered data, T – the enter of data recording regime (data or marks for ordering), W – data recording signal, R – data reading signal, D_out_0, D_out_1 – two input ports of sorted 8-rank data.

Architectural description of OAM

According to selected characteristics of OAM (N=8, l=m=4, n=k=2) data and indexes passing to ports by two groups, four elements in each group. The parallel recording of four elements of data and respective indexes in the memory takes place when the CLK signal transits from “0” to “1” and the Q-signal is high. It is possible to read sorted data by the next transition of the signal CLK from “0” to “1” when the second group of data was recorded in the memory. The reading of the result takes place when the R-signal is active and CLK transits from “0” to “1” [3, 4].

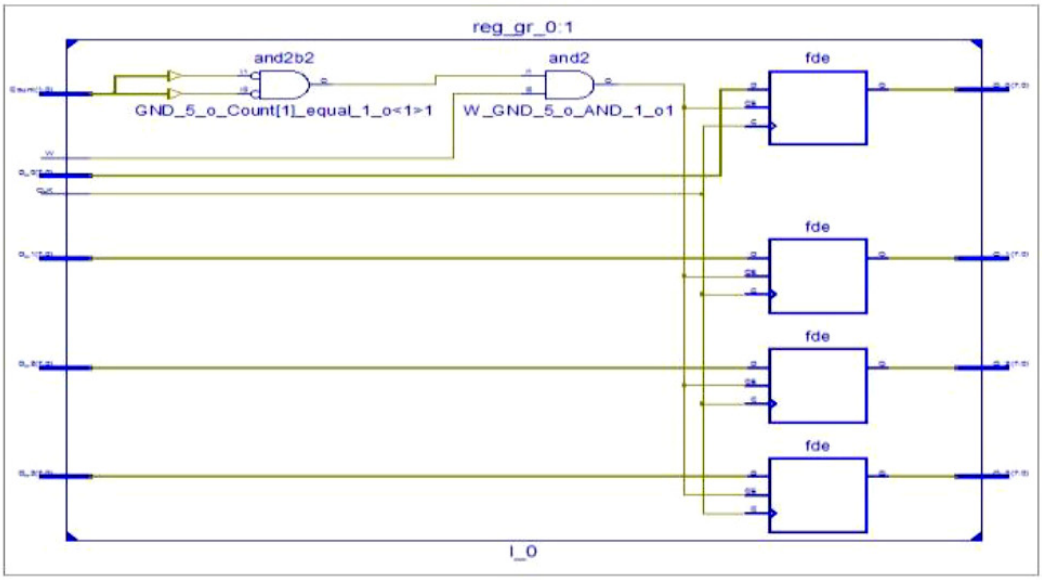

Figure 3. The internal structure of a data storage environment

```
ENTITY elem_compare IS
   PORT(
   ID0 : in std_logic_vector (7 downto 0);
   ID1 : in std_logic_vector (7 downto 0);
   T   : in std_logic;
   OD0 : out std_logic_vector (7 downto 0);
   OD1 : out std_logic_vector (7 downto 0)
   );
end elem_compare;
```

Figure 4. VHDL description of the interface of the basic element of the adjustable sorting network

