

УДК 621-192

**С.Ф. ТЮРИН, С.В. БОГАТЫРЕВ, А.В. ГОЛУБЕВ, А.В. ГРЕКОВ, А.А. ПРОХОРОВ,  
Д.А. ПРОХОРОВ**

*Пермская сельскохозяйственная академия им. академика Д.Н. Прянишникова, Россия*

## **ФУНКЦИОНАЛЬНО-ПОЛНЫЕ ТОЛЕРАНТНЫЕ ЦИФРОВЫЕ СХЕМЫ НА БАЗЕ ПЛИС ФИРМЫ «ALTERA»**

В связи с широким распространением программируемых логических интегральных схем (ПЛИС, programmable logic devices, PLD), которые способны реконфигурироваться в процессе эксплуатации, возникают новые возможности для построения отказоустойчивых цифровых автоматов. "Внутреннее" резервирование наиболее предпочтительно в программируемых логических устройствах с настройкой связей предложенных функционально-полных толерантных элементов, сохраняющих функциональную полноту при заданной модели отказов и обеспечивающих работоспособность на подмножестве базисов подмножества элементов. Это позволяет восстановить вычислительный процесс при некотором снижении производительности.

**ПЛИС, функционально-полные толерантные цифровые схемы, отказоустойчивость, система автоматизированного проектирования**

### **Введение**

ПЛИС – это интегральные микросхемы, содержащие программируемую матрицу элементов логического И (конъюнкторов), программируемую или фиксируемую матрицу элементов логического ИЛИ (дизъюнкторов) и так называемые макроячейки (в зарубежной литературе – macrocells). Макроячейки, как правило, включают в себя триггер, тристабильный буфер и вентиль исключающее ИЛИ, управляющий уровнем активности сигнала. Размерность матриц и конфигурация макроячеек определяют степень интеграции и логическую мощность ПЛИС [1 – 5].

В сочетании с разнообразными обратными связями перечисленные элементы формируют завершённую автоматную структуру, ориентированную на реализацию как комбинационных (дешифраторов, мультиплексоров, сумматоров), так и последовательностных схем (управляющих автоматов, контроллеров, счетчиков). В ПЛИС заложены возможности, которые позволяют превратить ее в ИС с любой функцией цифровой логики.

**Ведущие мировые компании по разработке ПЛИС:**

1. Компания *Xilinx, Inc.* (2100 Logic Drive, San Jose, CA 95124-3400, USA, [www.xilinx.com](http://www.xilinx.com)), была

основана в 1984 году, имеет представительство в России [www.plis.ru](http://www.plis.ru).

В настоящее время компания Xilinx выпускает несколько серий ПЛИС двух типов:

- FPGA - Field Programmable Gate Array (программируемые вентильные матрицы);
- CPLD - Complex Programmable Logic Device (сложные программируемые логические устройства).

Основные особенности ПЛИС Xilinx: значительный объём ресурсов - более 10 млн. системных вентилях на кристалл; высокая производительность с системными частотами до 500 МГц и др.

2. Компания *Actel Corporation* (955 East Arques Avenue, Sunnyvale, CA 94086-4533, USA, [www.actel.com](http://www.actel.com)) была основана в 1985 году. Имеет представительство в России и Украине [www.actel.ru](http://www.actel.ru).

Особенностью ПЛИС Actel является применение так называемой Antifuse технологии, представляющей собой создание металлизированной перемычки при программировании. Данная технология обеспечивает высокую надежность и гибкие ресурсы трассировки. К сожалению, микросхемы Actel, выпускаемые по Antifuse технологии, требуют применения специального программатора, стоимость которого пока еще весьма высока. Однако их отличает

высокая надежность, поэтому они являются весьма перспективной базой для специальных применений.

3. Фирма *Altera Corporation* (101 Innovation Drive, San Jose, CA 95134, USA, www.altera.com) была основана в июне 1983 года. При изготовлении ПЛИС фирмой Altera также используются 3 основные технологии на основе:

- SRAM (ПЛИС с архитектурой FPGA);
- FLASH (ПЛИС с архитектурой CPLD);
- EEPROM (ПЛИС с архитектурой CPLD).

