

МИКРОСХЕМЫ ИНТЕГРАЛЬНЫЕ 5576ХС2Т

Техническое описание

ГПКФ.431262.002ТО

Содержание

1	Введение	3
2	Назначение	3
3	Основные технические характеристики.....	3
3.1	Основные функциональные параметры.....	4
3.2	Особенности ПЛИС.....	4
3.3	Особенности внутренних межсоединений ПЛИС	4
3.4	Особенности элементов ввода-вывода ПЛИС	5
3.5	Особенности САПР для ПЛИС 5576XC2T.....	5
3.6	Значения электрических параметров микросхем.....	6
3.7	Предельно-допустимые режимы эксплуатации	8
4	Условное обозначение и назначение выводов микросхем	9
4.1	Условное графическое обозначение микросхем.....	9
4.2	Функциональное назначение выводов микросхем	11
5	Описание архитектуры ПЛИС	17
5.1	Логический блок.....	19
5.2	Логический элемент	21
5.2.1	Режимы работы логического элемента	22
5.2.1.1	Нормальный режим.....	22
5.2.1.2	Арифметический режим	23
5.2.1.3	Режим суммирующего/вычитающего счетчика.....	24
5.2.1.4	Режим сбрасывающего счетчика	25
5.2.2	Управление логикой сброса и установки	25
5.3	Система межсоединений	27
5.4	Элемент ввода-вывода	28
5.4.1	Соединение элементов ввода-вывода с горизонтальными каналами.....	30
5.4.2	Соединение элементов ввода-вывода с вертикальными каналами ..	31
5.4.3	Выделенные (специализированные) входы	31
5.5	Поддержка периферийного сканирования	33
5.5.1	Функциональная модель СПС	34
5.5.2	Архитектура Boundary-Scan регистра.....	36
5.6	Тестирование	38
6	Расчет мощности потребления ПЛИС	39

1 Введение

Настоящее техническое описание (ТО) предназначено для изучения микросхем интегральных 5576XC2T и содержит описание принципа работы, технические характеристики и другие сведения, необходимые для обеспечения полного использования технических возможностей микросхем.

2 Назначение

Данные микросхемы являются результатом разработки нового поколения КМОП БИС - программируемых пользователем логических ИС (ПЛИС) с возможностью многократного изменения конфигурации. Микросхемы предназначены для использования в системах управления при разработке аппаратуры специального назначения, а также для замещения импортных ПЛИС ф. Actel (A1010A/B, A1020A/B, A1225A, A1415, A1425), ф. Altera (EPF8282A/V, EPМ7128S/AE, EPМ7064S/AE), ф. Xilinx (XC3020A/V, XC3120A, XC3030A/V, XC3120A, XC4002XL, XC5202).

Логическая емкость ПЛИС 5576XC2T позволяет заменить до 35 восьмиразрядных счетчиков с ускоренным переносом или (25-⁷⁵) микросхем малой и средней степени интеграции.

3 Основные технические характеристики

ПЛИС 5576XC2T функционально совместима (но не является аналогом) с изделиями EPF8282A, EPF8282AV ф. Altera. Кристаллы ПЛИС изготовлены по 0,35 мкм КМОП - технологии на эпитаксиальных структурах с одним уровнем поликремния и четырьмя уровнями металлизации.

Микросхемы устойчивы к воздействию статического электричества с потенциалом не менее 2000 В.

3.1 Основные функциональные параметры ПЛИС 5576XC2Т приведены в таблице 1.

Таблица 1

Параметр	Значение
Число эквивалентных вентиляей	2500
Количество логических элементов	208
Количество логических блоков	26
Количество пользовательских триггеров	272 ¹⁾
Количество выводов, программируемых пользователем	64
¹⁾ 208 в логических элементах, 64 в элементах ввода-вывода.	

3.2 Особенности ПЛИС:

- напряжение питания 3,3 В;
- толерантность входов-выходов к 5 В внешним сигналам;
- толерантность к внешним сигналам при выключенном напряжении питания;
- логические матрицы для основных логических функций;
- встроенная система конфигурирования (ICR), обеспечивающая многократное перепрограммирование ПЛИС посредством внешних конфигурационных устройств или интеллектуальным контроллером через специальный загрузочный порт или JTAG порт;
- встроенный блок тестирования (Joint Test Action Group (JTAG)) с использованием схемы периферийного сканирования (BST), совместимый со стандартом IEEE Std. 1149.1-1990;
- режим циклической перезаписи конфигурационной памяти без выхода из рабочего режима;

- режим циклической верификации конфигурационной памяти без выхода из рабочего режима;

- пользовательские триггеры, триггеры загрузчика и конфигурационная память имеют систему восстановления данных после сбоев, вызванных воздействием тяжелых заряженных частиц.

3.3 Особенности внутренних межсоединений ПЛИС:

- внутренние быстрые межсоединения (FastTrack Interconnect), являющиеся быстрыми структурами с известными задержками;

- специализированная цепь переноса, обеспечивающая поддержку арифметических функций, таких как быстрые сумматоры, счётчики и компараторы (используется автоматически программными инструментами);

- специализированная цепь каскадирования, обеспечивающая поддержку высокоскоростных функций с высоким коэффициентом разветвления на входе (используется автоматически программными инструментами);

- внутренняя эмуляция третьего состояния;

- четыре выделенных (специализированных) входа, которые можно использовать в качестве формирователей глобальных тактовых импульсов.

3.4 Особенности элементов ввода-вывода ПЛИС:

- индивидуальный сигнал разрешения третьего состояния для каждого вывода;

- программируемая нагрузочная способность элементов ввода-вывода (12/24 мА);

- снижение помех при переключениях за счёт программируемой опции «slew-rate control»;

- программируемая опция «bus hold»;

- программируемая опция «pull-up»;

- программируемая опция «pull-down»;

- наличие периферийного регистра минимизирует задержку от тактового импульса до изменения информации на выходе.

3.5 Особенности САПР для ПЛИС 5576XC2T:

- для создания проектов используется САПР MAX+PLUS II ф. Altera для ПК на базе Windows, а также для рабочих станций Sun SP ARC station, IBM RISC

System/6000, HP 9000 Series 700/800. Возможен ввод проекта и моделирование, автоматическое размещение элементов и трассировка межсоединений, конфигурирование ПЛИС;

- для ввода проекта и моделирования можно использовать EDIF 200 и EDIF 300 нетлист-файлы, библиотеки параметризованных функций (LPM), Design Ware компоненты, Verilog HDL, VHDL и другие интерфейсы популярных инструментов САПР из таких программ как Cadence, Exemplar Logic, Mentor Graphics, OrCAD, Synopsys, Synplicity, VeriBest и Viewlogic;

- для программирования расширенной функциональности элементов ввода-вывода ПЛИС 5576XC2T (повышенная до 24 мА нагрузочная способность выходных буферов по высокому и низкому уровням сигнала, функции «bus hold», «pull up», «pull down») используется специальное программное обеспечение собственной разработки.

3.6 Значения электрических параметров микросхем интегральных 5576XC2T приведены в таблице 2.

Таблица 2 - Электрические параметры микросхем при приемке и поставке

Наименование параметра, единица измерения, режим измерения	Буквенное обозначение параметра	Норма параметра		Температура среды, °С
		не менее	не более	
1	2	3	4	5
Выходное напряжение низкого уровня, В, $U_{cc}=3,0$ В, $I_{OL}=24$ мА	UOL	-	0,55	-60±3 25 ±3 100 ±5
Выходное напряжение высокого уровня, В, $U_{cc}=3,0$ В, $I_{OH}=-24$ мА	UQH	2,2	-	
Ток потребления, мА, $U_{cc}=3,6$ В $U_I=0$ В или $U_I=U_{cc}$	Icc	—	10	

Окончание таблицы 2

1	2	3	4	5
Выходной ток в состоянии «Выключено», мкА, $U_{cc} = 3,6 \text{ В}$, $U_0 = 0 \text{ В}$ или $U_0 = 5,5 \text{ В}$	IGZ	— 40	40	-60 ± 3
Длительность тактового интервала межрегистровой пересылки, нс, $\Gamma_{cc} = 360 \text{ ИБ}$, $\Gamma_{шд} = 0 \text{ ИБ}$ $U_{ин} = U_{cc}$	t_{DRR}	-	15,8	25 ± 3 100 ± 5

Длительность тактового интервала межрегистровой пересылки t_{DRR} является основным динамическим параметром, характеризующим быстродействие ПЛИС. Контроль длительности тактового интервала межрегистровой пересылки t_{DRR} проводится на тестовой конфигурации ПЛИС, в которой задействуется строго определенное количество межсоединений (глобальных и локальных) и логических элементов, сконфигурированных определенным образом. На рисунке 1 представлена тестовая конфигурация ПЛИС 5576XC2T, предназначенная для контроля длительности тактового интервала межрегистровой пересылки t_{DRR} . Эта конфигурация представляет цепочку, состоящую из пяти логических элементов ЛЭ1 - ЛЭ5, трех глобальных межсоединений строки и четырех локальных межсоединений. Логические элементы ЛЭ1 и ЛЭ5 имеют конфигурацию D-триггера и размещаются таким образом, чтобы обеспечивалось одинаковое время прохождения сигнала от внешних входов С1 и С2 до тактовых входов триггеров, размещенных в этих ЛЭ. Остальные задействованные логические элементы ЛЭ2 - ЛЭ4 имеют конфигурацию буфера и размещаются в соответствии с рисунком 1. Длительность тактового интервала межрегистровой пересылки t_{DRR} устанавливается как предельно-минимальный временной интервал между импульсами С1 и С2, при котором не возникает брака по функционированию схемы.

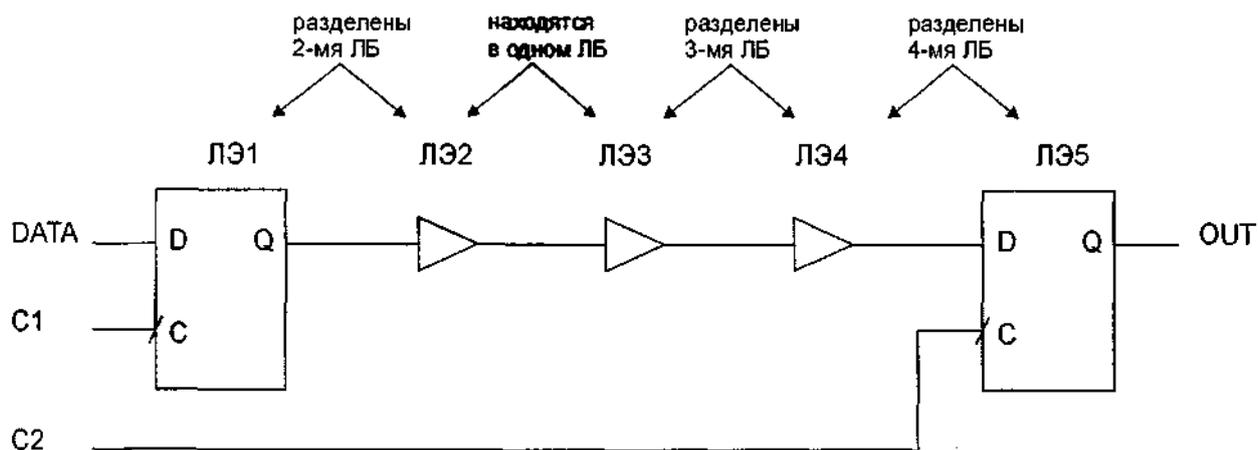


Рисунок 1 - Тестовая конфигурация ПЛИС 5576XC2Т для контроля длительности тактового интервала межрегистровой пересылки t_{DRR}

3.7 Пределно-допустимые и предельные режимы эксплуатации микросхем интегральных 5576XC2Т приведены в таблице 3.

Таблица 3 - Пределно-допустимые и предельные режимы эксплуатации микросхем

Наименование параметра режима, единица измерения	Буквенное обозначение параметра	Пределно-допустимый режим		Пределный режим	
		не менее	не более	не менее	не более
1	2	3	4	5	6
Напряжение питания, В	U_{CC}	3,0	3,6	-0,5	4,5
Входное напряжение низкого уровня, В	U_{IL}	0	0,8	-0,5	-
Входное напряжение высокого уровня, В	U_{IH}	2,0	5,5	-	6,0
Напряжение, прикладываемое к выходу в состоянии «Выключено», В	U_{OZ}	0	5,5	-0,5	6,0
Напряжение, прикладываемое к выходу ¹⁾ , В	U_{OUT}	0	U_{CC}	-0,5	$U_{CC}+0,5$
Выходной ток низкого уровня, мА	I_{OL}	-	24	-	50
Выходной ток высокого уровня, мА	I_{OH}	-24	-	-50	-
Ток по выводу питания ²⁾ , мА	I_{VCC}	-	-	-	100

Окончание таблицы 3

1	2	3	4	5	6
Ток по общему выводу ²⁾ , мА	I_{GND}	-	-	-100	-
¹⁾ В данных режимах выходные токи I_{OL} I_{OH} не должны превышать значений, указанных в настоящей таблице. ²⁾ По каждому выводу.					

4 Условное обозначение и назначение выводов микросхем

4.1 Микросхемы выполнены в планарных металлокерамических 108 - выводных корпусах 4226.108-2 с покрытием на основе золота.

Условное графическое обозначение микросхем 5576XC2Т приведено на рисунке 2.

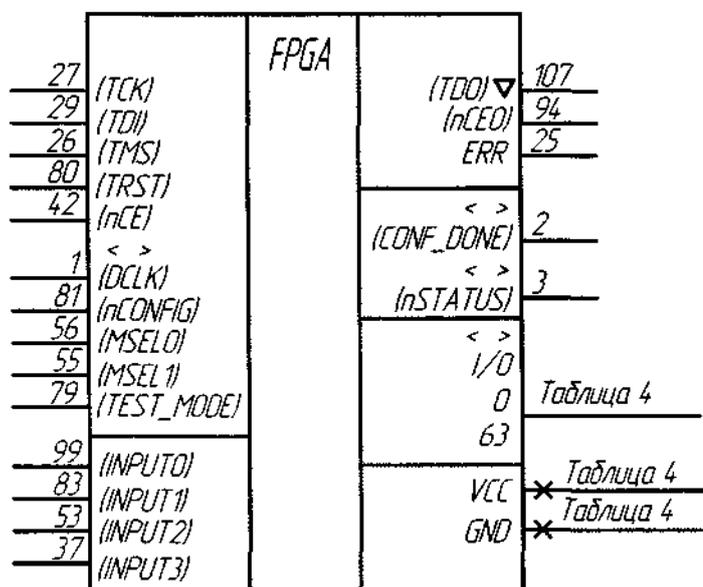


Рисунок 2, лист 1 - Условное графическое обозначение микросхем интегральных 5576XC2Т в корпусе 4226.108-2

- FPGA - программируемая логическая интегральная схема;
- TCK - тактовый вход JTAG;
- TDI - вход данных JTAG;
- TMS - управление состоянием JTAG;
- TRST - асинхронный сброс JTAG;
- nCE - разрешение конфигурирования;
- DCLK - тактовый вход при конфигурировании в режимах PS, PPS и выход в режиме AS;
- nCONFIG - сброс конфигурации;

MSEL - выбор типа конфигурирования;
 TEST_MODE - управление тестовым режимом;
 INPUT - выделенный вход; TDO - выход данных
 JTAG;
 nCEO - разрешение конфигурирования следующей схемы в цепочке;
 CONF_DONE — окончание конфигурирования;
 nSTATUS - готовность к конфигурированию или ошибка при конфигурировании;
 I/O - программируемый ввод-вывод;
 ERR - индикатор сбоя конфигурационного ОЗУ

Таблица 4

Номер вывода	Метка вывода						
4	I/O0	30	I/O 16	57	I/O32	84	I/O48
5	I/O1	31	I/O 17	58	I/O33	85	I/O49
7	I/O2	32	I/O 18	59	I/O34	86	I/O50
8	I/O3	34	I/O 19	62	I/O35	88	I/O51
9	I/O4	35	I/O20	63	I/O36	89	I/O52
10	I/O5	38	I/O21	64	I/O37	91	I/O53
12	I/O6	39	I/O22	66	I/O38	92	I/O54
13	I/O7	40	I/O23	67	I/O39	93	I/O55
15	I/O8	43	I/O24	69	I/O40	96	I/O56
16	I/O9	44	I/O25	70	I/O41	97	I/O57
18	I/O 10	45	I/O26	72	I/O42	98	I/O58
19	I/O11	47	I/O27	73	I/O43	101	I/O59
20	I/O12	48	I/O28	74	I/O44	102	I/O60
21	I/O13	50	I/O29	75	I/O45	104	I/O61
23	I/O 14	51	I/O30	77	I/O46	105	I/O62
24	I/O 15	52	I/O31	78	I/O47	106	I/O63

Примечания 1 Выводы VCC: 6, 14, 22, 33, 41, 49, 60, 68, 76, 87, 95, 103. 2 Выводы GND: 1, 17, 28, 36, 46, 54, 61, 65, 71, 82, 90, 100, 108. 3 Выводы I/O0 - I/O63: 4, 5, 7-10, 12, 13, 15, 16, 18-21, 23, 24, 30-32, 34, 35, 38-40, 43-45, 47, 48, 50-52, 57-59, 62-64, 66, 67, 69, 70, 72-75, 77, 78, 84-86, 88, 89, 91-93, 96-98, 101, 102, 104-106.

4.2 Функциональное назначение выводов микросхем 5576XC2T приведено в таблице 5. Таблица 5 - Назначение выводов микросхем интегральных 5576XC2T

Номер вывода	Обозначение вывода	Функциональное назначение вывода	Тип вывода	Статус вывода
1	DCLK	Тактовый вход при конфигурировании в режимах PS, PPS и выход в режиме AS	Вход-выход	Служебный
2	CONF_DONE	Окончание конфигурирования	Вход-выход	Служебный
3	nSTATUS	Готовность к конфигурированию или ошибка при конфигурировании	Вход-выход	Служебный
4	I/O0 ⁵⁾	Программируемый ввод-вывод	Вход-выход	Пользовательский
5	I/O1 ⁵⁾	Программируемый ввод-вывод	Вход-выход	Пользовательский
6	vcc	Питание		
7	I/O2 ⁵⁾	Программируемый ввод-вывод	Вход-выход	Пользовательский
8	I/O3 ⁵⁾	Программируемый ввод-вывод	Вход-выход	Пользовательский
9	I/O4 ⁵⁾	Программируемый ввод-вывод	Вход—выход	Пользовательский
10	I/O5 ⁵⁾	Программируемый ввод-вывод	Вход-выход	Пользовательский
11	GND	Общий		
12	I/O6	Программируемый ввод-вывод	Вход-выход	Пользовательский
13	I/O7	Программируемый ввод-вывод	Вход-выход	Пользовательский
14	VCC	Питание		
15	I/O8	Программируемый ввод-вывод	Вход-выход	Пользовательский
16	I/O9	Программируемый ввод-вывод	Вход-выход	Пользовательский
17	GND	Общий		
18	I/O 10	Программируемый ввод-вывод	Вход-выход	Пользовательский
19	I/O 11	Программируемый ввод—вывод	Вход—выход	Пользовательский
20	I/O 12	Программируемый ввод-вывод	Вход-выход	Пользовательский

Продолжение таблицы 5

Номер вывода	Обозначение вывода	Функциональное назначение вывода	Тип вывода	Статус вывода
21	I/O 13	Программируемый ввод-вывод	Вход-выход	Пользовательский
22	VCC	Питание		
23	I/O 14	Программируемый ввод-вывод	Вход— выход	Пользовательский
24	I/O 15	Программируемый ввод-вывод	Вход-выход	Пользовательский
25	ERR	Индикатор сбоя конфигурационного ОЗУ	Выход	Служебный
26	TMS	Управление состоянием JTAG	Вход	Служебный
27	ТСл	Тактовый вход JTAG	Вход	Служебный
28	GND	Общий		
29	TDI	Вход данных JTAG	Вход	Служебный
30	I/O 16	Программируемый ввод-вывод	Вход-выход	Пользовательский
31	I/O 17	Программируемый ввод-вывод	Вход-выход	Пользовательский
32	I/O 18	Программируемый ввод-вывод	Вход-выход	Пользовательский
33	VCC	Питание		
34	I/O 19	Программируемый ввод-вывод	Вход-выход	Пользовательский
35	I/O20	Программируемый ввод-вывод	Вход-выход	Пользовательский
36	GND	Общий		
37	INPUT3	Выделенный вход	Вход	Пользовательский
38	I/O21	Программируемый ввод-вывод	Вход-выход	Пользовательский
39	I/O22	Программируемый ввод-вывод	Вход-выход	Пользовательский
40	I/O23	Программируемый ввод-вывод	Вход-выход	Пользовательский
41	VCC	Питание		
42	пСЕ	Разрешение конфигурирования	Вход	Служебный
43	I/O24	Программируемый ввод-вывод	Вход-выход	Пользовательский
44	I/O25	Программируемый ввод-вывод	Вход-выход	Пользовательский
45	I/O26	Программируемый ввод-вывод	Вход-выход	Пользовательский