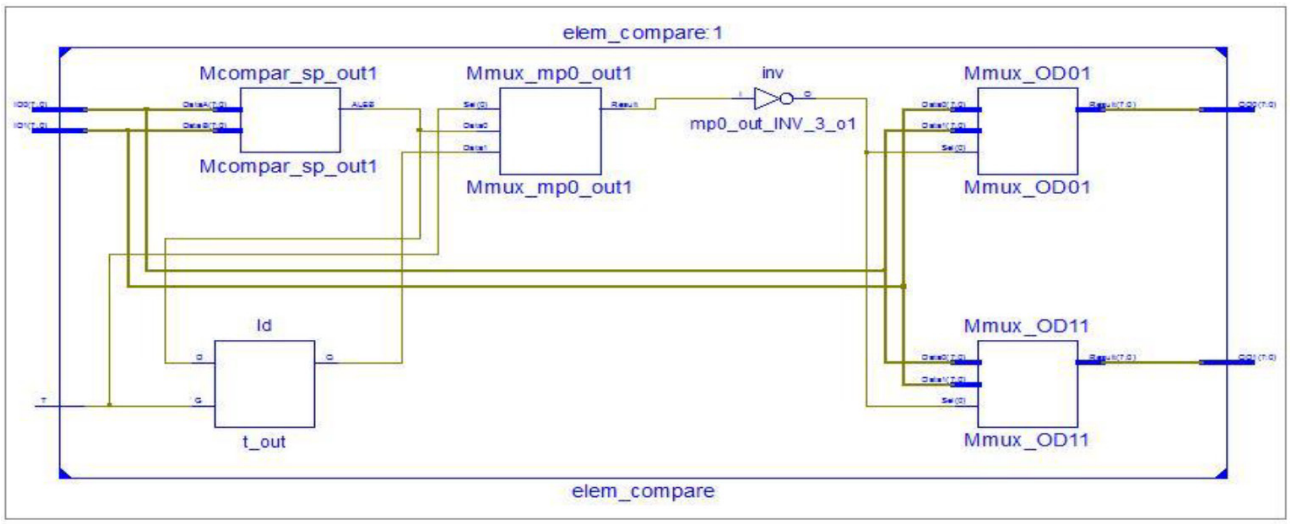

Figure 5. Internal structure of a basic element of the adjustable sorting network