Самыми знаменитыми семействами Altera являются MAX3000, MAX7000, MAX9000, MAXII, и FPGA семейств FLEX10K, FLEX8000, FLEX6000, ACEX 1K, Cyclone, Mercury, Excalibur, APEX 20K, APEX II, Stratix, Stratix GX, Stratix II. ПЛИС Altera имеет достаточное количество развитых бесплатных версий систем автоматизированного проектирования (САПР), такие как MAX+PLUS II BASELINE и Quartus II Web Edition, которые поддерживают все этапы проектирования от ввода проекта до компиляции и программирования. Кроме того, ПЛИС фирмы Altera выпускаются с возможностью программирования в системе непосредственно на плате. Для программирования и загрузки конфигурации устройств опубликована схема загрузочного кабеля ByteBlaster и ByteBlasterMV. Следует отметить, что новые конфигурационные ПЗУ EPC2 позволяют программирование с помощью этого устройства, тем самым отпадает нужда в программаторе, что естественно снижает стоимость владения технологией.

Анализ продукции вышерассмотренных конкурирующих известных производителей ПЛИС [5 – 8] позволяет отдать предпочтение фирме Altera, выпускающей качественные и более дешевые микросхемы программируемой логики с поддержкой мощных бесплатных САПР MAX+plus II и Quartus II Web Edition [8 – 10].

САПР **MAX+plus II** представляет собой интегрированную среду для разработки цифровых устройств на базе программируемых логических интегральных схем фирмы Altera и обеспечивает выполнение всех этапов, необходимых для выпуска готовых изделий [8 – 10]: создание проектов устройств;

синтез структур и трассировку внутренних связей ПЛИС; подготовку данных для программирования или конфигурирования ПЛИС (компиляцию); верификацию проектов (функциональное моделирование и временной анализ); программирование или конфигурирование ПЛИС.

**Цель статьи.** В связи с широким распространением ПЛИС, которые могут реконфигурироваться в процессе эксплуатации, возникают новые возможности для построения отказоустойчивых цифровых автоматов. Предлагается сохранять не исходные логические функции, а базисные функции, позволяющие вычислить исходные за большее время при заданной модели отказов [1 – 4]. Это позволяет восстановить вычислительный процесс при некотором снижении производительности.

### Результаты исследований

Рассмотрим реализацию на ПЛИС логической функции  $z$  мультиплексирования восьми переменных  $a, b, c, d, e, f, g, h$  (генератора функций 3-х аргументов) в ФПТ базисе [2]:

$$z = a\bar{x}_3\bar{x}_2\bar{x}_1 \vee b\bar{x}_3x_2x_1 \vee c\bar{x}_3x_2\bar{x}_1 \vee d\bar{x}_3x_2x_1 \vee ex_3\bar{x}_2\bar{x}_1 \vee f\bar{x}_3\bar{x}_2x_1 \vee gx_3x_2\bar{x}_1 \vee hx_3x_2x_1. \quad (1)$$

Очевидно, что заданием констант вместо переменных  $a, b, c, d, e, f, g, h$  можно получить любую СДНФ трех аргументов  $x_3, x_2, x_1$ . В ФПТ базисе получим

$$z = \overline{\overline{\overline{x_3x_2}(cx_1 \vee dx_1)} \vee \overline{\overline{\overline{x_2}(ax_1 \vee bx_1)} \vee \overline{\overline{\overline{x_3x_2}(cx_1 \vee hx_1)} \vee \overline{\overline{\overline{x_2}(ex_1 \vee fx_1)}}}}}}. \quad (2)$$

В этом выражении все операции имеют вид  $\overline{\overline{x_i x_j}} \vee \overline{\overline{x_j x_i}}$ . Таким образом, необходимо три каскада ФПТ элементов. Соответствующая схема представлена на рис. 1.