Продолжение таблицы 5

Номер вывода	Обозначение вывода	Функциональное назначение вывода	Тип вывода	Статус вывода
46	GND	Общий		
47	I/O27	Программируемый ввод-вывод	Вход-выход	Пользовательский
48	I/O28	Программируемый ввод-вывод	Зход-выход	Пользовательский
49	VCC	Питание		
50	I/O29	Программируемый ввод-вывод	Вход-выход	Пользовательский
51	I/O30	Программируемый ввод-вывод	Вход-выход	Пользовательский
52	I/O31	Программируемый ввод-вывод	Вход-выход	Пользовательский
53	INPUT2	Выделенный вход	Вход	Пользовательский
54	GND	Общий		
55	MSEL1	Выбор типа конфигурирования	Вход	Служебный
56	MSEL0	Выбор типа конфигурирования	Вход	Служебный
57	I/O32	Программируемый ввод-вывод	Вход-выход	Пользовательский
58	I/O33 ²⁾	Программируемый ввод-вывод	Вход-выход	Пользовательский
59	I/O34	Программируемый ввод-вывод	Вход-выход	Пользовательский
60	VCC	Питание		
61	GND	Общий		
62	I/O35	Программируемый ввод-вывод	Вход-выход	Пользовательский
63		Программируемый ввод-вывод	Вход-выход	Пользовательский
64	I/O37	Программируемый ввод-вывод	Вход-выход	Пользовательский
65	GND	Общий		
66	I/O38	Программируемый ввод-вывод	Вход-выход	Пользовательский
67	I/O39	Программируемый ввод-вывод	Вход-выход	Пользовательский
68	VCC	Питание		

Продолжение таблицы 5

Номер вывода	Обозначение вывода	Функциональное назначение вывода	Тип вывода	Статус вывода
69	I/O40	Программируемый ввод-вывод	Вход-выход	Пользовательский
69	I/O40	Программируемый ввод-вывод	Вход-выход	Пользовательский
70	I/O41	Программируемый ввод— вывод	Вход-выход	Пользовательский
71	GND	Общий		
72	I/O42	Программируемый ввод— вывод	Вход-выход	Пользовательский
73	I/O43	Программируемый Вход-выход ввод-вывод		Пользовательский
74	I/O44	Программируемый Вход-выход ввод-вывод		Пользовательский
75	I/O45	Программируемый ввод-вывод	Вход-выход	Пользовательский
76	VCC	Питание		
77	I/O46	Программируемый ввод-вывод	Вход-выход	Пользовательский
78	I/O47	Программируемый ввод-вывод	Вход-выход	Пользовательский
79	TEST_MODE ¹⁾	Управление тестовым режимом	Вход	Служебный
80	TRST	Асинхронный сброс JTAG	Вход	Служебный
81	nCONFIG	Сброс конфигурации	Вход	Служебный
82	GND	Общий		
83	INPUT 1	Выделенный вход	Вход	Пользовательский
84	I/O48	Программируемый ввод— вывод	Вход-выход	Пользовательский
85	I/O49	Программируемый ввод-вывод	Вход-выход	Пользовательский
86	I/O50	Программируемый ввод-вывод	Вход-выход	Пользовательский
87	VCC	Питание		
88	I/O51	Программируемый ввод-вывод	Вход-выход	Пользовательский

Продолжение таблицы 5

Номер вывода	Обозначение вывода	Функциональное назначение вывода	Тип вывода	Статус вывода
89	I/O52	Программируемый ввод-вывод	Вход-выход	Пользовательский
90	GND	Общий		
91	I/O53	Программируемый ввод-вывод	Вход-выход	Пользовательский
92	I/O54	Программируемый ввод— вывод	Вход-выход	Пользовательский
93	I/O55 ³⁾	Программируемый ввод-вывод	Вход-выход	Пользовательский
94	пCEO	Разрешение конфигурирования следующей схемы в цепочке	Выход	Служебный
95	VCC	Питание		
96	I/O56 ⁴⁾	Программируемый ввод-вывод	Вход-выход	Пользовательский
97	I/O57	Программируемый ввод-вывод	Вход-выход	Пользовательский
98	I/O58	Программируемый ввод-вывод	Вход-выход	Пользовательский
99	INPUT0	Выделенный вход	Вход	Пользовательский
100	GND	Общий		
101	I/O59	Программируемый ввод-вывод	Вход-выход	Пользовательский
102	I/O60	Программируемый ввод— вывод	Вход— выход	Пользовательский
103	VCC	Питание		
104	I/O61	Программируемый ввод-вывод	Вход-выход	Пользовательский
105	I/O62 ⁵⁾	Программируемый ввод-вывод	Вход-выход	Пользовательский
106	I/O63 ⁵⁾	Программируемый ввод— вывод	Вход-выход	Пользовательский
107	TDO	Выход данных JTAG	Выход	Служебный
108	GND	Общий		

- ¹⁾ Сигнал TEST_MODE предназначен для использования в режиме тестирования. В пользовательском режиме соответствующий ему 79 вывод должен быть соединен с общим выводом.
- ²⁾ Вывод I/O33 может использоваться как служебный CLKUSR, если это указано пользователем.
- ³⁾ После выхода ПЛИС в рабочий режим вывод I/O55 может использоваться как глобальный сигнал разрешения третьего состояния всех элементов ввода-вывода DEV OE, если это указано пользователем.
- ⁴⁾ После выхода ПЛИС в рабочий режим вывод I/O56 может использоваться как глобальный сигнал сброса всех триггеров DEV CLRn, если это указано пользователем.
- ⁵⁾ При конфигурировании ПЛИС в режиме PPS (пассивный параллельный синхронный) выводы I/O62, I/O63, I/O00, I/O01, I/O02, I/O03, I/O04, I/O05 являются входами данных DATA0, DATA1, DATA2, DATA3, DATA4, DATA5, DATA6, DATA7 соответственно. Кроме этого, вывод I/O62 является входом данных DATA0 при конфигурировании ПЛИС в режимах PS (пассивный последовательный) и AS (активный последовательный). После конфигурирования указанные выводы могут быть пользовательскими согласно

5 Описание архитектуры ПЛИС

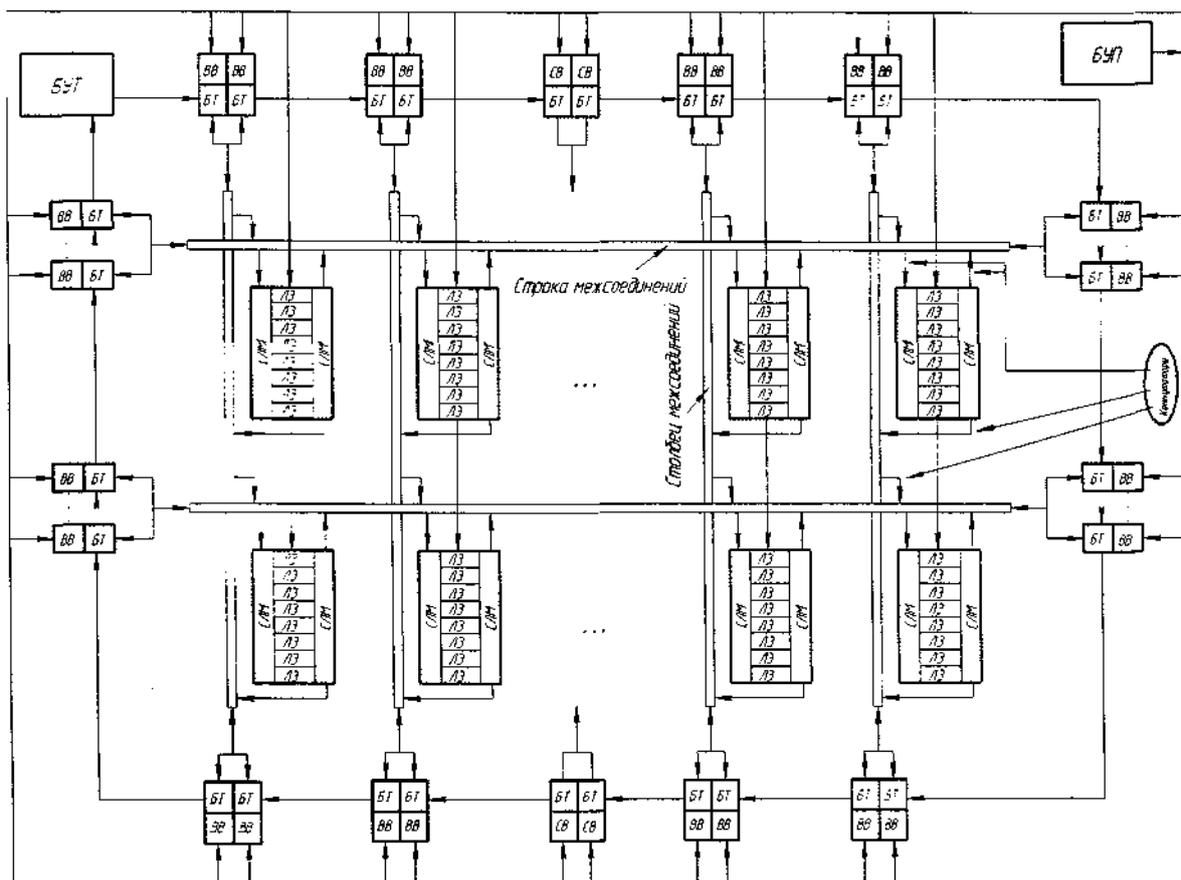
ПЛИС 5576XC2T - ИС со встроенной программируемой логикой, с возможностью реконфигурации в составе аппаратуры. Такая возможность позволяет на 100 % протестировать ИС до ее использования потребителем и делает не обязательной для разработчика генерацию векторов для тестирования по выявлению конкретных ошибок. Базой реконфигурируемых элементов служат 37524 ячейки статического запоминающего устройства (СОЗУ), распределенные по площади микросхемы. В основе архитектуры ПЛИС 5576XC2T лежат следующие компоненты: логические элементы (ЛЭ), объединенные в логические блоки (ЛБ), блоки формирования глобальных сигналов управления, система межсоединений, а также обслуживающие блоки: блоки тестирования и программирования. ЛЭ объединены в группы по 8 - логические блоки, число которых составляет 26. ЛБ составляют матрицу из 2 строк и 13 столбцов. Блок управления программированием вместе с блоками программирования образуют систему конфигурирования ПЛИС, основанную на элементах СОЗУ. Встроенный блок управления тестированием (БУТ) с использованием схемы периферийного сканирования, включающей блоки тестирования (БТ), составляют систему тестирования JTAG (Joint Test Action Group), совместимую со стандартом IEEE Std. 1149.1-1990.

Вокруг матричной части ПЛИС расположены общие выводы и выводы питания, служебные выводы, 64 пользовательских элемента ввода-вывода и 4 выделенных (специализированных) входа, которые можно использовать в качестве формирователей глобальных сигналов синхронизации. ЛБ и элементы ввода-вывода связаны друг с другом системой межсоединений.

Загрузка конфигурационных данных в ПЛИС 5576XC2T возможна в системе с использованием специальных микросхем памяти (ПЗУ) EPС2 (ф. Altera) и других, с помощью загрузочного кабеля BitBlaster (ф. Altera) по последовательному порту ПК, с помощью загрузочного кабеля VytBlasterMV (ф. Altera) через параллельный порт ПК или с использованием микропроцессора.

Конфигурационные данные могут загружаться в ПЛИС как через специализированный загрузочный порт, так и через порт JTAG. После конфигурирования ПЛИС может быть переконфигурирована путем загрузки новых данных. Реконфигурация занимает менее 100 мс при частоте 10 МГц. Кроме того, с использованием порта JTAG может быть выполнено тестирование ПЛИС 5576XC2T как до, так и после процесса конфигурирования.

Структурная схема ПЛИС 5576XC2T приведена на рисунке 3.



- БУП - блок управления программированием (DOWNLOAD - контроллер, JTAG - контроллер);
- БУТ - блок управления тестированием (JTAG - контроллер);
- СВ - специализированный вход;
- БТ - блок тестирования;
- ВВ - элемент ввода-вывода;
- ЛЭ - логический элемент;
- СЛМ - система локальных межсоединений логического блока

Рисунок 3 - Схема электрическая структурная ПЛИС 5576XC2T

5.1 Логический блок

ЛБ является основным элементом крупнозернистой архитектуры, основанной на эффективной трассировке с оптимальным использованием элементов и высокой производительностью, и служит для обеспечения эффективного размещения большинства проектов на ПЛИС.

ЛБ имеет систему локальных межсоединений (СЛМ), выводы управления триггерами ЛБ, каналы переноса и каскадирования. ЛБ подключен к системе межсоединений 24 входными и 8 выходными каналами. Управление триггерами, расположенными в ЛБ, осуществляется по четырем сигналам с программируемой инверсией, каждый из которых может быть использован во всех восьми ЛЭ. Два из этих сигналов используются как тактовые, остальные - как сигналы сброса и установки. Источником управляющих сигналов могут быть выделенные (специализированные) входы, сигналы с элементов ввода-вывода или внутренние сигналы СЛМ ЛБ. Как правило, в качестве внешних источников глобальных тактовых сигналов или сигналов сброса и установки используются выделенные (специализированные) входы. Структурная схема ЛБ приведена на рисунке 4.

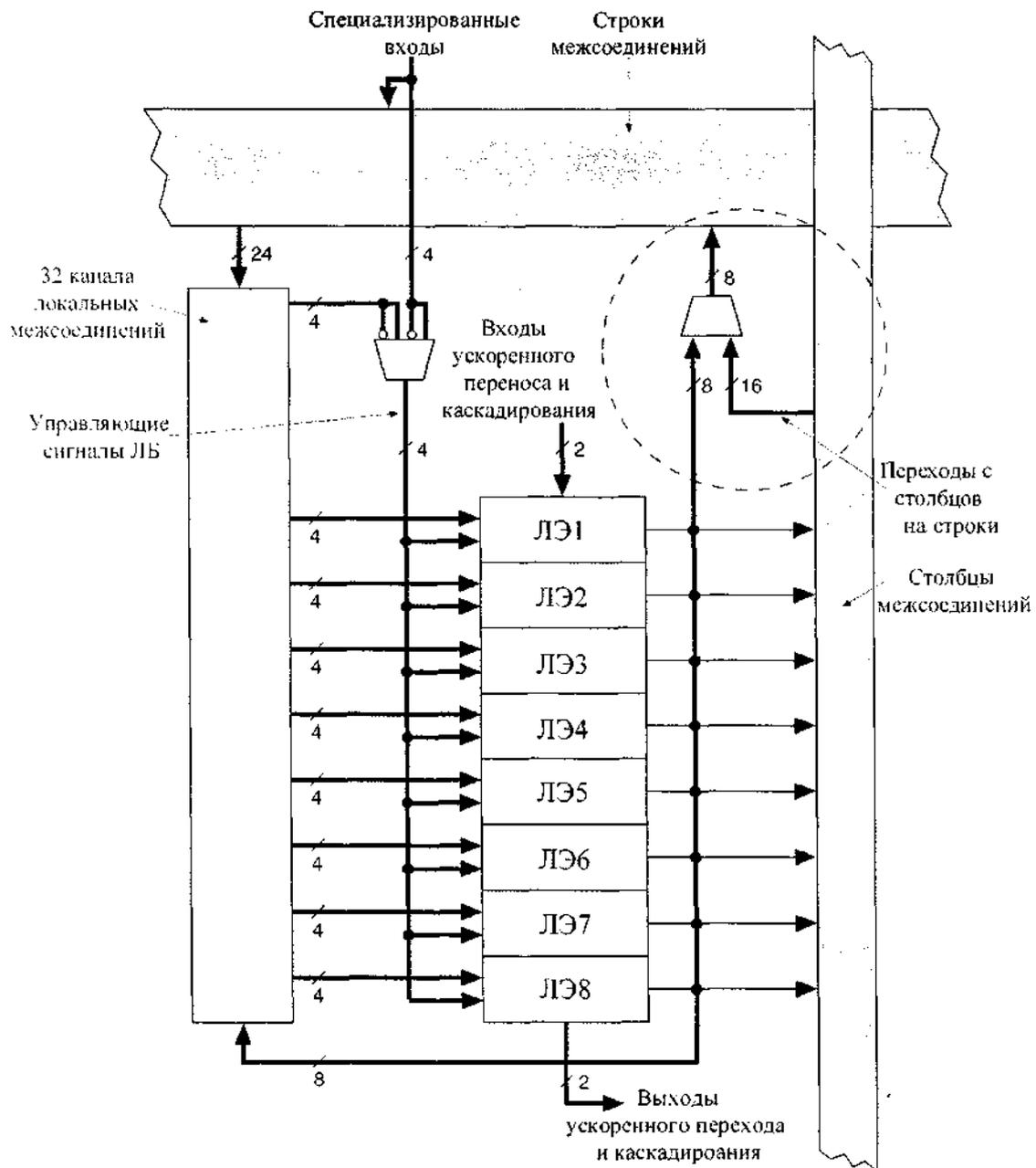


Рисунок 4 - Структурная схема логического блока

5.2 Логический элемент

ЛЭ является минимальным элементом логики, имеет компактные размеры и предоставляет эффективные средства для реализации различных логических функций. Каждый ЛЭ содержит четырехвходовую таблицу преобразования (lookup table - LUT), которая представляет собой генератор функций (ГФ), способный реализовать функцию четырех переменных. ГФ четырех переменных может быть представлен как два ГФ трех переменных и мультиплексор 2 в 1. Кроме того, каждый ЛЭ содержит программируемый триггер, цепи переноса и каскадирования, а также имеет выход как на локальные межсоединения ЛБ, так и на систему глобальных межсоединений. Структурная схема ЛЭ приведена на рисунке 5.

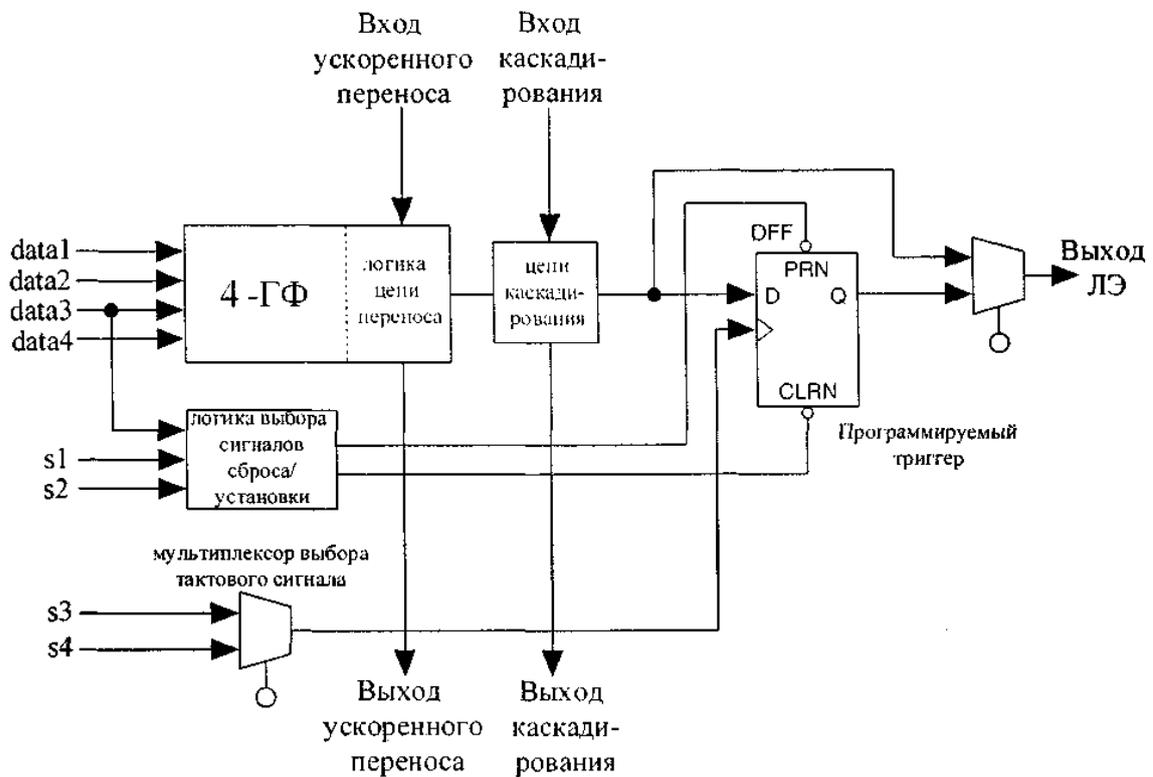


Рисунок 5 - Структурная схема логического элемента

Программируемый триггер ЛЭ может быть сконфигурирован как триггер D, T, JK или SR типа. Источником сигналов управления триггером (тактовый сигнал, сброс, установка) могут быть выделенные (специализированные) входы, сигналы с элементов ввода-вывода общего назначения или сигналы с внутренней логики. При реализации комбинаторных функций триггер не используется и сигнал с выхода ГФ будет поступать на выход ЛЭ.