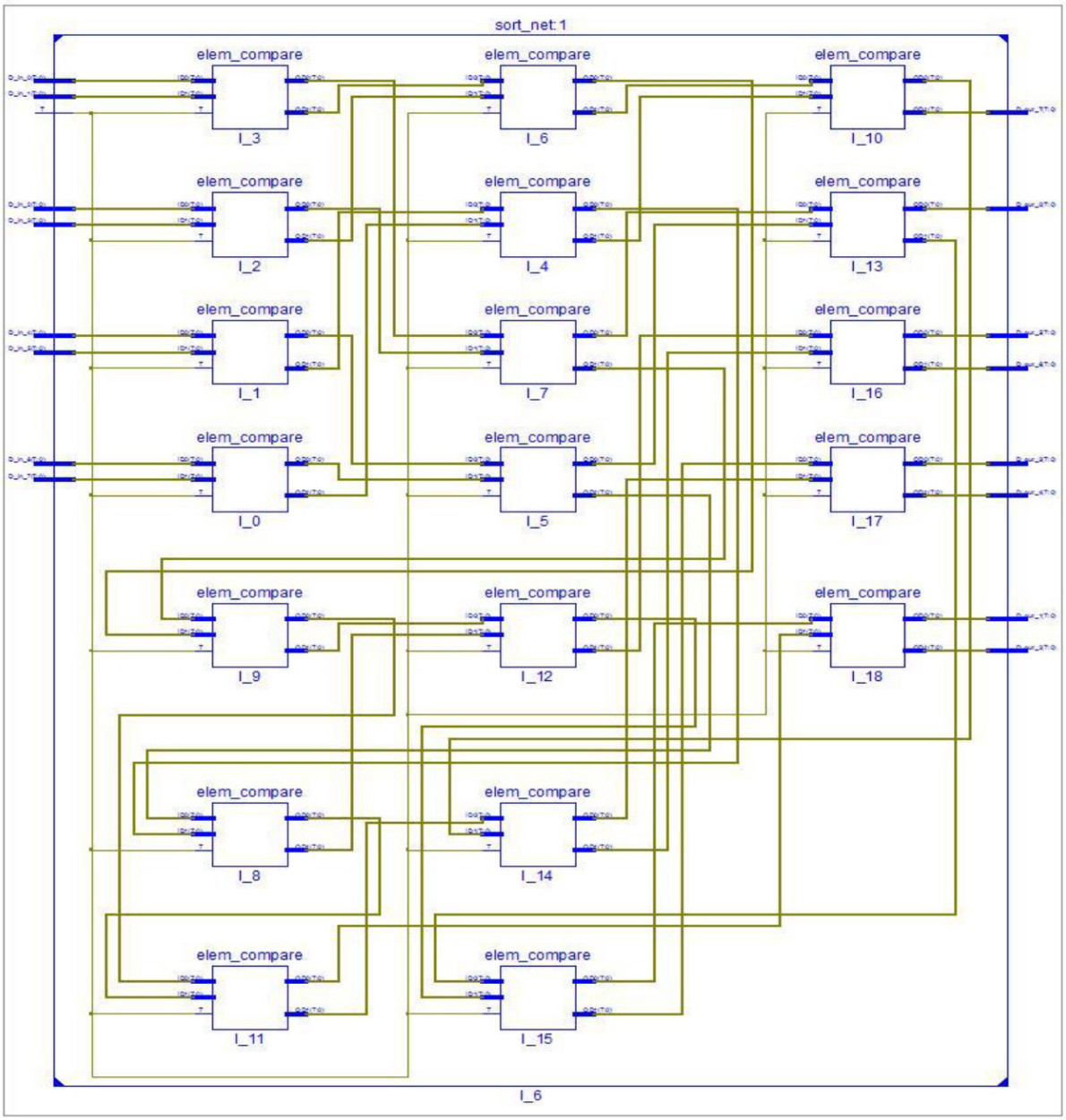

Figure 6. The internal structure of the adjustable sorting network

```
ENTITY pvdn IS
   PORT(
      D_in_0 : in std_logic_vector (7 downto 0);
      D_in_1 : in std_logic_vector (7 downto 0);
      D_in_2 : in std_logic_vector (7 downto 0);
      D_in_3 : in std_logic_vector (7 downto 0);

      T      : in std_logic;

      CLK       : in std_logic;
      W         : in std_logic;
      R         : in std_logic;

      D_out_0 : out std_logic_vector (7 downto 0);
      D_out_1 : out std_logic_vector (7 downto 0)
   );

END pvdn ;
```

Figure 7. VHDL description of OAM ASN interface

```
ENTITY pfvd IS
   PORT(
      D_in_0 : in std_logic_vector (7 downto 0);
      D_in_1 : in std_logic_vector (7 downto 0);
      D_in_2 : in std_logic_vector (7 downto 0);
      D_in_3 : in std_logic_vector (7 downto 0);

      CLK       : in std_logic;
      W         : in std_logic;
      R         : in std_logic;

      D_out_0 : out std_logic_vector (7 downto 0);
      D_out_1 : out std_logic_vector (7 downto 0)
   );

END pfvd ;
```

Figure 8. VHDL description of FOAM interface

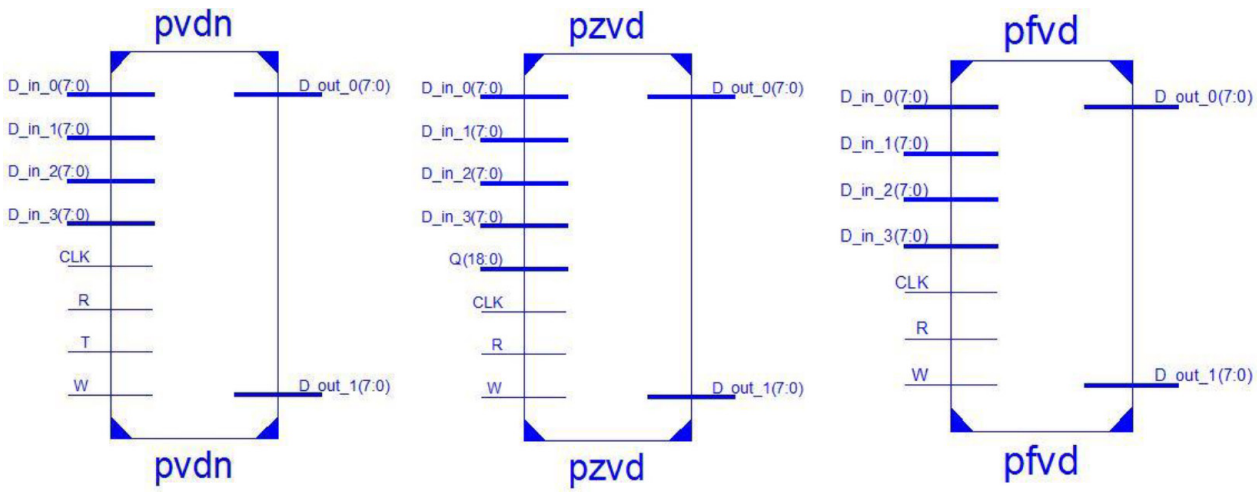

Figure 9. Graphical interpretation of OAM ASN, VOAM and FOAM interfaces

The scheme of ASN-based OAM, synthesized from an architectural description by VHDL in the programming environment of Xilinx ISE 13.2 is shown in fig.10.

The scheme of FOAM, synthesized from an architectural description by VHDL in the programming environment of Xilinx ISE 13.2 is shown in fig.11.