Если нет необходимости строить универсальную схему, то предполагается получение схемы трех аргументов, являющихся подсхемами этой схемы путем исключения избыточных элементов. Представим логическую схему мультиплексора в САПР Max+plus II (рис. 2), причём ФПТ элемент моделируем из простых и сложных элементов библиотеки.

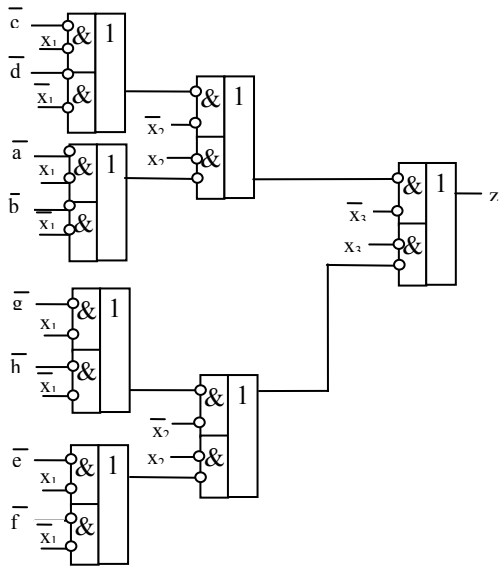


Рис. 1. Генератор функций трех аргументов в ФПТ базисе  $x_1x_2 \vee x_3x_4$

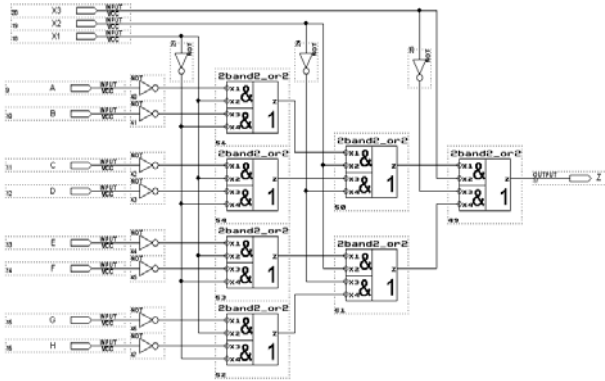


Рис. 2. Логическая схема мультиплексора в MAX+plus II

Сложный элемент «2И-НЕ-2ИЛИ» (рис. 3) является базисным и изображен как символ (рис. 3, а). Он составлен из простых элементов (рис. 3, б) и занесен в библиотеку функционально-полных толерантных схем как базисный элемент, из которого строятся все последующие схемы.

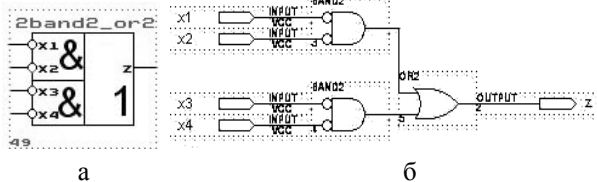


Рис. 3. Модель ФПТ элемента, как сложный элемент «2И-НЕ-2ИЛИ»: а – символ ФПТ элемента; б – логическая схема модели элемента

Для завершения работы над проектом, необходимо провести функциональное моделирование. САПР Max+plus II позволяет проверить правиль-

ность реализованной схемы путем симуляции непосредственно перед конфигурированием ПЛИС. Процесс симуляции отображен на рис. 4.

Проверка реализованной схемы осуществляется сравнением таблицы истинности с временными диаграммами. Убедившись, что временная диаграмма (рис. 4) полностью соответствует, таблице истинности мультиплексора (табл. 1) рассмотрим поведение работы мультиплексора с заданной моделью отказов.