Цепь переноса представляет собой быстродействующую функцию переноса между ЛЭ. Сигнал с входа цепи переноса передается от младших битов к старшим через цепь переноса и подается на ГФ и следующую часть цепи переноса. Эта функция позволяет реализовывать быстродействующие счетчики и сумматоры произвольной ширины.

С помощью цепи каскадирования реализуются функции, имеющие большой коэффициент объединения по входу. Соседние ЛЭ могут быть использованы для параллельного вычисления частей функции, цепь каскадирования последовательно соединяет промежуточные результаты. Цепь каскадирования может использовать логические функции «И» и «ИЛИ» для соединения входов соседних ЛЭ. Каждый дополнительный ЛЭ добавляет четыре входа к эффективной ширине функции.

Цепи каскадирования (переноса), объединяющие более восьми ЛЭ, реализуются путем соединения соседних ЛБ. В этом случае, восьмой ЛЭ предыдущего ЛБ соединяется с первым ЛЭ последующего ЛБ.

5.2.1 Режимы работы логического элемента

ЛЭ может работать в одном из следующих четырех режимов:

- нормальный режим;
- арифметический (счетный) режим;
- режим суммирующего/вычитающего счетчика;
- режим сбрасываемого счетчика.

5.2.1.1 Нормальный режим

Нормальный режим работы (рисунок 6) подходит для большинства логических приложений и функций декодирования, использующих

преимущества цепи каскадирования. В данном режиме четыре сигнала data1, data2, data3, data4 или data1, data2, data4 и сигнал с входа цепи переноса являются входными для ГФ. Выход ГФ может быть скомбинирован с сигналом с входа цепи каскадирования для формирования цепи каскадирования. В зависимости от состояния программируемого мультиплексора, выходным сигналом ЛЭ является либо этот комбинированный сигнал, либо выходные данные программируемого триггера.

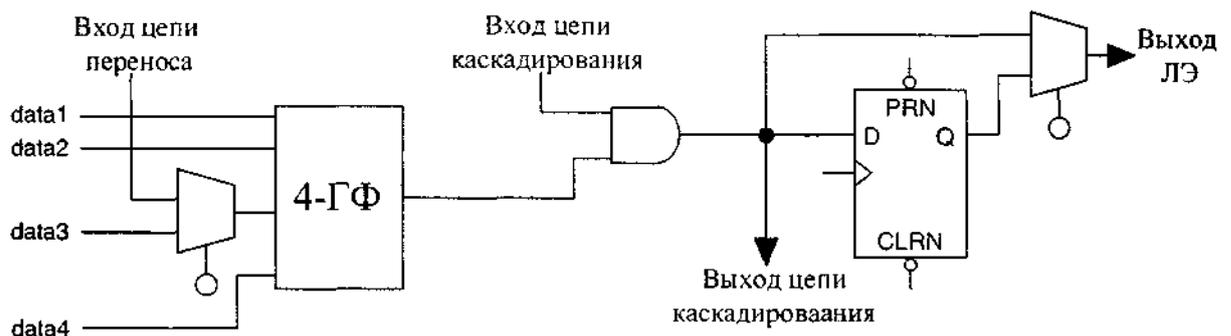


Рисунок 6 - Нормальный режим работы логического элемента

5.2.1.2 Арифметический режим

Арифметический режим (рисунок 7) предназначен для реализации сумматоров, компараторов и т.п. В данном режиме сигналы data1, data2 и сигнал с входа цепи переноса являются входными сигналами для двух трехвходовых ГФ, один из которых формирует сигнал цепи переноса, а другой реализует функцию трех переменных.

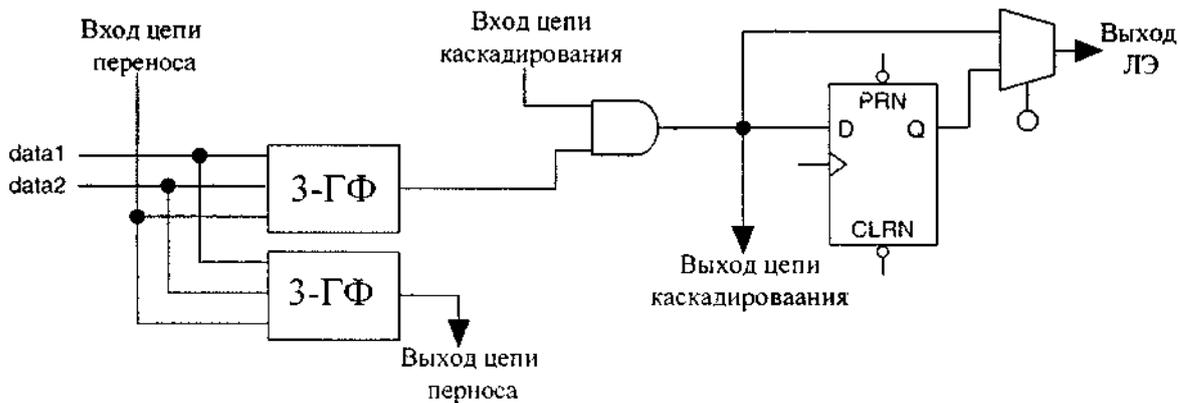


Рисунок 7 - Арифметический режим работы логического элемента

5.2.1.3 Режим суммирующего/вычитающего счетчика В данном режиме (рисунок 8) используются два трехвходовых ГФ. Один из них формирует данные счетчика с помощью сигналов управления *data1*, обратной связи триггера, входа цепи переноса. Другой формирует сигнал переноса с помощью сигналов управления обратной связи триггера, *data2*, входа цепи переноса. Сигнал *data4* предназначен для реализации возможности синхронной загрузки данных. Загрузка данных может быть асинхронной без использования ресурсов ГФ с помощью сигналов сброса и установки триггера. Функцию сброса в данном режиме выполняет входной сигнал цепи каскадирования.

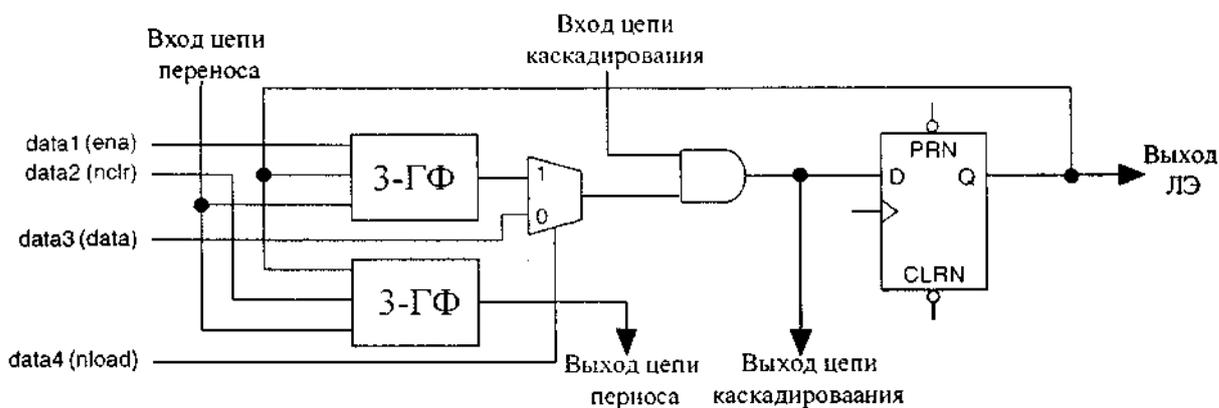


Рисунок 8 - Режим суммирующего/вычитающего счетчика

5.2.1.4 Режим сбрасываемого счетчика

Режим сбрасываемого счетчика (рисунок 9) подобен режиму суммирующего/вычитающего счетчика, но в отличие от него синхронный сброс осуществляется с входа data2.

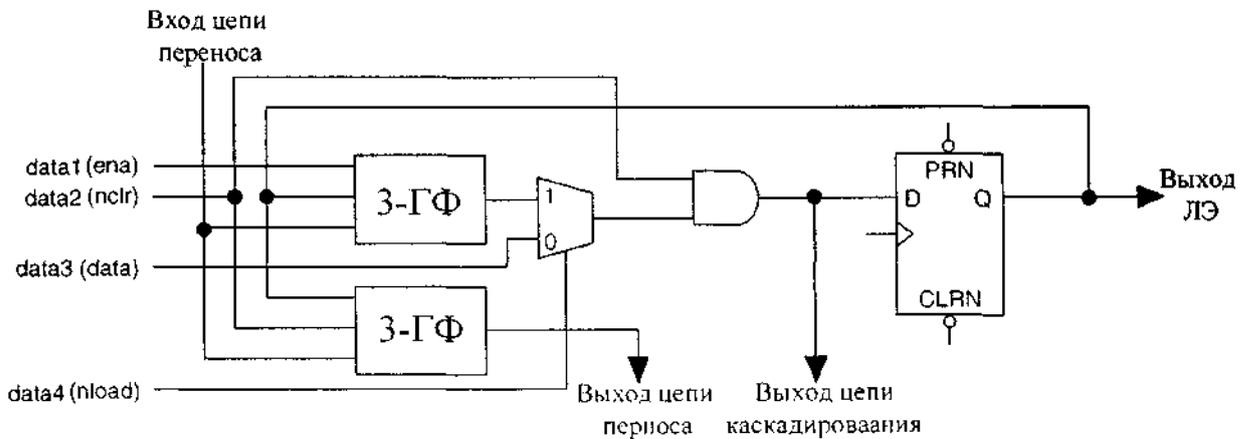


Рисунок 9 — Режим сбрасываемого счетчика

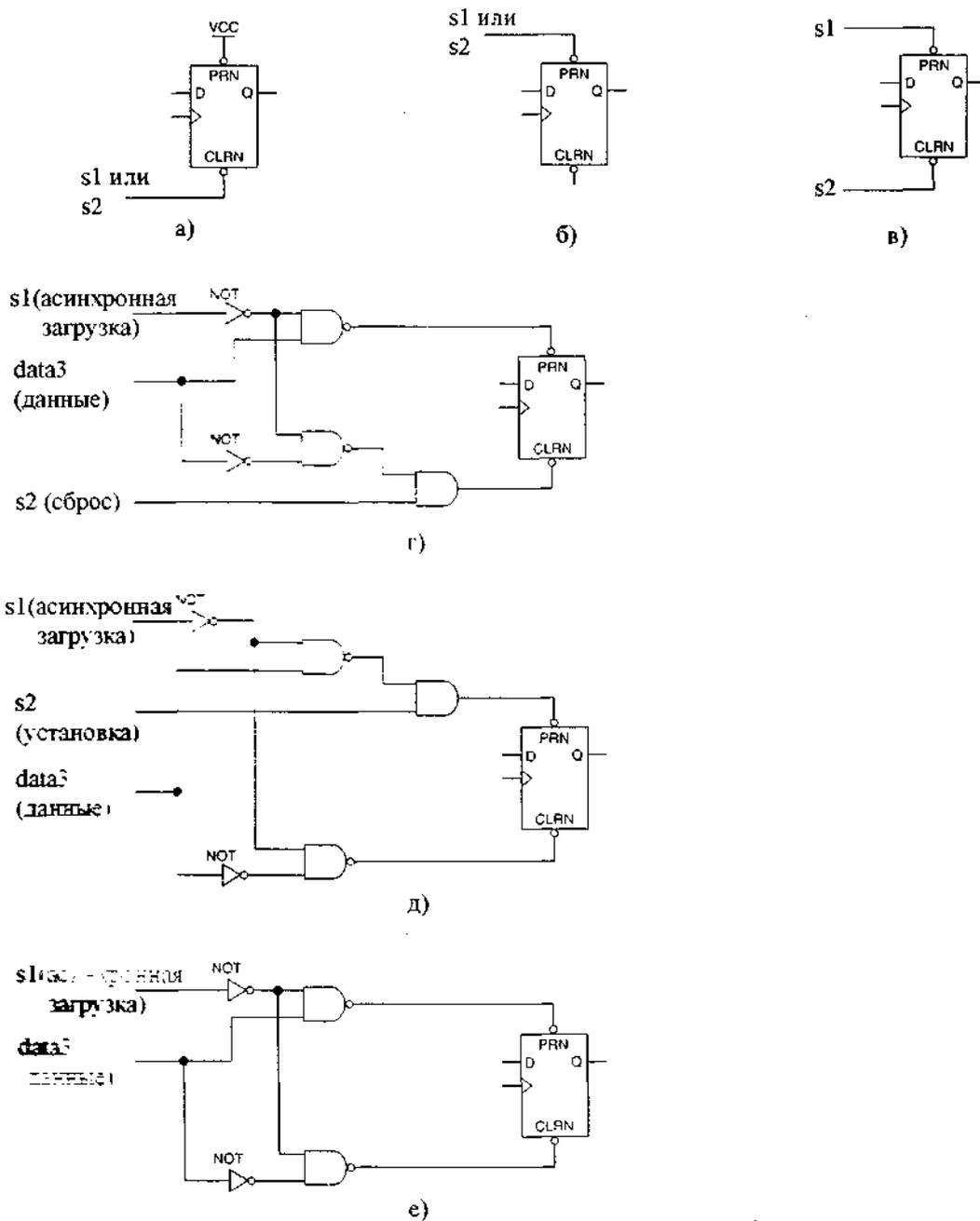
5.2.2 Управление логикой сброса и установки

Управление логикой сброса и установки программируемого триггера осуществляется с помощью сигналов data3, si, s2 ЛЭ.

Логика сброса и установки выполняется в следующих шести режимах, выбираемых на стадии введения данных в проект:

- асинхронный сброс;
- асинхронная установка;
- асинхронный сброс и установка;
- асинхронная загрузка со сбросом;
- асинхронная загрузка с установкой.
- асинхронная загрузка без сброса или без установки.

В зависимости от выбранного режима сигналы si, s2 выполняют конкретную функцию в соответствии с рисунком 10.



- а) асинхронный сброс;
- б) асинхронная установка;
- в) асинхронный сброс и установка;
- г) асинхронная загрузка со сбросом;
- д) асинхронная загрузка с установкой;
- е) асинхронная загрузка без сброса или без установки.

Рисунок 10 - Управление логикой сброса и установки

5.3 Система межсоединений

В архитектуре ПЛИС 5576XC2T связь между ЛЭ и элементами ввода-вывода осуществляются посредством FastTrack соединений (горизонтальных (строк) и вертикальных (столбцов)), которые связывают устройство в единое целое. При такой структуре трассировки ЛЭ соединяются друг с другом при помощи одного из каналов, что обеспечивает минимальную задержку даже при реализации сложных проектов. В противоположность этому, сегментная трассировка с помощью матричных перемычек требует соединения переменного числа трассировочных путей, что приводит к увеличению задержек между логическими ресурсами и, как следствие, снижению производительности.

Каждая строка FastTrack соединений состоит из 168 каналов и предназначена для управления элементами ввода-вывода и передачи информации в другие ЛБ схемы. Столбцы межсоединений, содержащие по 16 каналов, осуществляют связь между строками и также могут управлять элементами ввода-вывода. На рисунке 11 показано как ЛЭ подключен к горизонтальным и вертикальным каналам.

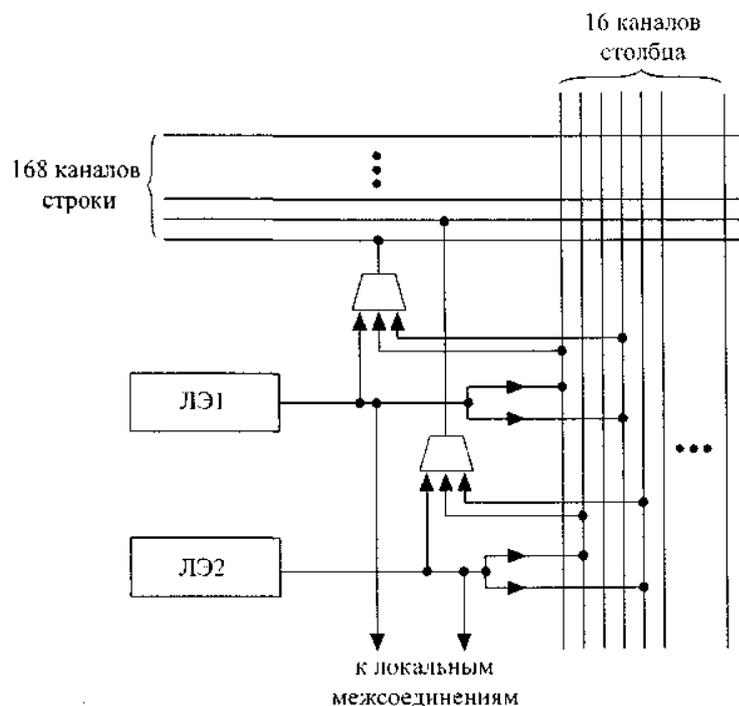


Рисунок 11 - Подключение ЛЭ к строкам и столбцам межсоединений

Каждый ЛЭ в ЛБ может быть подключен к одному из двух различных каналов столбца межсоединений. Программное обеспечение на этапе компиляции проекта автоматически выбирает, к какому каналу должен быть подключен ЛЭ. К каналу горизонтальной строки межсоединений через мультиплексор 3 в 1 могут быть подключены выход ЛЭ или один из двух вертикальных каналов. Таким образом, в каждом ЛБ мультиплексируются все 16 каналов столбца с доступом к 8 каналам горизонтальной строки межсоединений.

5.4 Элемент ввода-вывода

Для сопряжения внутренней части ПЛИС с большинством микросхем цифровой логики используется элемент ввода-вывода. Он содержит двунаправленный буфер и триггер, который можно использовать как входной регистр для внешних данных с малым временем предустановки или как выходной регистр с малой задержкой от тактового входа до выхода. Для управления триггером можно использовать сигналы с шестиразрядной шины управления элементами ввода-вывода, которая может быть сконфигурирована для поддержки до четырех сигналов разрешения вывода и до двух тактовых сигналов или сигналов сброса. Каждый элемент ввода-вывода может быть сконфигурирован как вход, выход или двунаправленный вход-выход. Схема элемента ввода—вывода приведена на рисунке 12.

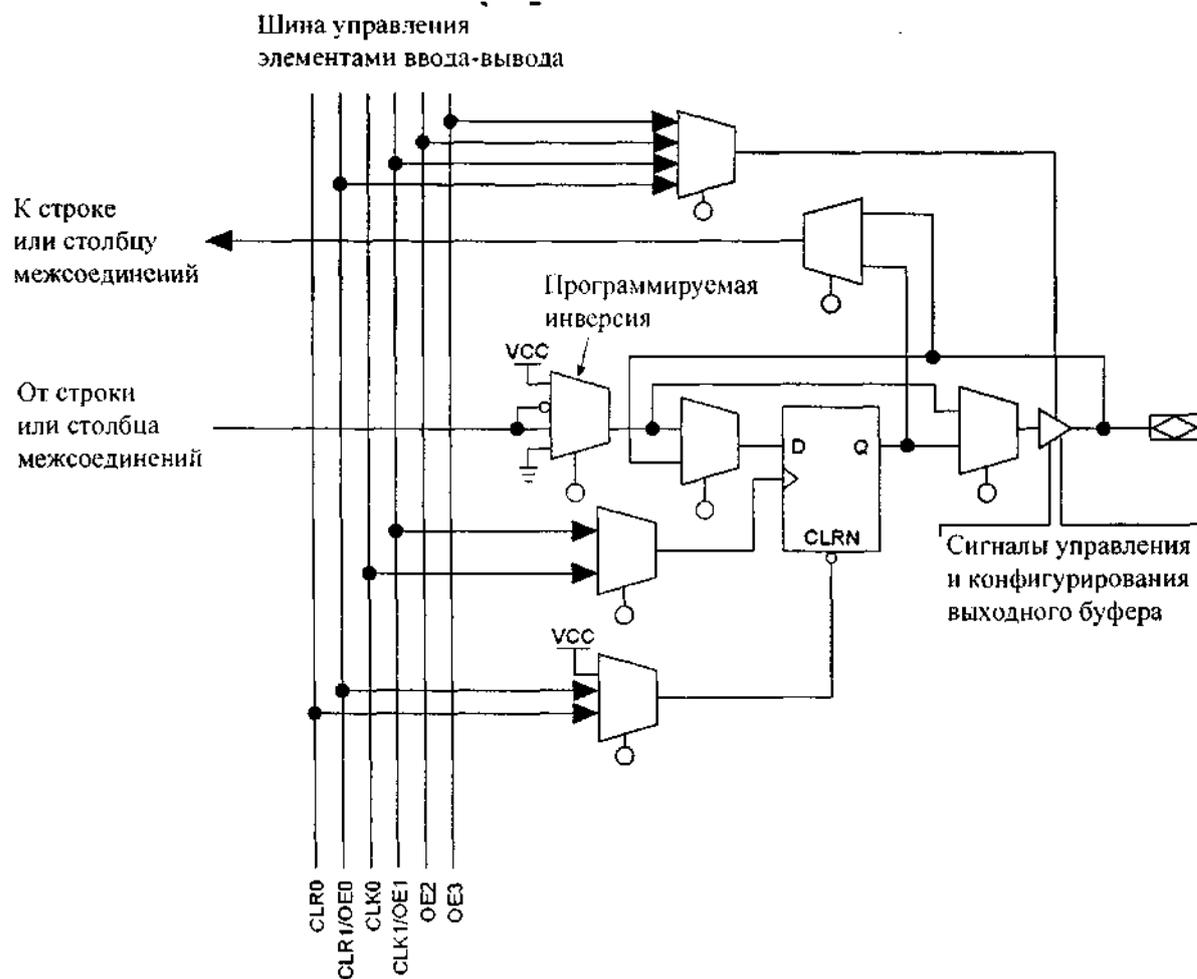


Рисунок 12 –Схема элемента ввода-вывода

Для повышения функциональности ПЛИС реализованы следующие режимы элементов ввода-вывода:

- программируемая нагрузочная способность выходных буферов (12/24 мА) по высокому и низкому уровням сигнала;
- снижение помех при переключениях за счет программируемой опции «slew-rate control»;
- программируемые опции «bus hold», «pull-up», «pull-down».

