

**Розробка засобів обчислювальної техніки  
та дослідження комп'ютерних мереж**

УДК 683.324

С.М. Вороной, канд. техн. наук, доц.,  
В.А. Патрушев, ст. преп.,  
И.В. Ярош, ассист.

ГБУЗ «Донецкий национальный технический университет»

**Исследование аппаратурных затрат при реализации блока  
постоянной памяти управляющего автомата в базисе элементов  
распределенной памяти ПЛИС FPGA Xilinx**

*В работе проводится исследование затрат аппаратуры на реализацию абстрактного блока постоянной памяти управляющего автомата в базисе распределенной логики микросхем типа FPGA фирмы Xilinx. В качестве объекта исследования выступает VHDL-модель блока постоянной памяти, логическая организация которого соответствует организации ряда логических блоков, входящих в структуры современных цифровых устройств управления. Численные значения аппаратурных затрат получены с использованием САПР Xilinx ISE при имплементации VHDL-модели на известных сериях ПЛИС типа FPGA фирмы Xilinx.*

**Ключевые слова:** управляющий автомат, блок памяти, аппаратурные затраты, VHDL-модель, FPGA.

**Введение**

Блоки постоянной памяти, реализуемые в элементном базисе ПЗУ или ППЗУ, традиционно являются составной частью большого множества структур управляющих автоматов (УА) [1, 2]. Во времена реализации схем УА в базисе элементов дискретной логики применение микросхем ПЗУ/ППЗУ вместо ПЛИМ или ПМЛ зачастую давало возможность уменьшить стоимость схемы устройства при приемлемом снижении быстродействия [2, 3]. Этот подход лег в основу широкого ряда методов оптимизации аппаратурных затрат в схемах УА, породив новые структуры УА и методы их синтеза [2, 3]. Анализ данных структур показывает, что многие из них содержат дополнительные блоки, реализуемые в базисе ПЗУ/ППЗУ, то есть имеющие логическую организацию блоков постоянной памяти. Некоторые структуры, рассмотренные в [2], содержат следующие логические блоки, ориентированные на базис постоянной памяти:

- Y-подсхема, используемая во многих структурах автоматов с «жесткой» логикой;
- управляющая память, используемая в различных структурах МУУ и композиционных МУУ (КМУУ);
- F-подсхема в PF-автомате Мили;
- схема CCS в P<sub>C</sub>Y-автомате Мура;

- схема CCS в MPELY-автомате Мура;
- подсхемы F, Y, CCS, в P<sub>C</sub>FY-автомате Мура;
- схемы ПКС и ПКН в автоматах с преобразованием кодов объектов;
- преобразователь адреса в КМУУ;
- преобразователь кодов в КМУУ с преобразованием кодов неэлементарных и элементарных операторных линейных цепей;
- различные преобразователи кодов в автоматах на счетчиках.

Сегодня основным элементарным базисом построения логических схем УА является базис ПЛИС, и в частности – базис ПЛИС типа FPGA фирмы Xilinx [4-6]. Использование языков описания аппаратуры типа VHDL или Verilog совместно со специализированными САПР позволяет синтезировать схему УА на кристалле одной микросхемы типа FPGA [7, 8]. Несмотря на то, что внутренние ресурсы ПЛИС достаточно велики, вопросы минимизации их затрат при синтезе цифровых устройств не теряют свою актуальность. Уменьшение сложности схемы позволяет использовать более простые, а следовательно – более дешевые микросхемы, удешевляя конечное изделие.

Внутренняя структура современных FPGA содержит различные блочные элементы, которые могут рассматриваться как внутренний элементарный базис ПЛИС. Такими элементами являются таблицы соответствия (LUTs, Look-Up

Tables), блочная память, блоки умножения, регистры, преобразователи сигналов, микропроцессорные ядра и др. [5] Возможности современных САПР ПЛИС позволяют проектировщику выбирать, на каких элементах ПЛИС синтезировать тот или иной логический блок.