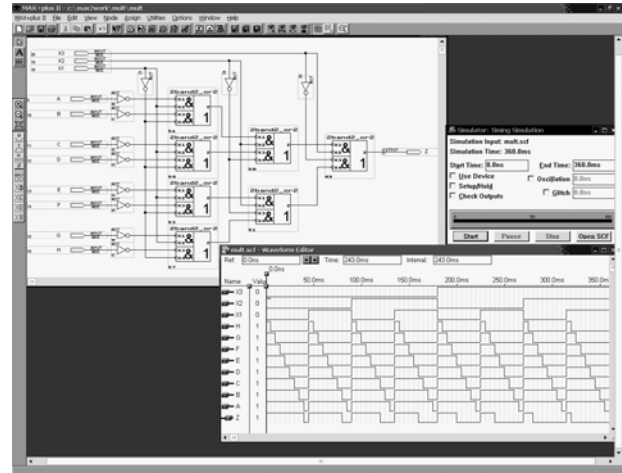


Рис. 4. Тестирование мультиплексора в Max+plus II

Таблица 1

Таблица истинности мультиплексора

X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	Z
0	0	0	a
0	0	1	b
0	1	0	c
0	1	1	d
1	0	0	e
1	0	1	f
1	1	0	g
1	1	1	h

Схема генератора функций реализована на семи ФПТ элементах с базисом  $\bar{x}_1\bar{x}_2 \vee \bar{x}_3\bar{x}_4$ . При синтезе в трехэлементном остаточном базисе  $\bar{x}_i \vee \bar{x}_j\bar{x}_k$ ,  $i, j, k \in \{1,2,3,4\}$  возникают особенности, связанные с отсутствием необходимости выделения одной из подфункций. Так функция (2) представляется в виде

$$z = \bar{x}_3 \vee \bar{x}_2 \vee \bar{a} \vee \bar{x}_1 \vee b\bar{x}_1 \vee \bar{x}_2c \vee \bar{x}_1 \vee \bar{x}_1d \vee \bar{x}_3 \bar{x}_2 \vee g \vee \bar{x}_1 \vee h\bar{x}_1 \vee \bar{x}_2e \vee \bar{x}_1 \vee f\bar{x}_1, \quad (3)$$

т.е., требуется 14 элементов с указанным остаточным базисом, что всего на один элемент превышает сложность реализации в базисе 4И-НЕ (рис. 5).

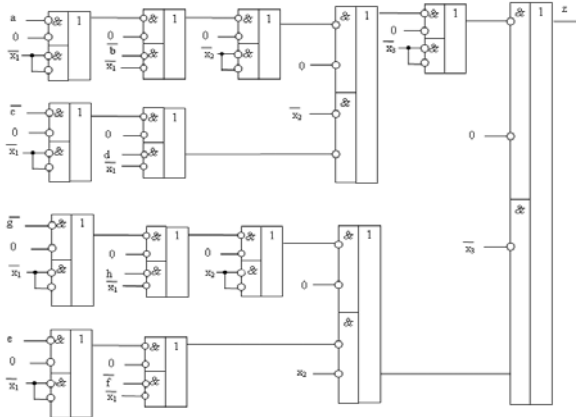


Рис. 5. Реализация генератора функций трех аргументов в остаточном базисе  $\bar{x}_1 \vee \bar{x}_3 \bar{x}_4$

При реализации функции  $z$  в одном базисе на исходном множестве элементов необходимо 2 такта с реконfigurацией связей. Причем возникает дополнительная задача программно (микропрограммно)-аппаратной декомпозиции. Дело в том, что если по результатам реализации верхней половины подсхемы получим логический “0”, то, очевидно, и вся функция равна логической “1” и дополнительных перекоммутаций и вычислений не требуется. Кроме того, заметим, что при вычислениях в соответствии с нижней половиной схемы полная коммутация не потребуется, так как часть входов (например,  $\bar{x}_1$ , “0”) остается той же самой, что позволяет сократить время реконfigurации. Реализация в остаточном базисе  $\bar{x}_i, \bar{x}_j, i, j \in \{1,2,3,4\}$  производится тривиально.