Для реализации этих режимов в конфигурационной памяти предусмотрены по четыре ячейки на каждый элемент ввода-вывода, обеспечивающие программирование этих функций.

5.4.1 Соединение элементов ввода-вывода с горизонтальными каналами На рисунке 13 показана связь между элементами ввода-вывода и строкой межсоединений. Когда элемент ввода-вывода используется в качестве входа, то он может управлять двумя каналами строки, которые доступны всем ЛЭ данной строки. Когда элемент ввода-вывода используется как выход, то сигнал подводится через мультиплексор 13 в 1, который выбирает канал из строки. К каждой строке межсоединений подключено по восемь элементов ввода-вывода с каждой стороны кристалла.

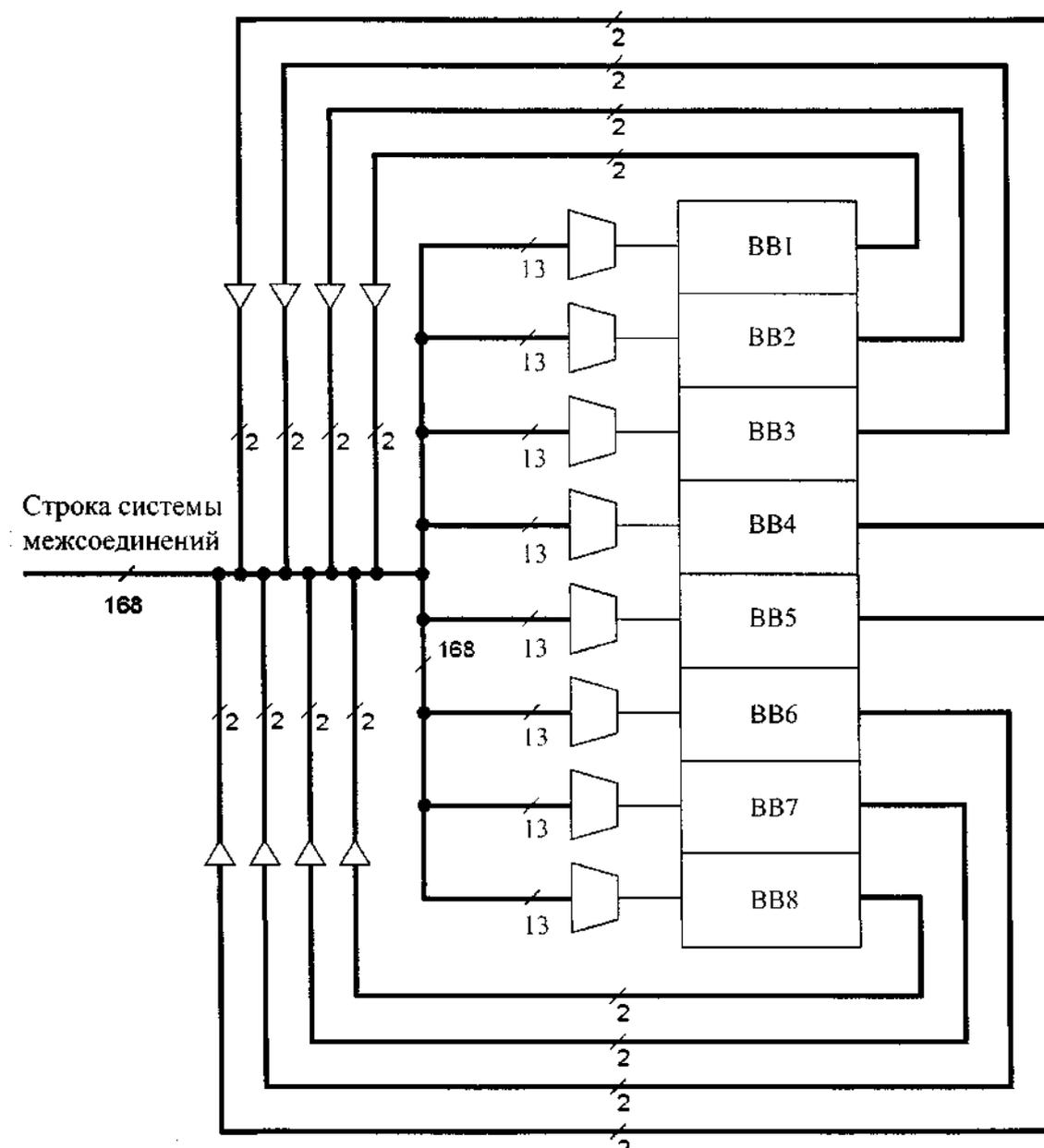


Рисунок 13 - Коммутация элементов ввода-вывода и строки межсоединений

5.4.2 Соединение элементов ввода-вывода с вертикальными каналами На рисунке 14 показано соединение между вертикальным столбцом межсоединений и элементами ввода-вывода. К каждому вертикальному столбцу межсоединений подключено по два элемента ввода-вывода с верхней и нижней сторон кристалла. Когда элемент ввода-вывода используется в качестве входа, то он может управлять двумя отдельными каналами столбца. Когда элемент ввода-вывода используется как выход, то сигнал подводится через мультиплексор 8 в 1, который выбирает канал из столбца.

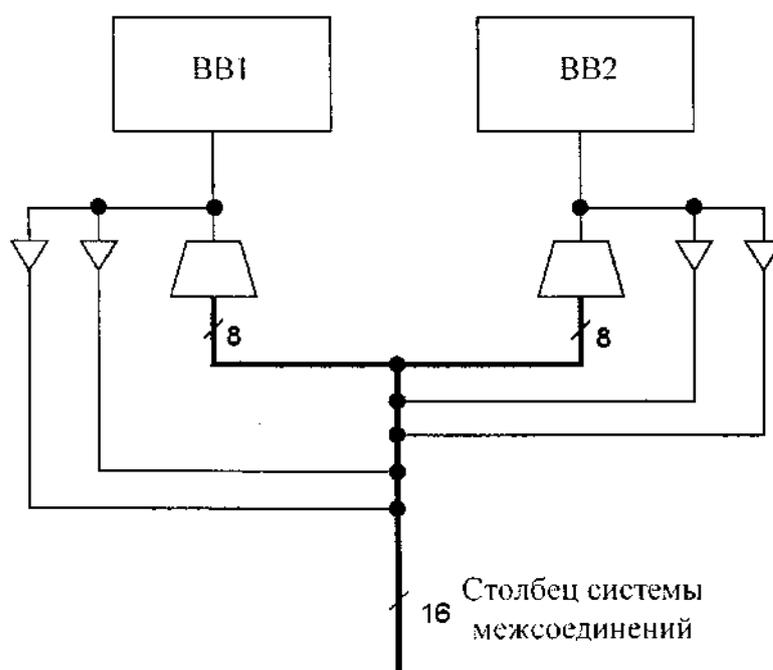


Рисунок 14 - Коммутация элементов ввода-вывода и столбца межсоединений.

5.4.3 Выделенные (специализированные) входы

В дополнение к элементам ввода-вывода общего назначения, ПЛИС 5576XC2T имеет четыре выделенных (специализированных) входа. Эти выделенные (специализированные) входы, обеспечивающие низкий уровень искажений и распространение сигнала по всему кристаллу, обычно используются в качестве глобальных тактовых сигналов и сигналов управления сбросом и

установкой. Сигналы с выделенных (специализированных) входов доступны как управляющие сигналы для триггеров, расположенных в ЛБ и элементах ввода-вывода ПЛИС. Также эти входы могут быть использованы как входы данных общего назначения для ГФ, так как сигналы с них могут быть переданы в СЛМ каждого ЛБ ПЛИС.

Внешние сигналы в ПЛИС 5576XC2T могут подаваться либо через элементы ввода-вывода общего назначения, либо через четыре выделенных (специализированных) входа.

Тактовые сигналы и сигналы управления сбросом, установкой и разрешением вывода для всех элементов ввода-вывода подаются по шестиразрядной шине управления. На эту шину сигнал может быть скоммутирован либо с четырех выделенных (специализированных) входов, либо с 13 каналов строки. Источником сигнала для этих каналов является первый ЛЭ каждого ЛБ. Все управляющие сигналы буферизуются на высокоскоростных элементах, что позволяет достичь минимального искажения данных сигналов при прохождении через ПЛИС. Шина управления элементами ввода—вывода может быть сконфигурирована для поддержки до четырех сигналов разрешения вывода и до двух тактовых сигналов или сигналов сброса. На рисунке 15 показана коммутация шины управления элементами ввода-вывода.

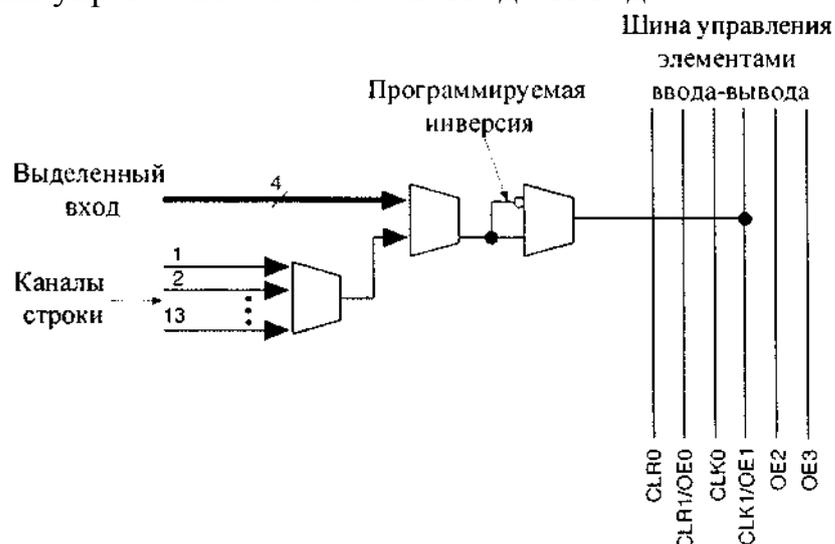


Рисунок 15 - Генерация сигнала шины управления элементами ввода-вывода

5.5 Поддержка периферийного сканирования

ПЛИС 5576XC2T поддерживает схему теста периферийного сканирования JTAG BST, которая удовлетворяет техническим требованиям стандарта IEEE Std. 1149.1-1990. ПЛИС 5576XC2T также может быть сконфигурирована через JTAG выходы при помощи последовательного кабеля загрузки BitBlaster через последовательный порт ПК, через параллельный порт с использованием загрузочного кабеля ByteBlasterMV или с помощью аппаратного обеспечения, которое использует язык Jam™ для программирования и тестирования. JTAG тестирование может быть выполнено до или после конфигурирования, но не во время него.

Архитектура схемы периферийного сканирования (СПС) позволяет тестировать соединения между выводами ПЛИС без проведения физических измерений и фиксировать функциональные данные, пока устройство функционирует нормально. Ячейка СПС может перегружать сигналы на выводах или захватывать данные с выводов или сигналы внутренней логики (ядра ПЛИС). Перегружаемые тестовые данные последовательно сдвигаются через ячейки СПС, захваченные данные последовательно выводятся из схемы и сравниваются с ожидаемым результатом. СПС ПЛИС 5576XC2T соответствует техническим требованиям стандарта IEEE Std. 1149.1-1990. Рисунок 16 показывает принцип работы СПС.

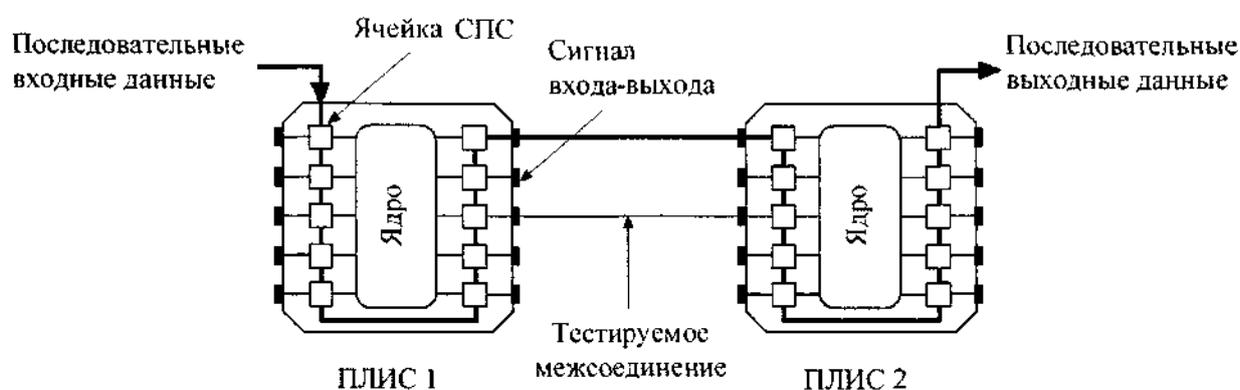


Рисунок 16 - Принцип работы схемы периферийного сканирования

JTAG интерфейс ПЛИС 5576XC2T реализован с помощью пяти выводов TDI, TDO, TMS, TCK и TRST. Описание и назначение этих выводов приведено в таблице 6.

Таблица 6

Вывод	Описание	Назначение
TDI	Вход данных JTAG	Последовательный вход данных для тестирования и программирования. Данные захватываются по переднему фронту сигнала TCK.
TDO	Выход данных JTAG	Последовательный выход данных для тестирования и программирования. Данные выводятся по заднему фронту сигнала TCK. Когда выход не используется, он переключается в третье состояние.
TMS	Управление состоянием JTAG	Вход управляющего сигнала, который определяет переходы автомата состояний TAP – контроллера. Сигнал TMS должен быть установлен до срабатывания переднего фронта TCK.
TCK	Тактовый вход JTAG	Вход тактового сигнала. Некоторые операции совершаются по переднему фронту этого сигнала, другие – по заднему.
TRST	Асинхронный сброс JTAG	Вход асинхронного сброса в режиме тестирования с активным низким уровнем.

5.5.1 Функциональная модель СПС

Работа СПС управляется блоком управления тестированием (TAP - контроллером). Сигналы с входов TMS, TRST и TCK управляют TAP - контроллером, выходы TDI и TDO осуществляют ввод и вывод данных в регистры. Кроме того, вход TDI обеспечивает ввод данных в регистр команд, который затем формирует управляющие сигналы для регистров данных. Функциональная схема СПС приведена на рисунке 17.

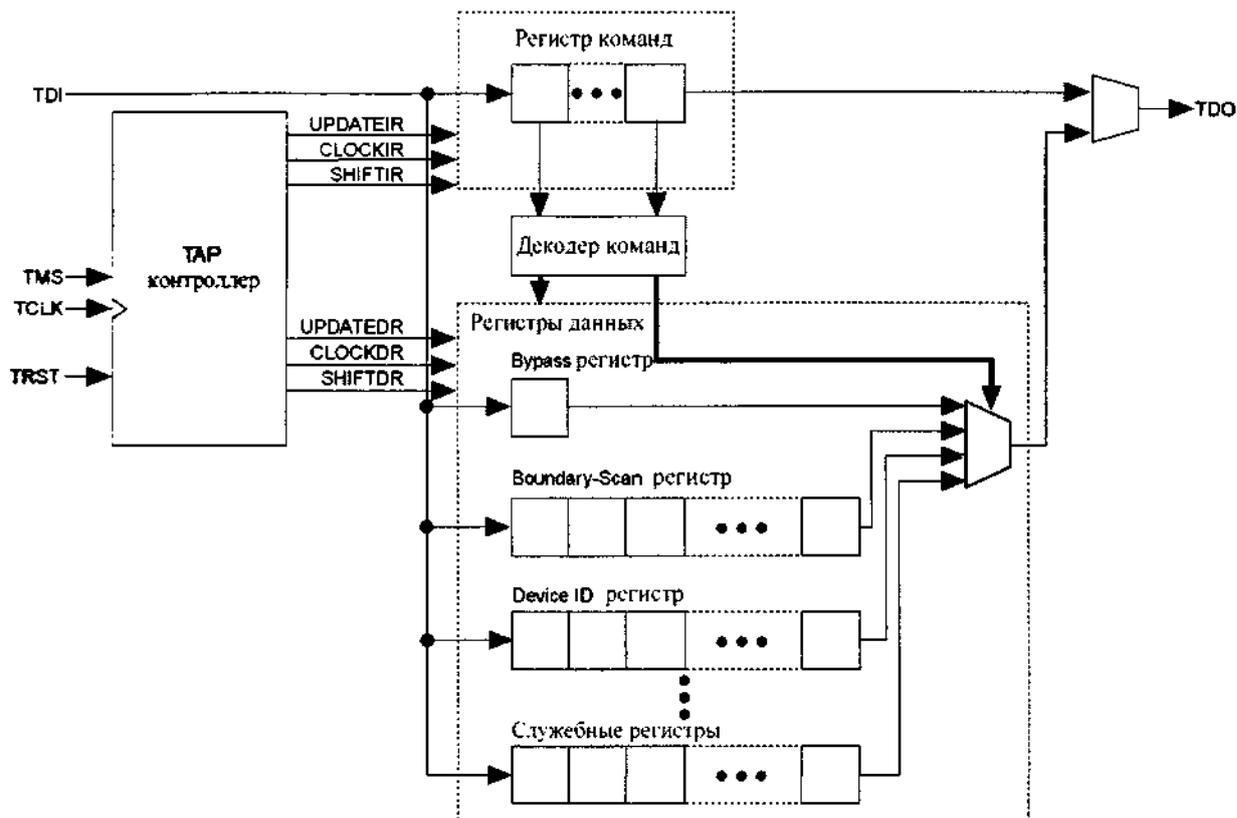


Рисунок 17- Функциональная схема периферийного сканирования

Длина регистра команд для ПЛИС 5576XC2Т составляет три бита, длина Boundary-Scan регистра - 231 бит. ПЛИС 5576XC2Т поддерживает команды СПС, показанные в таблице 7.

Таблица 7

Команды СПС	Описание
SAMPLE/PRELOAD	Позволяет снимать сигналы на выводах устройства, захватывать и проверять их во время нормального режима работы устройства.
EXTEST	Позволяет проверить соединения внешней схемы и на уровне плат.
BYPASS	Размещает 1 бит Вурасс регистра между TDI и TDO выводами, что обеспечивает синхронную передачу данных через выбранное устройство к соседнему устройству во время нормального режима работы устройства.

5.5.2 Архитектура Boundary-Scan регистра

Boundary—Scan регистр представляет последовательный сдвиговый регистр, который использует сигнал TDI как входной и сигнал TDO как выходной. Boundary-Scan регистр состоит из трехбитовых периферийных элементов от каждого элемента ввода-вывода, выделенного (специализированного) входа и служебного (конфигурационного) вывода. Boundary-Scan регистр используется для тестирования соединений внешних выводов или для захвата внутренних данных.

На рисунке 18 показано, как тестовые данные последовательно сдвигаются по периферии ПЛИС.