Блоки постоянной памяти, рассмотренные выше, могут быть реализованы как в базе специальной блочной памяти FPGA, так и в базе так называемой распределенной памяти, фактически представляющей собой LUT-элементы или Slice-элементы, совмещающие в себе несколько LUT элементов, регистры результата и др. В большинстве случаев целесообразным является использование блочной памяти, позволяющее сэкономить LUT-элементы кристалла FPGA. Однако при этом разработчику не следует забывать о следующем:

- блочная память может потребоваться для нужд других устройств, размещающихся на кристалле этой же ПЛИС;
- блочная память является фактически статической памятью типа SRAM и имеет более сложный интерфейс по сравнению с ПЗУ/ППЗУ (в частности, является синхронной);
- если информационный объем блока памяти некоторой структуры УА небольшой, то модуль блочной памяти может использоваться нерационально.

Таким образом, реализация блоков постоянной памяти в базе распределенной памяти вместо блочной памяти может в ряде случаев оказаться более предпочтительной, а иногда и безальтернативной. В этом аспекте интересным является вопрос затрат LUT-элементов на реализацию блоков постоянной памяти различных структур УА. В настоящей работе проводится исследование аппаратных затрат на реализацию некоторого абстрактного блока постоянной памяти при его реализации в базе распределенной памяти ПЛИС типа FPGA фирмы Xilinx.

### **VHDL-модель абстрактного блока постоянной памяти**

VHDL-модель блока постоянной памяти может быть построена различными способами. Используемая в данной работе модель представляет блок памяти в виде асинхронного ПЗУ, функциональное обозначение которого приведено на рис. 1. При разрядности адреса, равной  $R$ , и длине слова данных, равной  $N$ , информационная емкость ПЗУ составит  $N \cdot 2^R$ .

Листинг VHDL-модели приведен после рис. 1. Модель в своей структуре опирается на настраиваемые параметры  $R$  и  $N$  (число входов и

выходов блока соответственно). Блок данных ROM строится на базе пользовательского типа `rom_type`, являющегося одномерным массивом битовых векторов. Содержимое ПЗУ прочитывается из текстового файла «rom.mif» в специальной функции `read_rom`.

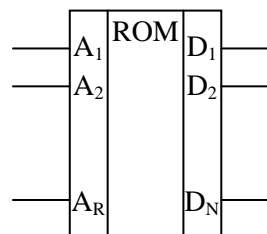


Рисунок 1 – Функциональное обозначение блока асинхронного ПЗУ

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use STD.TEXTIO.all;
-----
entity my_rom is
    generic
        (R: integer := 8;
         N: integer := 15);
    port
        (ADR : in std_logic_vector(R-1 downto 0);
         DO  : out std_logic_vector(N-1 downto 0));
end entity my_rom;

architecture my_rom_a of my_rom is

    type rom_type is array (2**R-1 downto 0) of
        bit_vector (N-1 downto 0);

    impure function read_rom (rom_file_name : in
        string) return rom_type is
        FILE rom_file : text is in rom_file_name;
        variable rom_word : line;
        variable rom : rom_type;
    begin
        for i in rom_type'range loop
            readline (rom_file, rom_word);
            read (rom_word, rom(i));
        end loop;
        return rom;
    end function;

    signal ROM: rom_type := read_rom("rom.mif");
    signal ADR_int: std_logic_vector (R-1 downto 0);
    signal D_int: std_logic_vector(N-1 downto 0);