Представим схему мультиплексора остаточном базисе в САПР (рис. 6) и проверим ее правильность в симуляторе Max+plus II (рис. 7). По аналогии с реализацией мультиплексора в остаточном базисе  $\bar{x}_1 \vee \bar{x}_3 \bar{x}_4$ , с помощью закона Де Моргана и законов элементарных преобразований, выведем из формулы мультиплексора в базисе  $\bar{x}_1 \bar{x}_2 \vee \bar{x}_3 \bar{x}_4$ , формулу мультиплексора в базисе  $\bar{x}_1 \bar{x}_2$ .

$$z = a \cdot \bar{x}_1 \cdot \bar{b} \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{c} \cdot \bar{x}_1 \cdot \bar{d} \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{e} \cdot \bar{x}_1 \cdot \bar{f} \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{g} \cdot \bar{x}_1 \cdot \bar{h} \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \quad (4)$$

Далее представим эту формулу в графическом виде (рис. 8), а также реализуем ее в САПР Max+plus II (рис. 9).

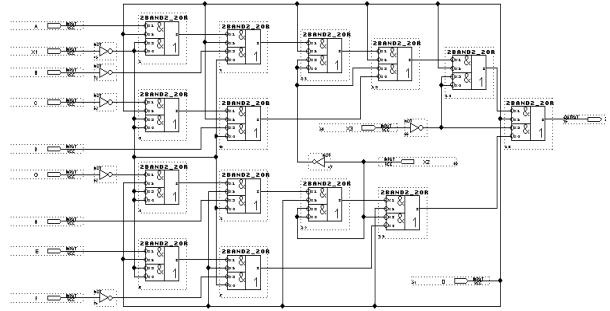


Рис. 6. Реализация генератора функций трех аргументов в остаточном базисе  $\bar{x}_1 \vee \bar{x}_3 \bar{x}_4$

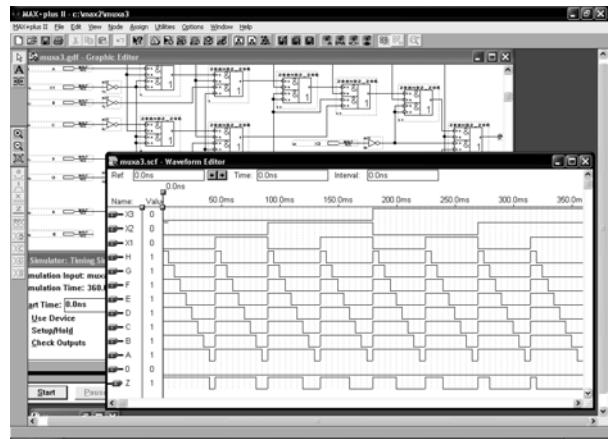


Рис. 7. Симуляция генератора функций трех аргументов в САПР Max+plus II

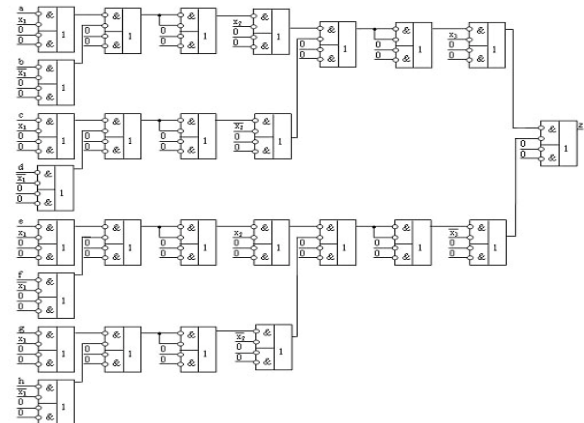


Рис. 8. Реализация генератора функций трех аргументов в остаточном базисе  $\bar{x}_1 \bar{x}_2$

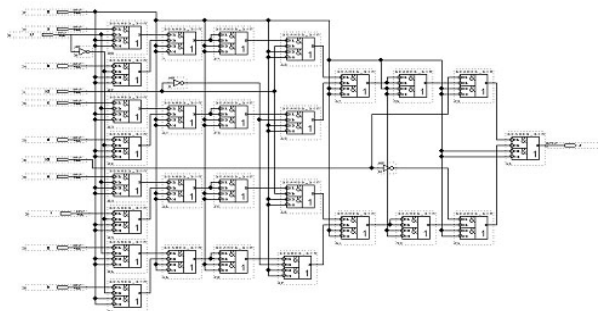


Рис. 9. Реализация генератора функций трех аргументов в САПР Max+plus II

Затем необходимо провести функциональное моделирование в САПР Max+plus II (рис. 10)