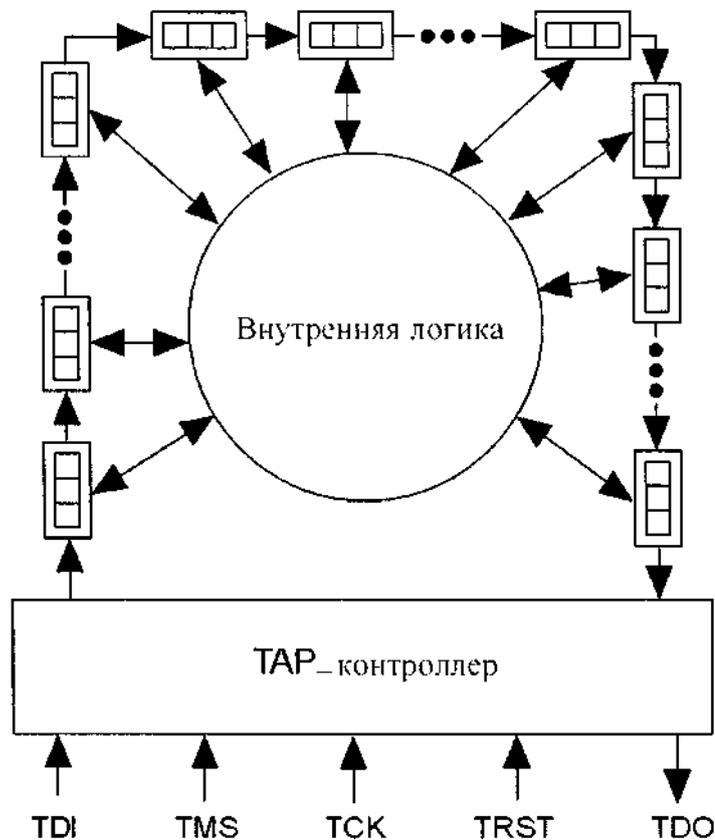


Рисунок 18 - Архитектура Boundary-Scan регистра

На рисунке 19 приведена временная диаграмма для работы с ПЛИС 5576XC2T по JTAG - схеме.

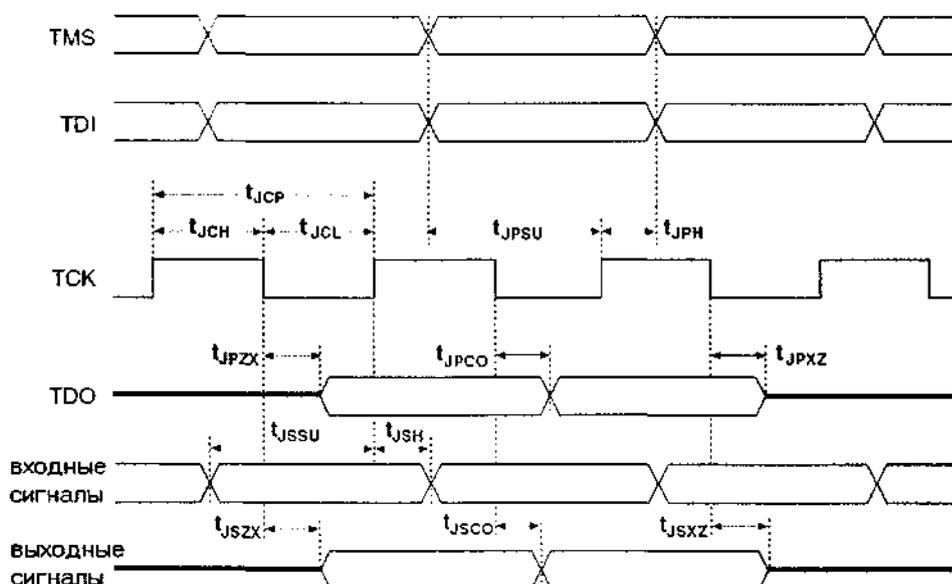


Рисунок 19 - Временная диаграмма для работы с ПЛИС 5576XC2T по JTAG - схеме

В таблице 8 приведены временные параметры и их значения для работы с ПЛИС 5576XC2T по JTAG - схеме. Таблица 8

Символ	Параметр	Значение	
		не менее, не	не более, не
t_{JCP}	Период тактового сигнала ТСК	100	-
t_{JCH}	Длительность импульса высокого уровня ТСК	50	-
t_{JCL}	Длительность импульса низкого уровня ТСК	50	-
t_{JPSU}	Время между переключением TMS и TDI и передним фронтом ТСК	20	-
t_{JPH}	Время между передним фронтом ТСК и переключением TMS и TDI	45	-

Окончание таблицы 8

Символ	Параметр	Значение	
		не менее, нс	не более, нс
t_{JPCO}	Время между задним фронтом ТСК и переключением TDO	-	25
t_{JPZX}	Время между задним фронтом ТСК и переключением TDO в активное состояние	-	25
t_{JPXZ}	Время между задним фронтом ТСК и переключением TDO в третье состояние	-	25
t_{JSSU}	Время между переключением входного захватываемого сигнала данных и передним фронтом ТСК	20	-
t_{JSH}	Время между передним фронтом ТСК и переключением входного захватываемого сигнала данных	45	-
t_{JSCO}	Время между задним фронтом ТСК и переключением выходного сигнала	-	35
t_{JSZX}	Время между задним фронтом ТСК и переключением выходного сигнала из третьего состояния в активное	-	35
t_{JSXZ}	Время между задним фронтом ТСК и переключением выходного сигнала в третье состояние	-	35

5.6 Тестирование

Каждая ПЛИС 5576XC2T проходит цикл функционального тестирования. Полное тестирование каждого бита конфигурационной памяти (СОЗУ) и всей функциональной логики гарантирует 100 % годность ПЛИС.

6 Расчет мощности потребления ПЛИС

Полная мощность потребления ПЛИС рассчитывается по формуле

$$P = P_{INT} + P_{IO}$$

где P - полная мощность потребления ПЛИС;

P_{INT} - мощность потребления ядра ПЛИС;

P_{IO} - мощность потребления периферии ПЛИС, которая зависит от нагрузок, подключенных к внешним выводам ПЛИС. P_{INT} можно рассчитать по формуле

$$P_{INT} = I_{CC} \times U_{CC} + C_{PD} + f_{cmax} \times U_{CC}^2,$$

где I_{CC} ~ ток потребления ПЛИС в статическом режиме без учета внешней нагрузки, А (таблица 2);

U_{CC} ~ напряжение питания ПЛИС, В (таблица 3);

C_{PD} - динамическая емкость ПЛИС, Ф;

f_{cmax} - максимальная частота следования импульсов тактового сигнала, Гц.

Динамическая емкость ПЛИС C_{PD} определяется выражением

$$C_{PD} = (0,192 + 4,823 \times \log_{LC}) * 10^{-10} \text{ Ф},$$

где \log_{LC} - средняя доля переключаемых логических ячеек в каждом такте.

P_{IO} можно рассчитать по формуле

$$P_{IO} = P_{DCIO} + P_{ACIO}$$

где P_{DCIO} - мощность, расходуемая на постоянную нагрузку периферии ПЛИС, например, подключенные резисторы между периферийными выводами и питанием или общим выводом;

P_{ACIO} - мощность, расходуемая на динамическую нагрузку периферии ПЛИС, т.е. на перезаряд нагрузочных емкостей, подключенных к внешним выводам.

$$P_{DCIO} = \sum P_{DCION}$$

где P_{DCIO} - мощность, расходуемая на постоянную нагрузку одного периферийного выхода ПЛИС. Ориентировочные значения P_{DCIO} Для различных типов нагрузки приведены в таблице 9.

Таблица 9

Тип нагрузки	P_{DCIO} , мВт
Резистор 1кОм, подключенный к выводу питания:	
- в нормальном режиме	0,21
- в режиме повышенной нагрузочной способности	0,12
Резистор 1кОм, подключенный к общему выводу:	
- в нормальном режиме	0,30
- в режиме повышенной нагрузочной способности	0,17
Вход КМОП микросхемы	0

P_{ACIO} зависит от емкости нагрузки, частоты переключения выходов и вычисляется по формуле

$$P_{ACIO} = 0,5 \times N_{IO} \times C_n \times f_{\text{сmax}} \times \text{tog}_{IO} \times U_{\text{cc}}^2,$$

где N_{IO} - количество используемых периферийных выходов;

C_n - средняя емкость внешней нагрузки на периферийный выход, Ф;

tog_{IO} - средняя доля переключаемых выходов в каждом такте.

Пример расчета:

$$f_{\text{сmax}} = 50 \text{ МГц}; \text{tog}_{LC} = 0,125, U_{\text{cc}} = 3,3 \text{ В}.$$

Получим

$$P_{INT} = 10 \times 10^{-3} \times 3,3 + (0,192 + 4,823 \times 0,125) \times 10^{-10} \times 50 \times 10^6 \times 3,3^2 = \\ = 0,033 + 0,043 = 0,076 \text{ Вт}$$

$N_{IO} = 64$; $\text{tog}_{IO} = 0,125$; нагрузки: $C_n = 35$ пФ и 50 pull-down резисторов номиналом 1 кОм, 25 из которых подключены к выводам, находящимся в нормальном режиме, 25 — к выводам, находящимся в режиме повышенной нагрузочной способности.

$$P_{DCIO} = (0,30 \times 10^{-3} \times 25) + (0,17 \times 10^{-3} \times 25) = 0,012 \text{ Вт}.$$

$$P_{ACIO} = 0,5 \times 64 \times 35 \times 10^{-12} \times 50 \times 10^6 \times 0,125 \times 3,3^2 = 0,076 \text{ Вт}.$$

$$P_{IO} = 0,012 + 0,076 = 0,088 \text{ Вт}.$$

Полная мощность потребления ПЛИС

$$P = 0,076 + 0,088 = 0,164 \text{ Вт.}$$

Эти вычисления предоставляют приблизительную оценку, основанную на типичных значениях.

МИКРОСХЕМЫ ИНТЕГРАЛЬНЫЕ 5576XC2T

Инструкция по программированию

ГПКФ.431262.002Д4

2007

Содержание

1	Введение.....	3
2	Схемы конфигурирования ПЛИС 5576XC2T.....	7
2.1	PS конфигурирование с активным конфигурационным ПЗУ.....	7
2.2	PS конфигурирование с загрузочным кабелем	12
2.3	PS конфигурирование с микропроцессором.....	17
2.4	PPS конфигурирование	21
2.5	AS конфигурирование с пассивным конфигурационным ПЗУ	24
2.6	JTAG конфигурирование.....	29
3	Программирование расширенной функциональности ПЛИС 5576XC2T ..,	35
3.1	Режимы устройства.....	36
3.2	Дополнительные режимы элементов ввода-вывода.....	38
4	Графический интерфейс	39
5	Консольная программа	42
5.1	Описание языка сценариев.....	43
5.2	Синтаксис языка сценариев	46
6	Программный интерфейс	48
6.1	Заголовочные файлы	48
6.2	Описание функций.....	48
6.2.1	Функция chip_open	48
6.2.2	Функция chip_close.....	49
6.2.3	Функция chip_set_device_option.....	49
6.2.4	Функция chip_get_device_option	51
6.2.5	Функция chip_set_pin_options.....	52
6.2.6	Функция chip_set_pin_options_by_name	53
6.2.7	Функция chip_get_pin_options	54
6.2.8	Функция chip_get_pin_options_by_name	55
6.2.9	Функция chip_save	56

1 Введение

Под программированием ПЛИС 5576XC2T следует понимать процесс конфигурирования, проходящий путем заполнения конфигурационной памяти данными. В настоящей инструкции приведены возможные схемы такого конфигурирования. Необходимые конфигурационные данные являются результатом проектирования в САПР MAX+PLUS II ф. Altera для ПК на базе Windows, рабочих станций Sun SPARCstation, HP 9000 Series 700/800.

ПЛИС 5576XC2T функционально совместима с изделиями EPF8282A, EPF8282AV ф. Altera, но не является их полным аналогом. По основным характеристикам разработанная микросхема соответствует ПЛИС EPF8282ALC84-2 ф. Altera. При создании проектов в MAX+PLUS II необходимо учитывать соответствие выводов изделий EPF8282A в корпусе PLCC84 и ПЛИС 5576XC2T в корпусе 4226.108-2 (таблица 1).

Таблица 1

Обозначение вывода ПЛИС 5576XC2T	Номер вывода в корпусе 4226.108-2	Обозначение вывода ПЛИС EPF8282A	Номер вывода в корпусе PLCC84	Номер ячейки для тестирования ПЛИС 5576XC2T (BST cell)
1	2	3	4	5
DCLK	1	DCLK	10	57
CONFIG DONE	2	CONFIG DONE	И	58
nSTATUS	3	nSTATUS	32	59
I/O0/DATA2	4	I/O8/DATA2	9	60
I/O1/DATA3	5	I/O7/DATA3	8	61
I/O2/DATA4	7	I/O6/DATA4	7	62
I/O3/DATA5	8	I/O5/DATA5	6	63
I/O4/DATA6	9	I/O4/DATA6	4	64
I/O5/DATA7	10	I/O3/DATA7	3	65
I/O6	12	I/O2	2	66
I/O7	13	I/O1	1	67
I/O8	15	I/O64	84	68
I/O9	16	I/O63	83	69
I/O10	18	I/O62	82	70
I/O11	19	I/O61	81	71
I/O12	20	I/O60/SDOUT	79	72
I/O13	21	I/O59/ADDO	78	73
I/O14	23	I/O58/RDYnBUSY	77	74
I/O15	24	I/O57/ADD1	76	75
ERR	25			76

Продолжение таблицы 1

1	2	3	4	5
TMS	26	I/O15/TMS	20	
TCK	27	I/O56/TCK	72	
TDI	29	I/O41/TDI	55	
I/O 16	30	I/O56/TCK	72	0
I/O 17	31	I/O55/ADD2	71	1
I/O 18	32	I/O54/ADD3	70	2
I/O 19	34	I/O53/ADD4	69	3
I/O20	35	I/O52/ADD5	67	4
INPUT3	37	INPUT4	73	5
I/O21	38	I/O51/ADD6	66	6
I/O22	39	I/O50/ADD7	65	7
I/O23	40	I/O49/ADD8	64	8
nCE*	42			9
I/O24	43	I/O48/ADD9	63	10
I/O25	44	I/O47/ADD10	62	11
I/O26	45	I/O46/ADD11	61	12
I/O27	47	I/O45/ADD12	60	13
I/O28	48	I/O44/ADD13	58	14
I/O29	50	I/O43/ADD14	57	15
I/O30	51	I/O42/ADD15	56	16
I/O31	52	I/O41/(TDI)	55	17
INPUT2	53	INPUT3	54	18
MSEL1	55	MSEL1	53	19
MSELO	56	MSELO	74	20
I/O32	57	I/O40/ADD17	51	21
I/O33/CLKUSR	58	I/O39/CLKUSR	50	22
I/O34	59	I/O38/RDCLK	49	23
I/O35	62	I/O37/nRS	48	24
I/O36	63	I/O36	46	25
I/O37	64	I/O35	45	26
I/O38	66	I/O34	44	27
I/O39	67	I/O33	43	28
I/O40	69	I/O32	42	29
I/O41	70	I/O31	41	30
I/O42	72	I/O30	40	31
I/O43	73	I/O29	39	32
I/O44	74	I/O28	37	33
I/O45	75	I/O27/ADD16	36	34
I/O46	77	I/O26	35	35
I/O47	78	I/O25	34	36

Окончание таблицы 1

1	2	3	4	5
TEST MODE**	79			
TRST	80	TRST	52	
nCONFIG	81	nCONFIG	33	37
INPUT 1	83	INPUT2	31	38
Ю48	84	I/O24/nWS	30	39
I/O49	85	I/O23/nCS	29	40
I/O50	86	I/O22/CS	28	41
I/O51	88	I/O21/(TDO)	27	42
I/O52	89	I/O20	25	43
I/O53	91	I/O 19	24	44
I/O54	92	I/O 18	23	45
I/O55/DEV OE	93	I/O 17	22	46
nCEO *	94			47
I/O56/DEV CLRn	96	I/O 16	21	48
I/O57	97	I/O15/TMS	20	49
I/O58	98	I/O 14	19	50
INPUTO	99	INPUT 1	12	51
I/O59	101	I/O 13	18	52
I/O60	102	I/O 12	16	53
I/O61	104	I/O11	15	54
I/O62/DATA0	105	I/O10/DATA0	14	55
I/O63/DATA1	106	I/O9/DATA1	13	56
TDO	107	I/O21/TDO	27	

Примечания

1 Выводы питания для корпуса 4226.108-2: 6, 14, 22, 33, 41, 49, 60, 68, 76, 87,95,103.

2 Выводы общие для корпуса 4226.108-2: 1 1, 17, 28, 36, 46, 54, 61, 65, 71, 82,90,100, 108.

3 Выводы питания для корпуса PLCC84: 17, 38, 59, 80.

4 Выводы общие для корпуса PLCC84: 5, 26, 47, 68.

* Данные выводы у ПЛИС ERF8282A отсутствуют.

** Вывод TEST MODE предназначен для тестирования ПЛИС 5576XC2T. При использовании в пользовательском режиме данный вывод должен быть подключен к GND.

ПЛИС 5576XC2T имеет расширенную функциональность элементов ввода-вывода, несколько дополнительных режимов работы, в том числе, режимы верификации и циклической перезаписи конфигурационной памяти. В настоящей

инструкции приведено также описание специального программного обеспечения, предназначенного для реализации этих дополнительных возможностей.

Часть выводов ПЛИС является выводами двойного назначения, которые могут использоваться, как в процессе конфигурирования, так и в пользовательском | режиме. Используемые в настоящей инструкции обозначения для этих выводов **DATA** [7...0], **CLKUSR**, **DEV_CLRn**, **DEV_OE** являются общепринятыми для процесса конфигурирования. Соответствие обозначений указанных выводов обозначениям выводов на схеме электрической структурной приведено в таблице 2.

Таблица 2 - Соответствие обозначений выводов

Обозначение вывода в настоящей инструкции	Обозначение вывода на схеме электрической структурной	Номер вывода в корпусе 4226. 108-2
DATA0	I/O62	105
DATA1	I/O63	106
DATA2	I/O0	4
DATA3	I/O1	5
DATA4	I/O2	7
DATA5	I/O3	8
DATA6	I/O4	9
DATA7	I/O5	10
CLKUSR	I/O33	58
DEV_CLRn	I/O56	96
DEV_OE	I/O55	93

Перед конфигурированием ПЛИС 5576XC2T необходимо обеспечить следующие условия:

- все выводы **GND** должны быть подключены к отрицательному выводу источника питания («общему»);

- все выводы **VCC** должны быть подключены к положительному выводу источника питания (3,3±0,3) В;

- все входы ПЛИС 5576XC2T должны быть подключены к отрицательному или положительному выводу источника питания напрямую или через резисторы pull-up, pull-down, или к выходам других микросхем. При неопределенности выбора предпочтение следует отдавать подключению к отрицательному выводу источника питания.

2 Схемы конфигурирования ПЛИС 5576XC2T

Для конфигурирования ПЛИС 5576XC2T существуют следующие конфигурационные схемы:

- PS (Passive Serial - пассивное последовательное) конфигурирование с активным конфигурационным ПЗУ;
- PS конфигурирование с загрузочным кабелем;
- PS конфигурирование с микропроцессором;
- PPS (Passive Parallel Synchronous - пассивное параллельное синхронное) конфигурирование;
- AS (Active Serial - активное последовательное) конфигурирование с пассивным конфигурационным ПЗУ;
- JTAG конфигурирование.

Для конфигурационных схем с использованием ПЗУ рекомендуется применять ПЗУ EPC2 ф. Altera или ПЗУ серии AT17LV ф. Atmel. При этом следует учитывать, что создание файла .prof для EPC2 средствами MAX+ PLUS II не предусмотрено для ПЛИС семейства FLEX 8000. Однако для данного типа ПЗУ файл .prof можно получить при помощи специального обеспечения, разработанного для поддержки расширенной функциональности ПЛИС 5576XC2T и описанного в данной инструкции.

2.1 PS конфигурирование с активным конфигурационным ПЗУ Схема для конфигурирования ПЛИС 5576XC2T с активным последовательным ПЗУ приведена на рисунке 1.

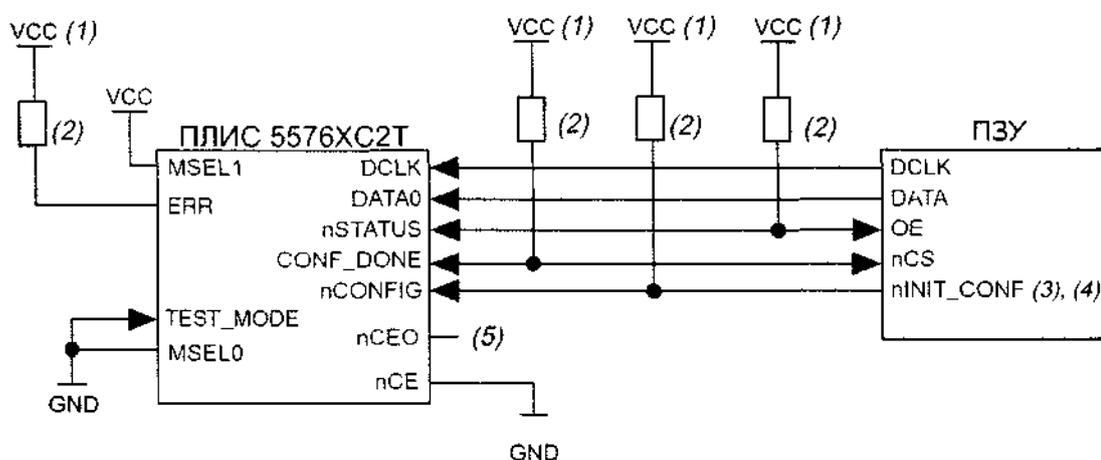


Рисунок 1, лист 1 - Конфигурирование ПЛИС 5576XC2T с активным последовательным ПЗУ

Примечания

1 Pull-up резисторы следует подключать к тому же напряжению питания, что и ПЗУ.

2 Все pull-up резисторы имеют номинал 1 кОм. В устройстве EPC2 выводы OE и nCS имеют внутренние, конфигурируемые пользователем pull-up резисторы. При использовании внутренних pull-up резисторов, нет необходимости применять внешние резисторы на этих выводах.