    begin
        ADR_int <= ADR;
        D_int <= to_stdlogicvector (ROM(conv_integer(ADR_int)));
        DO <= D_int;
    end my_rom_a;
```

Блок описания архитектуры не содержит процессов и представляет собой фактически коммутацию входных и выходных шин.

Приведенная модель, безусловно, не является единственно возможной в нашем случае. Тем не менее, она допускает корректные синтез и имплементацию в ПЛИС FPGA с использованием САПР Xilinx ISE с получением аппаратурных характеристик синтезированного устройства.

### Исследование аппаратурных затрат в блоке постоянной памяти

Суть проводимого исследования заключается в следующем. Для различных параметров R и N блока памяти выполняется синтез VHDL-модели на ПЛИС типа FPGA с помощью САПР Xilinx ISE. При этом в настройках процесса имплементации указано принудительное требование реализовывать схему устройства не на блочной памяти ПЛИС, а на LUT-элементах [4, 5]. Данный эксперимент выполняется для нескольких микросхем ПЛИС различных серий. Настройки средств синтеза и имплементации выбраны «по умолчанию».

Диапазон изменения значений R и N выбран в соответствии с реальными значениями разрядности кода состояния и количества формируемых микроопераций управляющих автоматов различной сложности [2, 3].

1. Микросхема XC3S1000 серии Spartan-3 в корпусе FG320 (общее количество 4-входовых LUT-элементов – 15360).

В табл. 1 представлены значения аппаратурных затрат на реализацию блока памяти, измеряемые в количестве используемых LUT-элементов кристалла ПЛИС. Результаты получены для различных значений R и N.

Таблица 1. Результаты синтеза для ПЛИС XC3S1000 серии Spartan-3

R\N	5	10	15	20	25	30
4	5	10	15	20	25	30
6	20	40	60	80	100	120
8	80	160	245	325	405	484
10	410	799	1144	1509	1869	2324
12	1728	3252	4665	5993	7286	8557

Анализ табл. 1 позволяет сделать следующие выводы:

1) При небольшом размере синтезируемого блока памяти его реализация в базисе LUT-элементов вполне допустима, поскольку составляет несколько процентов от общего числа LUT-элементов кристалла данной ПЛИС.

2) При различных значениях разрядности адреса зависимость роста аппаратурных затрат от количества выходов ПЗУ оказывается близкой к линейной. Данный факт является ожидаемым в силу регулярности внутренней структуры блока памяти и при необходимости позволяет аналитически прогнозировать количество

требуемых ресурсов кристалла при выбранных параметрах блока памяти.

3) При R=10, N=30 затраты на синтезируемый блок памяти составляют 15% LUT-ресурсов кристалла ПЛИС. Учитывая относительно низкую стоимость ПЛИС данной серии, такой показатель является приемлемым и позволяет синтезировать на данной ПЛИС блоки памяти Y-подсхемы и управляющей памяти для среднего числа состояний автомата (до 1024) и количества формируемых микроопераций (до 30).

4) При N=12, N=30 блок памяти занимает более половины ресурсов кристалла ПЛИС, что в ряде случаев может сделать невозможным совместный синтез других узлов автомата на одном кристалле ПЛИС.

2. Микросхема XC6SLX9 серии Spartan-6 в корпусе TQG144 (общее количество slice-элементов – 5720).

В табл. 2 представлены значения аппаратурных затрат на реализацию блока памяти, измеряемые в количестве используемых Slice-элементов кристалла ПЛИС. Результаты получены для различных значений R и N.

Таблица 2. Результаты синтеза для ПЛИС XC6SLX9 серии Spartan-6

R\N	5	10	15	20	25	30
4	3	5	8	10	13	15
6	5	10	15	20	25	30
8	20	40	60	80	100	120
10	85	170	255	340	425	510
12	343	685	1028	1370	1713	2055

Как можно заметить, за счет более высокой ресурсоемкости данной ПЛИС блок памяти с параметрами R=12, N=30 занимает около 36% ресурсов кристалла.

3. Микросхема XC4VLX15 серии Virtex-4 в корпусе SF363 (общее количество 4-входовых LUT-элементов – 12288).