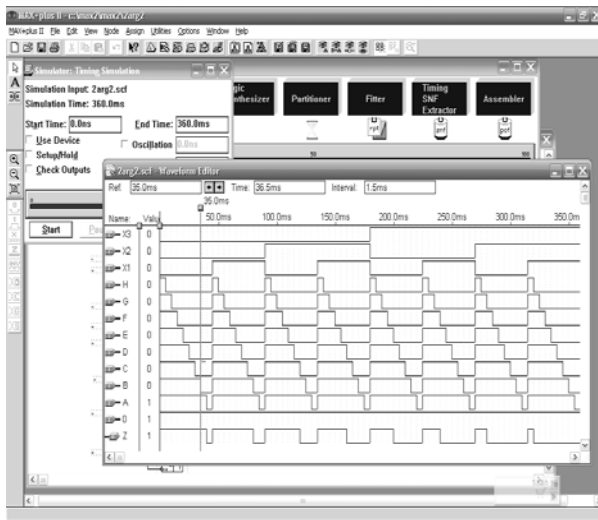


Рис. 10. Симуляция генератора функций трех аргументов в САПР Max+plus II

### Выводы

Таким образом, при использовании ФПТ элементов в качестве базовых, на ПЛИС могут быть реализованы отказоустойчивые цифровые автоматы, которые могут функционировать в однократных случаях константных отказов входов ФПТ элементов, причём для этого необходима соответствующая реконфигурация схем. В случае недостаточности количества элементов, вычисления автоматных отображений могут быть выполнены за несколько тактов.

В случае массированных отказов элементов, например, в катастрофических ситуациях, общий базис для всего массива может привести к снижению эффективности кристалла по функциональной возможности и временным характеристикам. Целесообразно произвести выбор таких общих базисов для подмножества элементов. В этом случае возникает задача оптимизации выбора общего базиса при разделении массива вентилей на заданное реконфигурируемое подмножество.

Изложенные подходы могут быть использованы для повышения глубины диагностирования схем, для повышения процента "выхода годных" при производстве СБИС.

### Литература

1. Тюрин С.Ф. Синтез адаптируемой к отказам цифровой аппаратуры с резервированием базисных функций // Приборостроение. – 1999. – № 1. – С. 36-39.
2. Тюрин С.Ф. Адаптация к отказам одновыходных схем на генераторах функций с функционально-полными толерантными элементами // Приборостроение. – 1999. – № 7. – С. 32-34.
3. Тюрин С.Ф. Проблема сохранения функциональной полноты булевых функций при «отказах» аргументов // Автоматика и телемеханика. – 1999. – № 9. – С. 176-186.
4. Тюрин С.Ф., Несмелов В.А., Харитонов В.А. и др. Программируемое логическое устройство. – Патент РФ № 2146840. Оpubл. БИ № 8. 2000 г.
5. Бродин В.Б., Калинин А.В. Системы на микроконтроллерах и БИС программируемой логики. – М.: ЭКОМ, 2002. – 344 с.
6. Зотов В.Ю. Проектирование цифровых устройств на основе ПЛИС фирмы Xilinx в САПР WebPACK ISE. – М.: Горячая линия-Телеком, 2003. – 522 с.
7. Официальный сайт фирмы «Xilinx» [Электрон. ресурс]. – Режим доступа: <http://www.xilinx.com>.
8. Представительство «Altera» в России [Электрон. ресурс]. – Режим доступа: [www.altera.ru](http://www.altera.ru).
9. Официальный сайт фирмы «Altera» [Электрон. ресурс]. – Режим доступа: <http://www.altera.com>.
10. Представительство «Actel» в России и Украине [Электрон. ресурс]. – Режим доступа: [www.actel.ru](http://www.actel.ru).

Поступила в редакцию 22.02.2007

**Рецензент:** д-р техн. наук, проф. В.А. Краснобаев, Харьковский национальный технический университет сельского хозяйства им. П. Василенко, Харьков.