3 Наличие вывода nINIT_CONF зависит от конкретного типа используемого ПЗУ (имеется, например, в ПЗУ EPC2). Если используются ПЗУ, у которых данный вывод отсутствует (например, EPC1), то вывод nCONFIG должен быть подключен к VCC напрямую или через pull-up резистор.

4 Вывод nINIT_CONF имеет встроенный pull-up резистор, который всегда активен в устройстве EPC2. Внешний pull-up резистор в этом случае на выводе nINIT_CONF не обязателен.

5 Вывод nCEO отсоединен.

При включении питания вывод nCONFIG переключается из состояния низкого уровня в состояние высокого уровня и инициализируется конфигурирование. ПЛИС 5576XC2T переводит вывод CONF_DONE в состояние низкого уровня, который затем приходит на вывод nCS ПЗУ. В ПЛИС при этом проходит процесс сброса всех триггеров (Power On Reset или POR), а вывод nSTATUS переходит в пассивное состояние.

После включения питания перед началом конфигурирования ПЛИС в ПЗУ также проходит POR. В это время стабилизируется напряжение питания и ПЗУ формирует сигнал низкого уровня на выводе OE, который задерживает конфигурирование, так как данный вывод соединен с выводом nSTATUS ПЛИС 5576XC2T. После завершения режима POR на обоих устройствах, выводы nSTATUS и OE переходят в пассивное состояние, а сигнал на этих выводах подтягивается pull-up резистором до высокого уровня. Когда конфигурируется устройство из нескольких микросхем, конфигурирование не начинается, пока все устройства не переведут свои выводы OE и nSTATUS в пассивное состояние. После этого ПЗУ последовательно передает данные на вывод DATA0 ПЛИС 5576XC2T.

После успешного конфигурирования ПЗУ продолжает подавать тактовый сигнал на вход DCLK ПЛИС 5576XC2T с целью инициализации. ПЛИС 5576XC2T переводит вывод CONF_DONE в пассивное состояние и сигнал

дотягивается до высокого уровня pull-up резистором. После того, как инициализация завершена, ПЛИС переходит в пользовательский режим.

Если в процессе конфигурирования возникает ошибка, то ПЛИС 5576XC2T формирует на выводе nSTATUS сигнал низкого уровня, который приводит к сбросу ПЛИС и ПЗУ. При включенной опции Auto-restart configuration on frame error, доступной в MAX+PLUS II Global Project Device Options (Assign меню), устройство при обнаружении ошибки реконфигурируется автоматически.

Если данная опция отключена, то должна быть внешняя система, которая будет проверять вывод nSTATUS на наличие ошибок и в случае обнаружения ошибки посылать сигнал низкого уровня на вывод nCONFIG для перезапуска конфигурирования. Внешняя система может посылать сигнал на вывод nCONFIG, если он находится под контролем системы, а не подключен к VCC. Когда конфигурирование завершено, вывод CONF_DONE переводится в пассивное состояние, при этом из-за перевода вывода nCS в состояние высокого уровня pull-up резистором блокируется ПЗУ. Вывод DCLK до и после конфигурирования находится в состоянии низкого уровня.

Если после передачи всех данных из ПЗУ вывод CONF_DONE остаётся в состоянии низкого уровня, это означает, что конфигурирование прошло неудачно. В этом случае, ПЗУ на несколько микросекунд переводит вывод OE в состояние низкого уровня, при этом вывод nSTATUS ПЛИС 5576XC2T также переходит в состояние низкого уровня. Если опция Auto-restart configuration on frame error включена, то после обнаружения ошибки вывод nSTATUS возвращается в состояние высокого уровня, после чего ПЗУ реконфигурирует ПЛИС.

На рисунке 2 представлена схема конфигурирования устройства из нескольких ПЛИС 5576XC2T с ПЗУ.

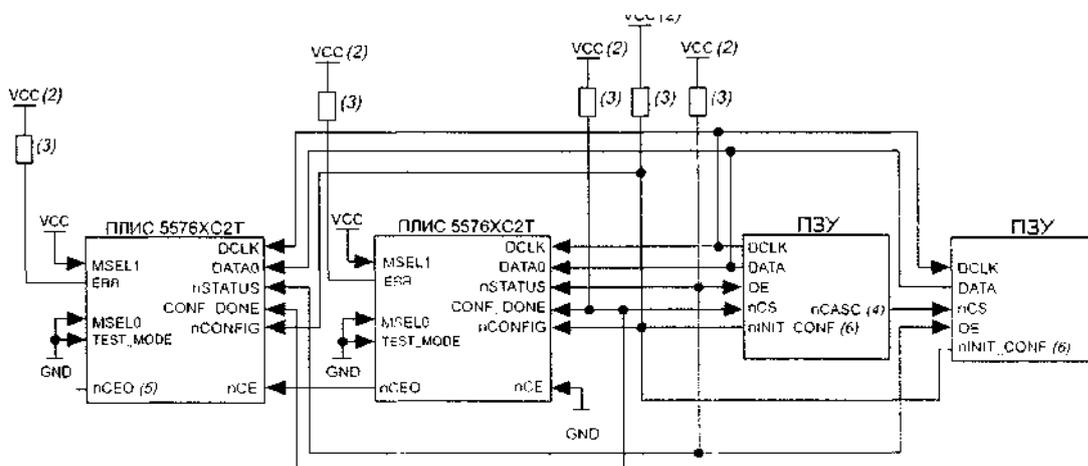


Рисунок 2, лист 1 - PS конфигурирование нескольких ПЛИС 5576XC2T с ПЗУ

1 Когда в последовательной конфигурации используется несколько ПЛИС, для ПЗУ необходимо создать файл Programmer Object File (.pof) из файлов SRAM Object File (.sof) каждого проекта. Для этого можно комбинировать несколько .sof файлов, используя Combine Programming Files (File меню) программного обеспечения MAX+PLUS II.

2 Pull-up резисторы следует подключить к тому же напряжению питания, что и ПЗУ.

3 Все pull-up резисторы имеют номинал 1 кОм. В устройстве EPC2 выводы OE и nCS имеют внутренние, конфигурируемые пользователем pull-up резисторы. При использовании внутренних pull-up резисторов, нет необходимости применять внешние резисторы на этих выводах.

4 ПЗУ, у которых отсутствует вывод nCASC, нельзя каскадировать.

5 Вывод nCEO последней ПЛИС цепи отсоединен.

6 Наличие вывода nINIT_CONF зависит от конкретного типа используемого ПЗУ (имеется, например, в ПЗУ EPC2). Если используются ПЗУ, у которых данный вывод отсутствует (например, EPC1), то вывод nCONFIG должен быть подключен к VCC напрямую или через pull-up резистор.

После завершения конфигурирования первого устройства, его вывод nCEO активизирует вывод nCE во втором устройстве, заставляя второе устройство начать конфигурирование. Так как все выводы CONF_DONE объединены, то все ПЛИС инициализируются и переходят в пользовательский режим одновременно.

Выводы nSTATUS всех ПЛИС также объединены, следовательно, если одно из устройств (в том числе и ПЗУ) обнаруживает ошибку, то конфигурирование прекращается. Если первое ПЗУ не обнаруживает на выводе CONF_DONE высокого уровня в конце конфигурирования, то через несколько микросекунд происходит сброс конфигурации путем подачи импульса низкого уровня на вывод OE, который переводит вывод OE на других ПЗУ в состояние низкого уровня и устанавливает вывод nSTATUS на всех ПЛИС в состояние ошибки.

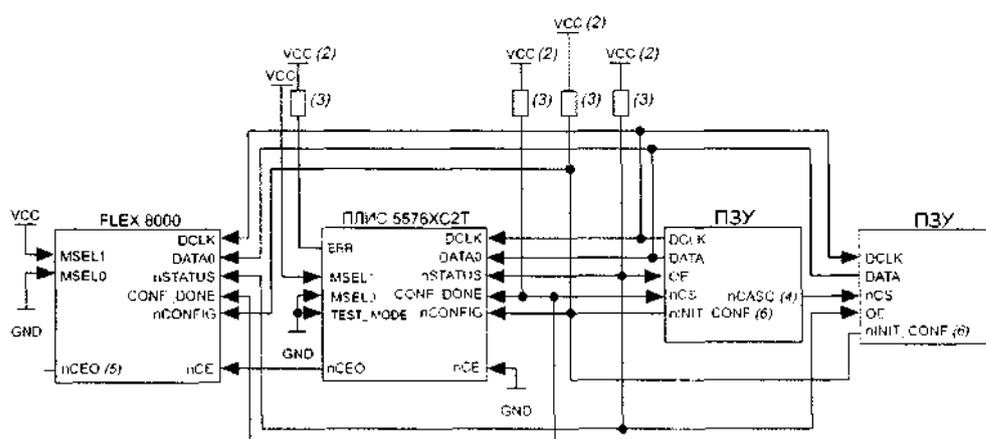
Если включена опция Auto-restart configuration on frame error, то после сброса ПЛИС устанавливают выводы nSTATUS в пассивное состояние. Когда вывод nSTATUS пассивен и сигнал дотягивается до высокого уровня, происходит процесс реконфигурирования. Если эта опция отключена, вывод nSTATUS ПЛИС 5576XC2T будет находиться в состоянии низкого уровня, пока на вывод nCONFIG не будет подан сигнал низкого уровня.

Можно каскадировать несколько ПЗУ для конфигурирования сложных устройств. Когда все данные из первого ПЗУ переданы, на выводе nCASC появ-

Можно каскадировать несколько ПЗУ для конфигурирования сложных устройств. Когда все данные из первого ПЗУ переданы, на выводе nCASC появляется сигнал низкого уровня, который управляет входом nCS на следующем ПЗУ. Поскольку для активации ПЗУ требуется меньше, чем один цикл переключений, то поток данных получится непрерывный.

Можно использовать одну конфигурационную цепь для конфигурирования составных устройств. В этой схеме вывод nCEO первого устройства соединяется с выводом nCE следующего устройства в цепочке. Выводы CONF_DONE и nSTATUS всех устройств должны быть объединены вместе.

На рисунке 3 показан пример конфигурирования составного устройства с использованием ПЗУ.



Примечания

1 Если в последовательной конфигурации используется несколько ПЛИС, то для ПЗУ необходимо создать файл Programmer Object File (.pof) из файлов SRAM Object File (.sof) каждого проекта. Для этого можно комбинировать несколько .sof файлов, используя Combine Programming Files (File меню) программного обеспечения MAX+PLUS II.

2 Pull-up резисторы следует подключить к тому же напряжению питания, что и ПЗУ.

3 Все pull-up резисторы имеют номинал 1 кОм. В устройстве EPC2 выходы OE и nCS имеют внутренние, конфигурируемые пользователем pull-up резисторы. При использовании внутренних pull-up резисторов, нет необходимости применять внешние резисторы на этих выводах.

4 ПЗУ, у которых отсутствует вывод nCASC, нельзя каскадировать.

5 Вывод nCEO последней ПЛИС в цепи отсоединен.

Рисунок 3, лист 1 - Конфигурирование составного устройства с активным ПЗУ

6 Наличие вывода nINIT_CONF зависит от конкретного типа используемого ПЗУ (имеется, например, в ПЗУ EPC2). Если используются ПЗУ, у которых данный вывод отсутствует (например, EPC1), то вывод nCONFIG должен быть подключен к VCC напрямую или через pull-up резистор.

Таблица 3 описывает назначение выводов ПЛИС 5576XC2T во время и после конфигурирования.

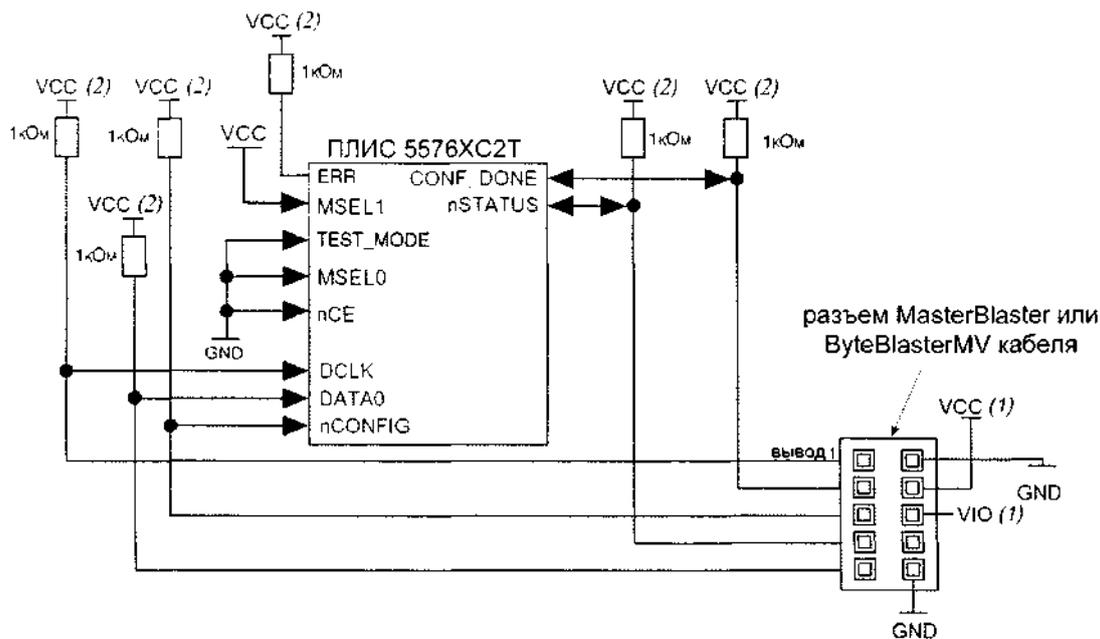
Таблица 3 - Назначение выводов ПЛИС 5576XC2T во время конфигурирования и после конфигурирования

Вывод	Назначение	
	в процессе конфигурирования	после конфигурирования
DATA0	Вход данных для конфигурирования	Определяется пользователем
DATA[7...1] *	Вход данных в некоторых конфигурационных режимах	
I/O выводы	Пользовательский вход-выход (третье состояние)	
* Функциональное назначение выводов зависит от установленных в программном обеспечении MAX+PLUS II (диалоговое окно Global Project Device Options) параметров.		

2.2 PS конфигурирование с загрузочным кабелем

Для начала конфигурирования по данной схеме программное обеспечение MAX+PLUS II (интеллектуальный хост) через кабель переводит сигнал на выводе nCONFIG из состояния низкого уровня в состояние высокого уровня. Затем конфигурационные данные по одному биту передаются через кабель MasterBlaster или ByteBlasterMV на вывод DATA0 ПЛИС 5576XC2T. Данные в ПЛИС будут передаваться до тех пор, пока вывод CONF_DONE не перейдет в состояние высокого уровня.

При данной схеме конфигурирования опция Auto-restart on frame error не влияет на конфигурационный цикл, поскольку программное обеспечение MAX+PLUS II самостоятельно перезапустит конфигурирование при обнаружении ошибки. Рисунок 4 показывает PS конфигурирование устройства с использованием кабеля MasterBlaster или ByteBlasterMV.



Примечания

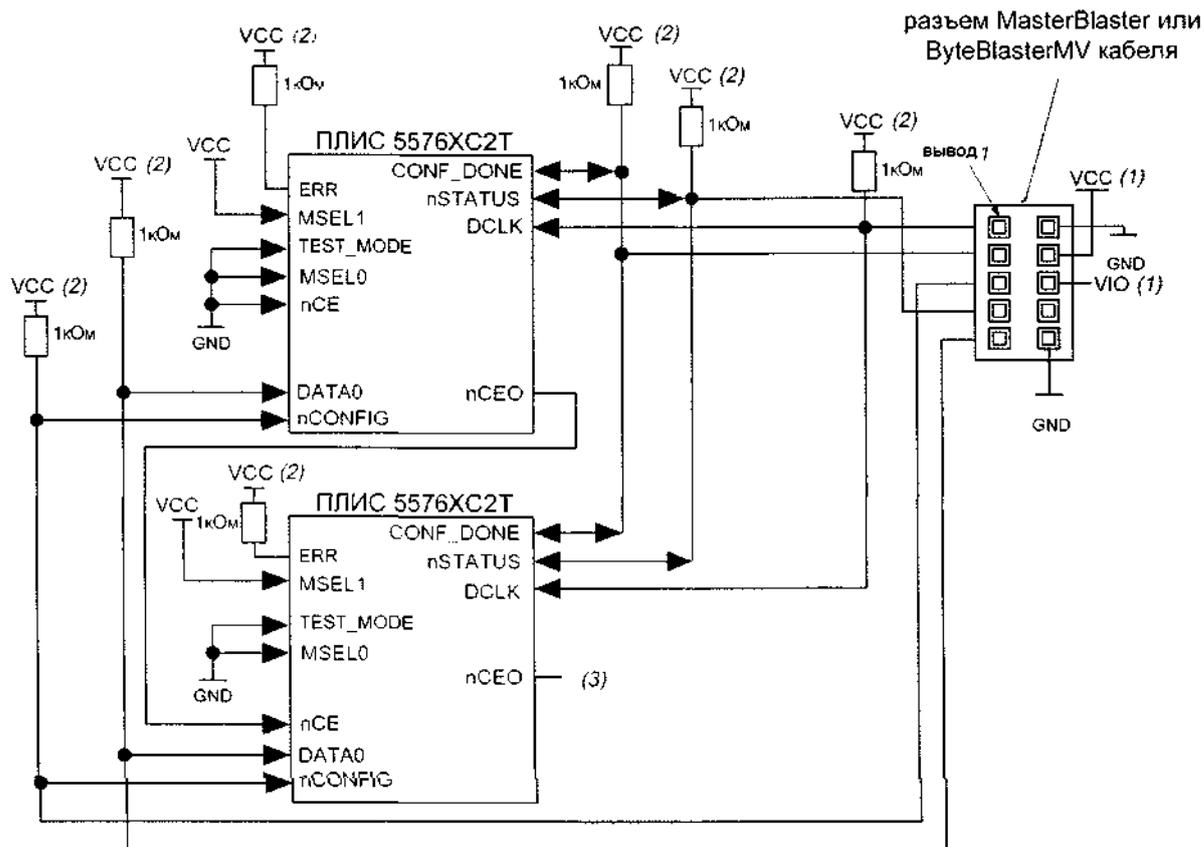
1 Напряжение питания для кабеля MasterBlaster или ByteBlasterMV и на выводе VIO должно соответствовать напряжению питания ПЛИС.

2 Pull-up резисторы следует подключать к такому же напряжению питания, что и вывод VIO кабеля MasterBlaster или ByteBlasterMV.

Рисунок 4 - PS конфигурирование с использованием кабеля MasterBlaster или ByteBlasterMV

Для конфигурирования сложных устройств необходимо подсоединить вывод nCEO первой ПЛИС к выводу nCE последующей. Все выходы CONF_DONE объединены, следовательно, все ПЛИС переходят в пользовательский режим одновременно. Поскольку выходы nSTATUS также объединены, то при обнаружении ошибки каким-либо из устройств, конфигурирование всей цепочки останавливается. При этом программное обеспечение MAX+PLUS II самостоятельно перезапускает процесс конфигурирования (опция Auto-restart configuration on frame error не влияет на конфигурационный цикл).

На рисунке 5 показано как конфигурировать устройства с несколькими ПЛИС 5576XC2T с помощью кабеля MasterBlaster или ByteBlasterMV.