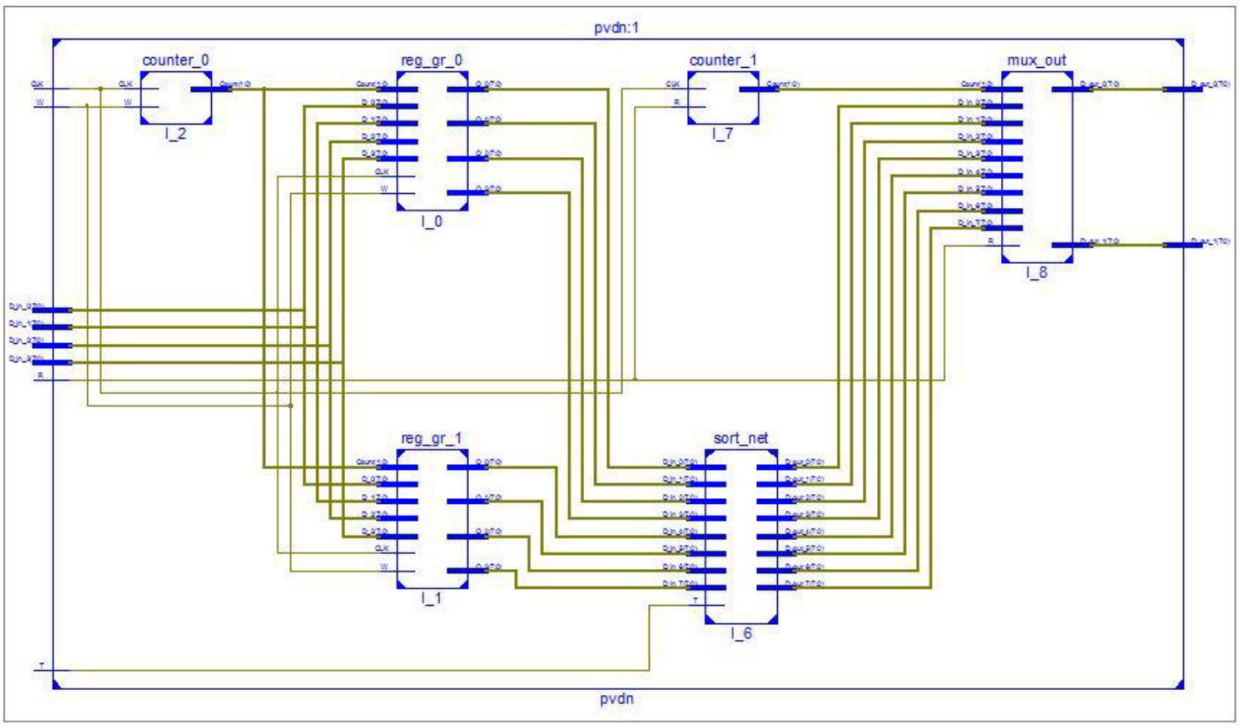

Figure 10. The structure of ASN-based OAM as the result of the synthesis in Xilinx ISE 13.2