В табл. 3 представлены значения аппаратурных затрат на реализацию блока памяти, измеряемые в количестве используемых LUT-элементов кристалла ПЛИС. Результаты получены для различных значений R и N.

Таблица 3. Результаты синтеза для ПЛИС XC4VLX15 серии Virtex-4

R\N	5	10	15	20	25	30
4	5	10	15	20	25	30
6	20	40	60	80	100	120
8	80	160	239	319	399	479
10	356	780	1149	1503	1830	2157
12	1728	3252	4665	5993	7327	8605

Как и в случае ПЛИС серии Spartan 3, при R=4 каждый выходной сигнал реализуется на

отдельном LUT-элементе, вследствие чего количество затраченных LUT-элементов совпадает со значением N.

Значения в табл. 3 близки к значениям в табл. 1, что также объясняется схожей организацией распределенной логики в ПЛИС данных семейств.

4. Микросхема XC6VCSX130T серии Virtex-6 в корпусе FF484 (общее количество slice-элементов – 20000).

В табл. 4 представлены значения аппаратных затрат на реализацию блока памяти, измеряемые в количестве используемых Slice-элементов кристалла ПЛИС. Результаты получены для различных значений R и N.

Таблица 4. Результаты синтеза для ПЛИС XC6VCSX130T серии Virtex-6

R\N	5	10	15	20	25	30
4	3	5	8	10	13	15
6	5	10	15	20	25	30
8	20	40	60	80	100	120
10	85	170	255	340	425	510
12	343	685	1028	1370	1713	2055

Содержимое табл. 4 полностью идентично содержимому табл. 2. Этому есть две причины:

1) Внутренняя структура распределенной логики данных ПЛИС является схожей.

2) Используемая в экспериментах VHDL-модель является в некотором смысле универсальной и не учитывает архитектурных особенностей конкретной ПЛИС.

Также следует подчеркнуть, что все полученные зависимости имеют характер, схожий с линейным. При необходимости можно получить соответствующие аналитические выражения, которые могут быть применены в системах

автоматизированного проектирования управляющих автоматов на ПЛИС на этапе оценки возможности выбора той или иной ПЛИС в качестве элементного базиса.

### Заключение

Блоки памяти, являющиеся составной частью многих известных структур управляющих автоматов, могут быть реализованы в элементном базисе LUT- или Slice-элементов ПЛИС FPGA фирмы Xilinx. В целом рост аппаратных затрат пропорционален росту информационной емкости синтезируемого блока памяти, что объясняется высокой регулярностью используемого элементного базиса. Блоки памяти небольшой емкости, часто используемые для таких блоков УА, как преобразователи кодов, преобразователи адреса и т.п., могут быть реализованы на LUT-элементах без значительного ущерба свободным ресурсам кристалла ПЛИС. Блоки памяти, реализующие управляющую память и Y-подсхемы и характеризующиеся относительно высокой информационной емкостью, могут занимать свыше 50% ресурсов кристалла недорогой ПЛИС серии Spartan-3, при этом, безусловно, проигрывая в быстродействии варианту их реализации в базисе блочной памяти ПЛИС. Таким образом, наиболее целесообразным следует ожидать гетерогенный подход к реализации блоков памяти управляющих автоматов, при котором блоки памяти малой емкости реализуются на LUT-элементах, а относительно емкие блоки – в базисе распределенной памяти. Данную возможность планируется рассмотреть в ходе дальнейших исследований.

### Список литературы

- Новиков Ю.В. Основы цифровой схемотехники. Базовые элементы и схемы / Ю.В. Новиков. – М.: Мир, 2001. – 379 с.
- Баркалов А.А. Синтез устройств управления на программируемых логических устройствах / А.А. Баркалов. – Донецк: ДонНТУ, 2002. – 262 с.
- Баркалов А.А. Синтез микропрограммных устройств управления / А.А. Баркалов, А.В. Палагин. – К.: ИК НАН Украины, 1997. – 136 с.
- Зотов В.Ю. Проектирование цифровых устройств на основе ПЛИС фирмы Xilinx в САПР WebPack ISE / В.Ю. Зотов. – М.: Горячая линия-Телеком, 2003. – 624 с., ил.
- Кузелин М.О. Современные семейства ПЛИС фирмы Xilinx. Справочное пособие / М.О. Кузелин, Д.А. Кнышев, В.Ю. Зотов. – М.: Горячая линия-Телеком, 2004. – 440 с.