Примечания

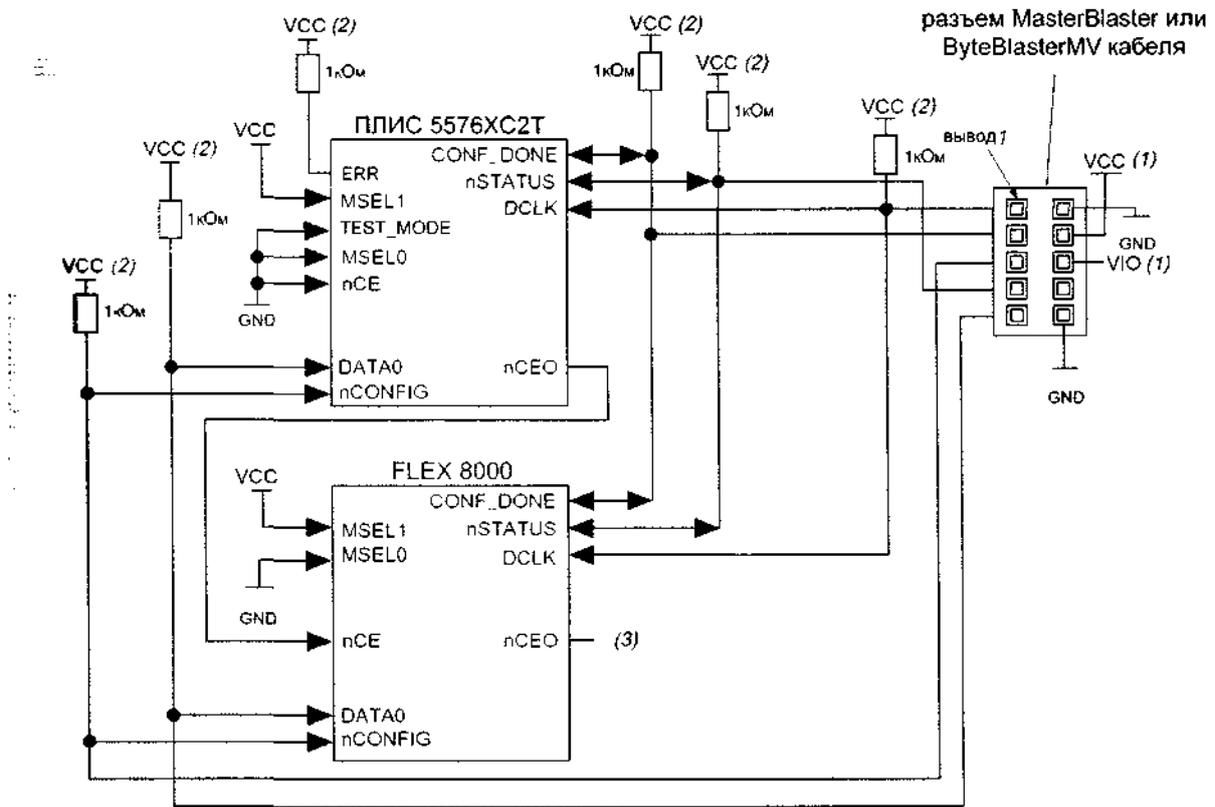
1 Напряжение питания для кабеля MasterBlaster или ByteBlasterMV и на выводе VIO должно соответствовать напряжению питания ПЛИС.

2 Pull-up резисторы следует подключать к такому же напряжению питания, что и вывод VIO кабеля MasterBlaster или ByteBlasterMV.

3 Вывод nCEO последней ПЛИС цепи отсоединен.

Рисунок 5 - Конфигурирование устройства с несколькими ПЛИС 5576XC2T с использованием кабеля MasterBlaster или ByteBlasterMV

На рисунке 6 показано, как конфигурировать сложные устройства с загрузочным кабелем MasterBlaster или ByteBlasterMV.



Примечания

1 Напряжение питания для кабеля MasterBlaster или ByteBlasterMV и на выводе VIO должно соответствовать напряжению питания ПЛИС.

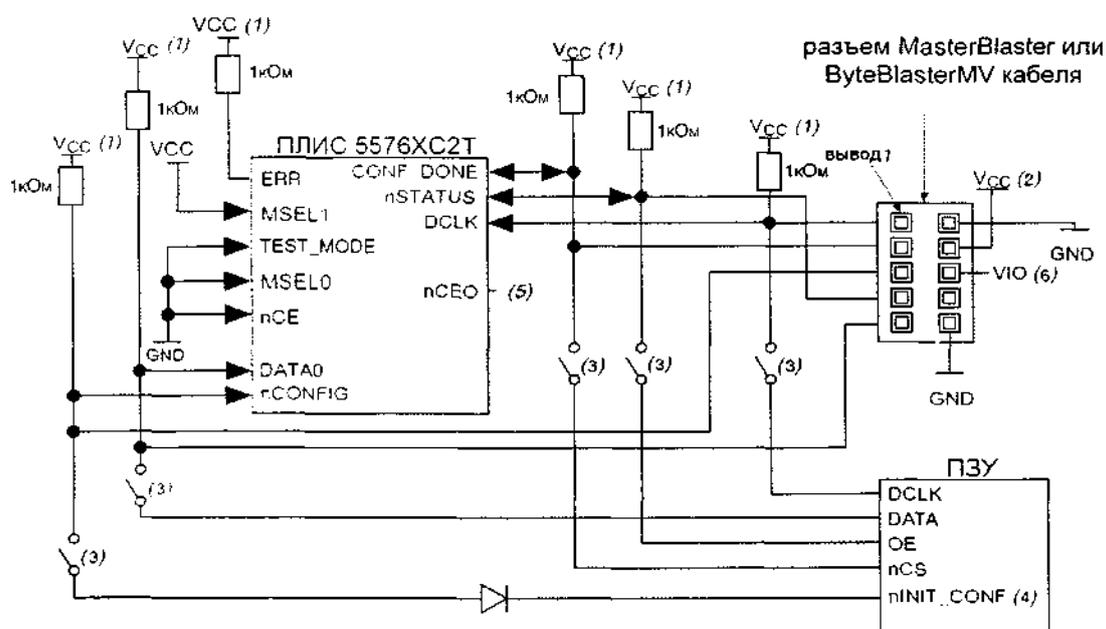
2 Pull-up резисторы следует подключать к такому же напряжению питания, что и вывод VIO кабеля MasterBlaster или ByteBlasterMV.

3 Вывод nCEO последней ПЛИС цепи отсоединен.

Рисунок 6 - Конфигурирование сложных устройств с кабелем MasterBlaster или ByteBlasterMV

Если использовать кабель MasterBlaster или ByteBlasterMV для конфигурирования ПЛИС на одной плате с конфигурационным ПЗУ, то необходимо изолировать конфигурационное ПЗУ. Для этого можно использовать дополнительную логику, например, мультиплексор, который выбирает между конфигурационным устройством и кабелем. Мультиплексор должен иметь двунаправленный перенос на сигналы nSTATUS и CONF_DONE. Также можно использовать дополнительные ключи к пяти общим сигналам (CONF_DONE, nSTATUS, DCLK, nCONFIG и DATA0) между кабелем и конфигурационным устройством.

На рисунке 7 показана комбинация конфигурационного устройства и кабеля MasterBlaster или ByteBlasterMV для конфигурирования ПЛИС.



Примечания

1 Pull-up резисторы следует подключать к такому же напряжению питания, что и ПЗУ.

2 Напряжение питания для кабеля MasterBlaster или ByteBlasterMV должно соответствовать напряжению питания ПЛИС.

3 Запрещается проводить конфигурирование с помощью кабеля MasterBlaster или ByteBlasterMV при подключенном к ПЛИС ПЗУ. Необходимо или удалить устройство конфигурирования с платы, или разместить переключатели на пяти общих сигналах между загрузочным кабелем и конфигурационным устройством.

4 Наличие вывода nINIT_CONF зависит от конкретного типа используемого ПЗУ (имеется, например, в ПЗУ EPC2). Если используются ПЗУ, у которых данный вывод отсутствует (например, EPC1), то вывод nCONFIG должен быть подключен к VCC напрямую или через pull-up резистор.

5 Вывод nCEO отсоединен.

6 Напряжение на выводе VIO должно соответствовать напряжению питания ПЛИС.

Рисунок 7 - Комбинация конфигурационного устройства и кабеля Master-Blaster или ByteBlasterMV

2.3 PS конфигурирование с микропроцессором

При PS конфигурировании ПЛИС с микропроцессором, последний передает данные от запоминающего устройства к ПЛИС 5576XC2T. Для начала конфигурирования по данной схеме микропроцессор должен перевести сигнал на выводе nCOXFIG из состояния низкого уровня в состояние высокого уровня, а ПЛИС 5576XC2T - перевести вывод nSTATUS в пассивное состояние. Микропроцессор или программирующая аппаратура затем передает конфигурационные данные по одному биту за такт на вывод DATAO ПЛИС 5576XC2T. Младший бит (least signification bit (LSB)) в каждом байте данных должен быть представлен первым.

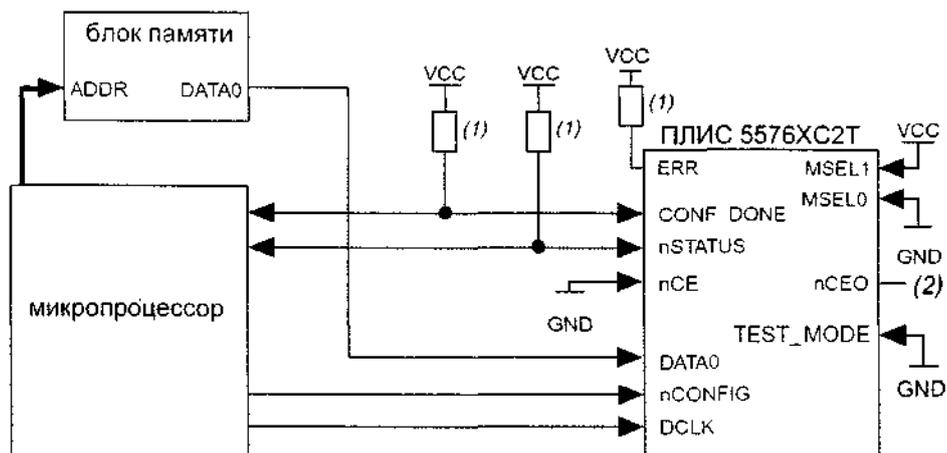
После того, как все данные переданы, вывод DCLK должен переключиться десять раз для инициализации ПЛИС 5576XC2T, после чего вывод CONF_DONE переходит в состояние высокого уровня, показывая тем самым, что конфигурирование прошло успешно. Конфигурационные файлы создаются с помощью программного обеспечения MAX+PLUS II и включают дополнительные биты для инициализации. Переключения на выводе DCLK после конфигурирования не влияют на работу ПЛИС.

Ограничений на максимальный период следования импульсов тактового сигнала для вывода DCLK нет, поэтому можно приостановить конфигурирование путем прекращения подачи импульсов на неограниченное время.

Если ПЛИС 5576XC2T обнаруживает ошибку в процессе конфигурирования, на выводе nSTATUS появляется сигнал низкого уровня. В этом случае микропроцессор может подать импульс низкого уровня на вывод nCONFIG для перезапуска процесса конфигурирования. Если опция Auto-restart configuration on frame error включена в программном обеспечении MAX+PLUS II, то ПЛИС 5576XC2T переводит вывод nSTATUS в пассивное состояние после сброса конфигурации, а затем микропроцессор может переконфигурировать ПЛИС 5576XC2T без требуемого импульса низкого уровня на выводе nCONFIG.

Микропроцессор может также отслеживать состояние вывода CONF_DONE. Если микропроцессор отправил все данные, но вывод CONF_DONE не перешел в состояние высокого уровня, то должна произойти реконфигурация ПЛИС.

На рисунке 8 показана схема PS конфигурирования с микропроцессором.



Примечания

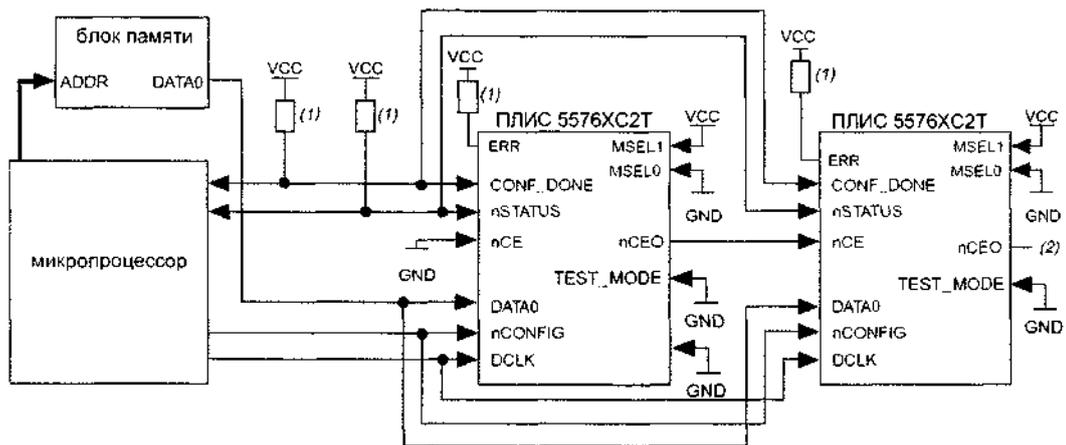
- 1 Все pull-up резисторы имеют номинал 1 кОм.
- 2 Вывод nCEO отсоединен.

Рисунок 8 - PS конфигурирование с микропроцессором

При PS конфигурировании составных устройств с помощью микропроцессора вывод nCEO первой ПЛИС должен быть соединен с выводом nCE следующей ПЛИС. Последующее устройство в цепочке начинает конфигурироваться после предыдущего в пределах одного тактового цикла, поэтому процесс передачи данных будет для микропроцессора непрерывным. Так как выходы CONF_DONE всех ПЛИС объединены между собой, то все устройства инициализируются и переходят в пользовательский режим одновременно.

Поскольку выходы nSTATUS также объединены, то при обнаружении ошибки хотя бы одним устройством, процесс конфигурирования всей цепочки останавливается. После этого микропроцессор может перезапустить конфигурирование, послав сигнал низкого уровня на вывод nCONFIG. Если включена опция Auto-restart configuration on frame error в программном обеспечении MAX+PLUS II, то ПЛИС 5576XC2T переводит вывод nSTATUS в пассивное состояние после сброса конфигурации, после чего микропроцессор может пере-конфигурировать ПЛИС.

На рисунке 9 показана схема конфигурирования нескольких ПЛИС 5576XC2T при помощи микропроцессора.

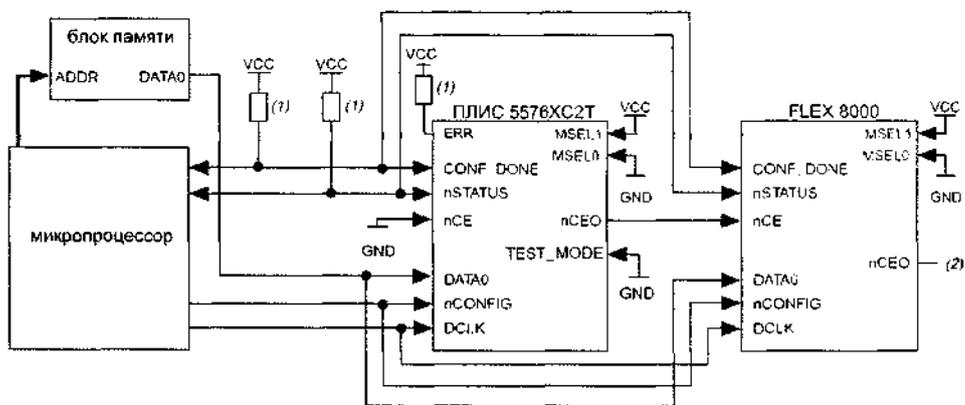


Примечания

- 1 Все pull-up резисторы имеют номинал 1 кОм.
- 2 Вывод nCEO последнего устройства отсоединен.

Рисунок 9 - Конфигурирование нескольких ПЛИС 5576XC2T при помощи микропроцессора

На рисунке 10 показан пример схемы PS конфигурирования сложносоставных устройств при помощи микропроцессора.

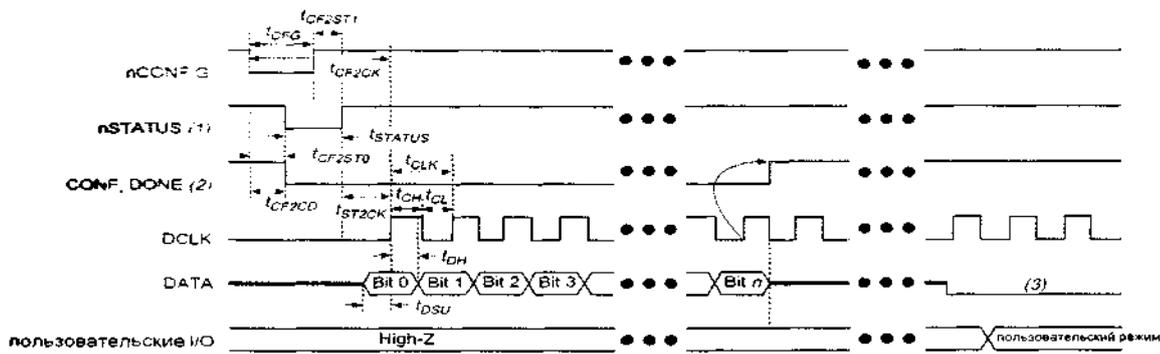


Примечания

- 1 Все pull-up резисторы имеют номинал 1 кОм.
- 2 Вывод nCEO последнего устройства отсоединен.

Рисунок 10 - PS конфигурирование сложносоставных устройств при помощи микропроцессора

На рисунке 11 представлены временные диаграммы при PS конфигурировании ПЛИС 5576XC2T.



Примечания

1 После включения питания ПЛИС 5576XC2T устанавливает вывод nSTATUS в состояние низкого уровня не менее чем на 1 мкс после того, как напряжение питания достигает требуемого минимального значения.

2 После включения питания и перед конфигурированием вывод CONF_DONE находится в состоянии низкого уровня.

3 После окончания процесса конфигурирования на входе данных DATA0 должно быть состояние высокого или низкого уровня.

Рисунок 11 - Временные диаграммы при PS конфигурировании ПЛИС 5576XC2T

В таблице 4 содержится информация о временных характеристиках при PS конфигурировании ПЛИС 5576XC2T.

Таблица 4 - Временные характеристики при PS конфигурировании ПЛИС 5576XC2T

Обозначение	Параметр	Значение		Единица измерения
		не менее	не более	
1	2	3	4	5
t_{CF2D}	Время между переключениями nCONFIG и CONF_DONE в состояние низкого уровня	-	200	нс
t_{CF2ST0}	Время между переключениями nCONFIG и nSTATUS в состояние низкого уровня	-	200	нс

Окончание таблицы 4

1	2	3	4	5
t_{CF2ST1}	Время между переключениями nCONFIG и nSTATUS в состояние высокого уровня	-	4	мкс
t_{CFG}	Длительность импульса низкого уровня nCONFIG	2	-	мкс
t_{SIATUS}	Длительность импульса низкого уровня nSTATUS	1	-	мкс
t_{CF2CK}	Время между переключением nCONFIG в состояние низкого уровня и первым передним фронтом DCLK	5	-	мкс
t_{ST2CK}	Время между переключением nSTATUS в состояние высокого уровня и первым передним фронтом DCLK	1	-	мкс
t_{DSU}	Время предустановки данных перед передним фронтом DCLK	10	-	нс
t_{DN}	Время удержания данных после переднего фронта DCLK	0	-	нс
t_{CH}	Длительность импульса высокого уровня DCLK	50	-	нс
t_{CL}	Длительность импульса низкого уровня DCLK	50	-	нс
t_{CLK}	Период DCLK	100	-	нс
f_{MAX}	Максимальная частота DCLK	-	10	МГц

2.4 PPS конфигурирование

В PPS конфигурационной схеме конфигурированием ПЛИС управляет интеллектуальный хост, который выдает тактовый сигнал и данные побайтно на ПЛИС 5576XC2T. Выводы DCLK, CONF_DONE, nCONFIG, nSTATUS и DATA [7...0] соединяются через порт с интеллектуальным хостом так же, как с микропроцессором. Для того чтобы начать конфигурирование, вывод nCONFIG должен переключиться из состояния низкого уровня в состояние высокого уровня, после чего хост расположит восьмибитное слово на выводах DATA [7...0] ПЛИС.

По первому переднему фронту тактового сигнала DCLK байт конфигурационных данных защелкивается в ПЛИС. Следующие восемь задних фронтов тактового сигнала упорядочивают (последовательно загружают) данные в ПЛИС, по девятому переднему фронту защелкивается

следующий байт конфигурационных данных и т. д. Можно приостановить конфигурирование путем прекращения подачи импульсов на вывод DCLK в любой момент времени. Если происходит ошибка в процессе конфигурирования, на выводе nSTATUS появляется сигнал низкого уровня. В этом случае хост начинает реконфигурацию или выдает сообщение об ошибке.

После последнего байта данных вывод DCLK должен переключиться еще десять раз, чтобы перевести вывод CONF_DONE в пассивное состояние и инициализировать устройство. После перехода вывода CONF_DONE в состояние высокого уровня конфигурирование завершено. На рисунке 12 показан пример схемы PPS конфигурирования ПЛИС.