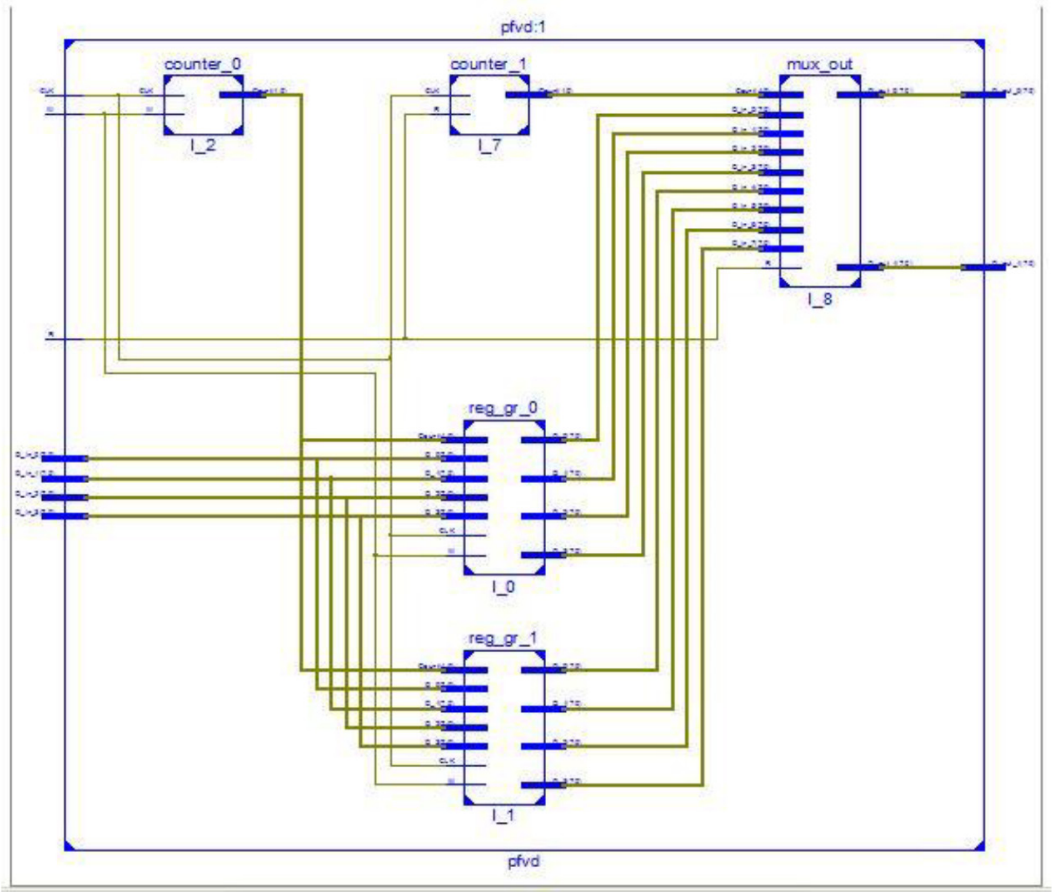

Figure 11. The structure of FOAM as the result of the synthesis in Xilinx ISE 13.2

The results of the synthesis of developed programming models of OAM on the PLD 6vcx75tff484-2 of XILINX Inc. are introduced in the table.

Table. The results of synthesis of ASN-based OAM in PLS 6vcx75tff484-2 of XILINX Inc.

```
Selected Device : 6vcx75tff484-2

Slice Logic Utilization:
 Number of Slice Registers:              86  out of  93120     0%
 Number of Slice LUTs:                  579  out of  46560     1%
    Number used as Logic:               579  out of  46560     1%

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:    643
   Number with an unused Flip Flop:     557  out of    643    86%
   Number with an unused LUT:            64  out of    643     9%
   Number of fully used LUT-FF pairs:    22  out of    643     3%
   Number of unique control sets:         5

IO Utilization:
 Number of IOs:                         52
 Number of bonded IOBs:                 52  out of    240    21%

Specific Feature Utilization:
 Number of BUFG/BUFGCTRLs:               2  out of     32     6%
Minimum period: 1.247ns (Maximum Frequency: 802.137MHz)
Theoretical frequency, the current memory can run at maximum frequency PLD about 250-
300 MHz.
```

Conclusion

The synthesis of programming models of the ordered access memory based on adjustable sorting networks, fixed ordered access memory and variable ordered access memory based on commutating networks was carried out. VHDL was used for creating of the models. Engineering solutions of developed memory types were obtained. The programming models of OAM ASN, VOAM and FOAM based on commutating networks were implemented on the PLD 6vcx75tff484-2 of XILINX Inc.

References

1. Kuzhelin M. (2001). PLIS firmyi Xilinx: semeystvo Spartan-II [PLD Xilinx: Spartan-II range]. *Komponentyi i tehnologii.* No 3.
2. Solovev, V. V. *Proektirovanie tsifrovyih sistem na osnove programmiruemyih logicheskih integralnyih shem* [Design of digital systems on the basis of programmable logical integrated circuits]. Sankt-Peterburg: Goryachaya Liniya, Telekom, 2001, 636 p.
3. Maltsev P. P., Garbuzov N.I., Sharapov A.P., Knyishev D.A. *Programmiruemyie logicheskie IMS na KMOP-strukturah i ih primenenie* [Programmable logical IMS on KMOP-structures and their application]. Mocsow, Energoatomizdat, 1998.
4. Solovev V. V., Klimovich A. Logicheskoe proektirovanie tsifrovyih sistem na osnove programmiruemyih logicheskih integralnyih shem [Logical design of digital systems on the basis of programmable logical integrated circuits]. Sankt-Peterburg: Goryachaya Liniya, Telekom, 2008, 376 p.