- Программируемые логические ИМС на КМОП-структурах и их применение / [П.П. Мальцев, Н.И. Гарбузов, А.П. Шарапов, Д.А. Кнышев]. – М.: Энергоатомиздат, 1998. – 160 с.
- Бибило П.Н. Основы языка VHDL / П.Н. Бибило. - [3-е изд., доп.]. – М.: Издательство ЛКИ, 2007. – 328 с.
- Грушвицкий Р.И. Проектирование схем на микросхемах программируемой логики / Р.И. Грушвицкий, А.Х. Мурсаев, Е.П. Угрюмов. – СПб.: БХВ-Петербург, 2002. – 608 с.

Надійшла до редакції 09.04.2013

**С.М. ВОРОНОЙ, В.О. ПАТРУЩЕВ, І.В. ЯРОШ**

ДВНЗ «Донецький національний технічний університет»

**ДОСЛІДЖЕННЯ АПАРАТУРНИХ ВИТРАТ ПРИ РЕАЛІЗАЦІЇ БЛОКУ ПОСТІЙНОЇ ПАМ'ЯТІ КЕРУЮЧОГО АВТОМАТА В БАЗИСІ ЕЛЕМЕНТІВ РОЗПОДІЛЕНОЇ ПАМ'ЯТІ ПЛІС FPGA XILINX**

В роботі проводиться дослідження витрат апаратури на реалізацію абстрактного блоку постійної пам'яті керуючого автомата в базисі розподіленої логіки мікросхем типу FPGA фірми Xilinx. У якості об'єкту дослідження виступає VHDL-модель блоку постійної пам'яті, логічна організація якого відповідає організації ряду логічних блоків, що належать структурам сучасних цифрових пристроїв керування. Чисельні значення апаратурних витрат отримані із використанням САПР Xilinx ISE при імплементації VHDL-моделі на відомих серіях ПЛІС типу FPGA фірми Xilinx.

*Ключові слова: керуючий автомат, блок пам'яті, апаратурні витрати, VHDL-модель, FPGA.*

**S.M. VORONOI, V.A. PATRUSHCHEV, I.V. YAROSH**

Donetsk National Technical University

**HARDWARE EXPENCES DURING REALIZATION OF A CONTROL UNIT CONSTANT MEMORY BLOCK IN THE BASIS OF THE ELEMENTS OF DISTRIBUTED MEMORY OF PLD XILINX FPGA**

The paper is devoted to hardware expenses for realization of control unit constant memory abstract block in the basis of distributed logic of chips FPGA Xilinx. Constant memory blocks realized in the element basis of ROM or PROM are traditionally a component of a great number of control units' structures. There are a number of methods of optimization of hardware expenses in the circuits of control units, which are based on the use of ROM instead of PLM or PAL. These methods allow reducing the total cost of the device's scheme, but they add into the automatic unit structure new memory blocks realized in the ROM basis. Today the basic element basis for constructing logic circuits of control units is the basis PLD, and in particular – the basis PLD of FPGA type by Xilinx. Using special languages of hardware description like VHDL together with specialized CADs allows synthesizing a control unit circuit on a FPGA chip crystal. In spite of the fact that internal resources of PLD are great enough, the problem of minimizing their expenses during the synthesis of digital devices is still urgent. Reducing the complexity of the circuit allows using more simple and, consequently, cheaper FPGAs, thus reducing the price of the final product. When using modern FPGA chips the realization of memory blocks is possible both in the basis of block memory of a crystal, and with the use of distributed memory, which is realized on LUT-elements. With small capacity of the memory block the use of LUT-elements can be more rational, than the use of a block memory module. We studied the expenses of LUT-elements for realization of the constant memory block depending on block parameters. A special VHDL-model of a constant memory asynchronous block was developed as an object for this research. The logic organization of the model corresponds to the organization of the memory blocks belonging to the structures of modern digital control units. Numerical values of hardware expenses are obtained using CAD Xilinx ISE during the implementation of VHDL-model on the known PLD series of Xilinx FPGA type.

*Keywords: control unit, memory block, hardware expenses, VHDL model, FPGA.*