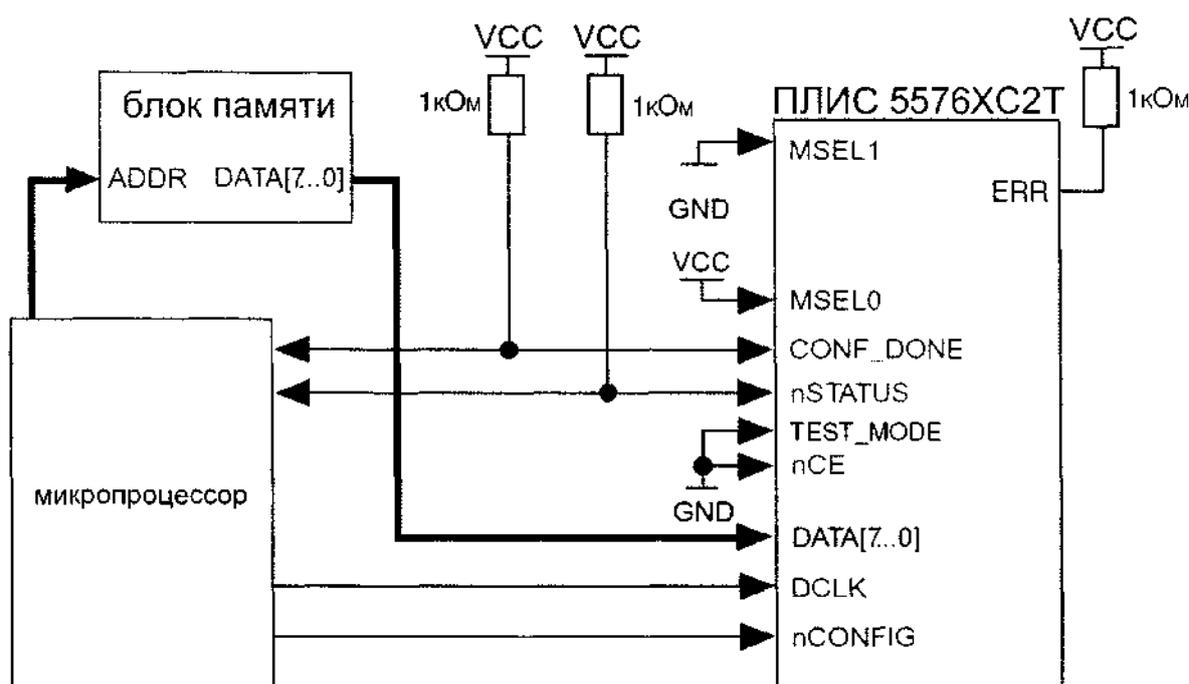


Рисунок 12 - PPS конфигурирование ПЛИС 5576XC2T

Рисунок 13 показывает пример схемы PPS конфигурирования составных устройств. После успешного конфигурирования первого устройства цепочки на его выводе nCEO устанавливается сигнал низкого уровня, тем самым позволяя начать конфигурирование следующего устройства цепочки. Второе устройство начинает конфигурироваться в пределах одного тактового периода. Так как выходы CONF_DONE всех устройств объединены, то все устройства инициализируются и переходят в пользовательский режим одновременно. Поскольку

выводы nSTATUS также объединены, то при обнаружении ошибки, происходит автоматическая реконфигурация.

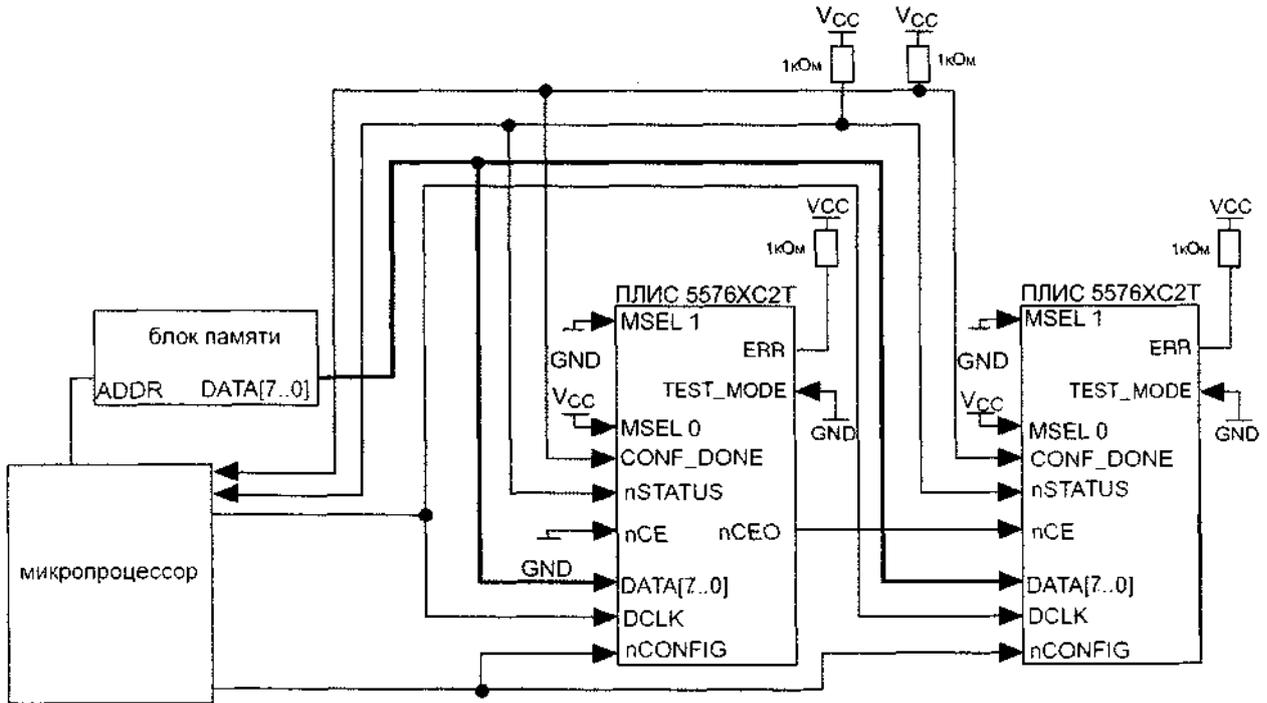


Рисунок 13 - PPS конфигурирование составных устройств

На рисунке 14 представлены временные диаграммы при PPS конфигурировании.

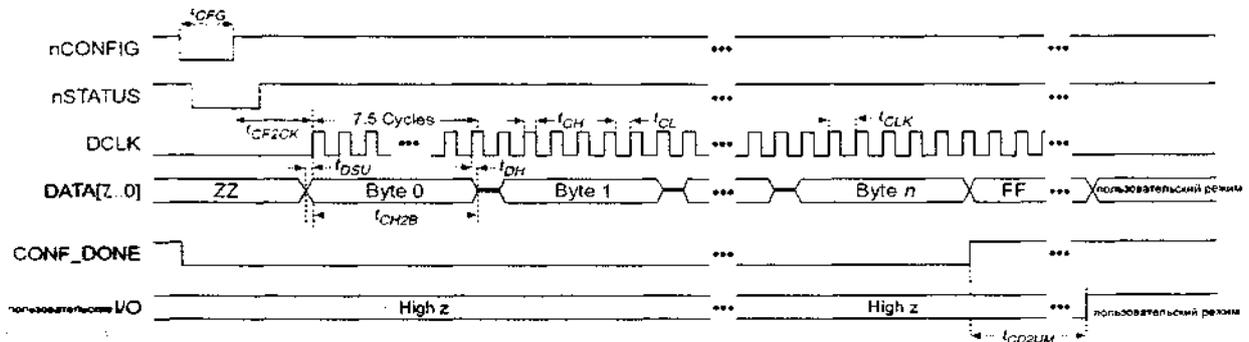


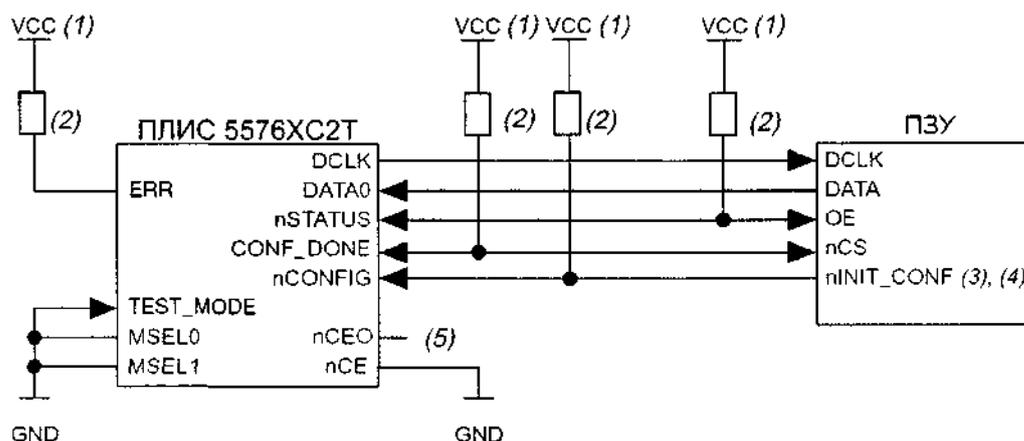
Рисунок 14 - Временные диаграммы при PPS конфигурировании

В таблице 5 приведены временные параметры при PPS конфигурировании ПЛИС 5576XC2T.

Таблица 5 - Временные параметры при PPS конфигурировании ПЛИС 5576XC2T

Обозначение	Параметр	Значение		Единица измерения
		не менее	не более	
t_{CF2CK}	Время между переключением nCONFIG в состояние высокого уровня и первым передним фронтом DCLK	5	-	мкс
t_{DSU}	Время предустановки данных перед передним фронтом DCLK	10	-	нс
t_{DN}	Время удержания данных после переднего фронта DCLK	0	-	нс
t_{CFG}	Длительность импульса низкого уровня nCONFIG	2	-	мкс
t_{CH}	Длительность импульса высокого уровня DCLK	50	-	нс
t_{CL}	Длительность импульса низкого уровня DCLK	50	-	нс
t_{CLK}	Период DCLK	100		нс
f_{max}	Максимальная частота DCLK		10	МГц
t_{CD2UM}	Время между переключением ICONF DONE в состояние высокого уровня и выходом в пользовательский режим	0,6	2	мкс

2.5 AS конфигурирование с пассивным конфигурационным ПЗУ
 Конфигурирование с пассивным конфигурационным ПЗУ отличается от PS конфигурирования тем, что тактовый сигнал формируется ПЛИС 5576XC2T, т. е. вывод DCLK ПЛИС 5576XC2T при AS конфигурировании является выходом генерируемого тактового сигнала. При конфигурировании ПЛИС, работающей в активном режиме, рекомендуется использовать пассивные последовательные ПЗУ (ПЗУ, вывод DCLK у которых является входом тактового сигнала), например пассивные ПЗУ EPC1213 ф. Altera или AT17C65A, AT17LV65A, AT17C128A, AT17LV128A, AT17C256A, AT17LV256A ф. Atmel. Схема для конфигурирования ПЛИС 5576XC2T с пассивным последовательным ПЗУ приведена на рисунке 15.



Примечания

1 Pull-up резисторы следует подключать к тому же напряжению питания, что и ПЗУ.

2 Все pull-up резисторы имеют номинал 1 кОм.

3 Если вывод nINIT_CONF отсутствует, то вывод nCONFIG должен быть подключен к VCC напрямую или через pull-up резистор.

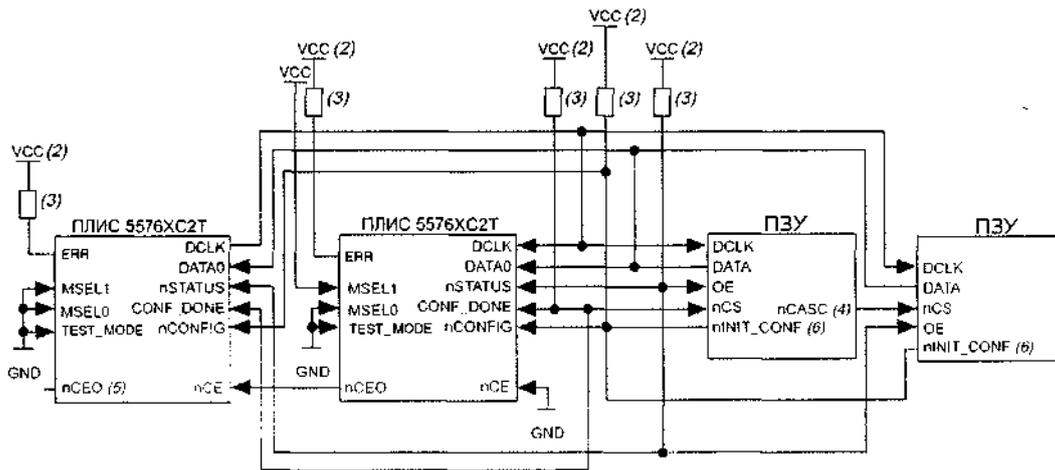
4 Если вывод nINIT_CONF имеет встроенный pull-up резистор, то внешний pull-up резистор на данном выводе не обязателен.

5 Вывод nCEO отсоединен.

Рисунок 15 - Конфигурирование ПЛИС 5576XC2T с пассивным последовательным ПЗУ

AS конфигурирование нескольких ПЛИС (с пассивным конфигурационным ПЗУ) отличается от PS конфигурирования нескольких ПЛИС (с активным конфигурационным ПЗУ) тем, что тактовый сигнал формируется одной из ПЛИС 5576XC2T.

На рисунке 16 представлена схема AS конфигурирования устройства из нескольких ПЛИС с ПЗУ.



Примечания

1 Когда в последовательной конфигурации используется несколько ПЛИС, для ПЗУ необходимо создать файл Programmer Object File (.pof) из файлов SRAM Object File (.sof) каждого проекта. Для этого можно комбинировать несколько .sof файлов, используя Combine Programming Files (File меню) про граммного обеспечения MAX+PLUS II.

2 Pull-up резисторы следует подключить к тому же напряжению питания, что и ПЗУ.

3 Все pull-up резисторы имеют номинал 1 кОм.

4 ПЗУ, у которых отсутствует вывод nCASC, нельзя каскадировать.

5 Вывод nCEO последней ПЛИС в цепи отсоединен.

6 Если вывод nINIT_CONF имеет встроенный pull-up резистор, то внешний pull-up резистор на данном выводе не обязателен.

Рисунок 16 - AS конфигурирование нескольких ПЛИС 5576XC2T с ПЗУ

После завершения конфигурирования первого устройства, его вывод nCEO активизирует вывод nCE следующего устройства цепочки, тем самым позволяя начать его конфигурирование. Так как все выходы CONF_DONE объединены, то все ПЛИС инициализируются и переходят в пользовательский режим одновременно.

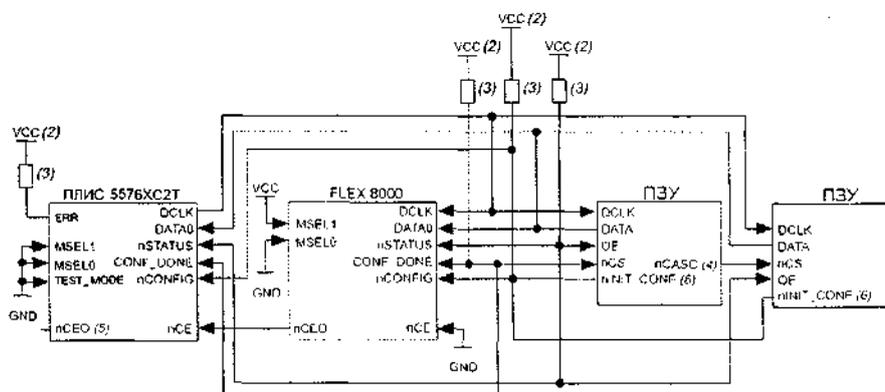
Выводы nSTATUS всех ПЛИС также объединены, следовательно, если какое-либо из устройств (в том числе и ПЗУ) обнаруживает ошибку, то конфигурирование останавливается. Если первое ПЗУ цепочки не обнаруживает на выводе CONF_DONE высокого уровня в конце конфигурирования, то через несколько микросекунд происходит сбрасывание конфигурации путем подачи

импульса низкого уровня на вывод OE, который переводит вывод OE на других ПЗУ в состояние низкого уровня и устанавливает вывод nSTATUS на всех ПЛИС в состояние ошибки.

Если включена опция Auto-restart configuration on frame error, то ПЛИС устанавливают выходы nSTATUS в пассивное состояние, после чего начинается процесс реконфигурирования. Если эта опция отключена, вывод nSTATUS ПЛИС 5576XC2T будет находиться в состоянии низкого уровня до тех пор, пока nCONFIG не будет сброшен сигналом низкого уровня.

Можно использовать одну конфигурационную цепь для конфигурирования составных устройств. В этой схеме вывод nCEO первого устройства присоединяется к выводу nCE следующего устройства в цепочке. Выводы CONF_DONE и nSTATUS всех устройств должны быть объединены.

На рисунке 17 показан пример конфигурирования составного устройства с использованием ПЗУ.



Примечания

1 Когда в последовательной конфигурации используется несколько ПЛИС, для ПЗУ необходимо создать файл Programmer Object File (.pof) из файлов SRAM Object File (.sof) каждого проекта. Для этого можно комбинировать несколько .sof файлов, используя Combine Programming Files (File меню) программного обеспечения MAX+PLUS II.

2 Pull-up резисторы следует подключить к тому же напряжению питания, что и ПЗУ.

3 Все pull-up резисторы имеют номинал 1 кОм.

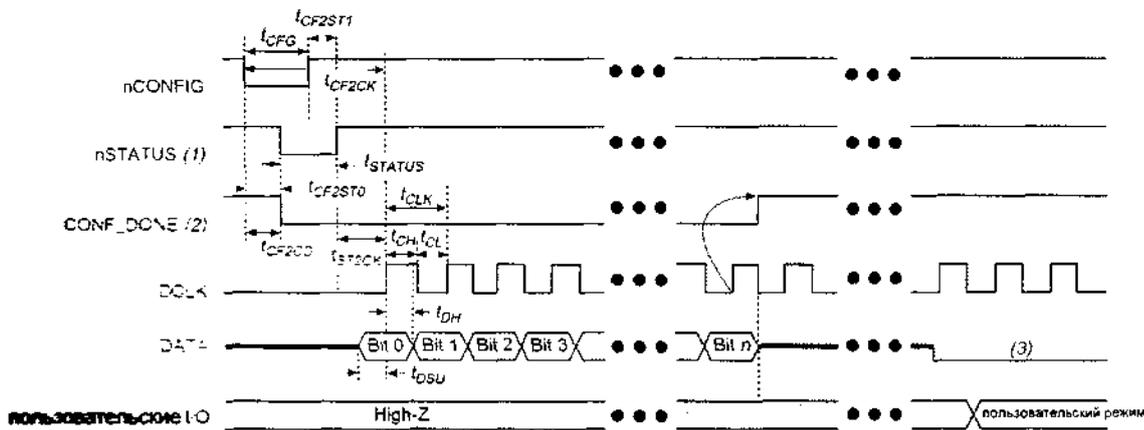
4 ПЗУ, у которых отсутствует вывод nCASC, нельзя каскадировать.

5 Вывод nCEO последней ПЛИС в цепи отсоединен.

6 Если вывод nINIT_CONF имеет встроенный pull-up резистор, то внешний pull-up резистор на данном выводе не обязателен.

Рисунок 17 - Конфигурирование составных устройств с пассивными ПЗУ

На рисунке 18 представлены временные диаграммы при AS конфигурировании ПЛИС 5576XC2T.



Примечания

1 После включения питания ПЛИС 5576XC2T устанавливает вывод nSTATUS в состояние низкого уровня не менее чем на 1 мкс после того, как напряжение питания достигает требуемого минимального значения.

2 После включения питания и перед конфигурированием вывод CONF_DONE находится в состоянии низкого уровня.

3 После окончания процесса конфигурирования на входе DAT АО должно быть состояние высокого или низкого уровня.

Рисунок 18 - Временные диаграммы при AS конфигурировании

Таблица 6 содержит информацию о временных характеристиках при AS конфигурировании ПЛИС 5576XC2T.

Таблица 6 - Временные характеристики при AS конфигурировании ПЛИС 5576XC2T

Обозначение	Параметр	Значение		Единица измерения
		не менее	не более	
1	2	3	4	5
t _{CF2CD}	Время между переключениями nCONFIG и CONF_DONE в состояние низкого уровня	-	200	нс
t _{CF2ST0}	Время между переключениями nCONFIG и nSTATUS в состояние низкого уровня	-	200	нс

Окончание таблицы 6

1	2	3	4	5
t_{CF2ST1}	Время между переключениями nCONFIG и nSTATUS в состояние высокого уровня	-	4	мкс
t_{CFG}	Длительность импульса низкого уровня nCONFIG	2	-	мкс
t_{STATUS}	Длительность импульса низкого уровня nSTATUS	1	-	мкс
t_{CF2CK}	Время между переключением nCONFIG в состояние низкого уровня и первым передним фронтом DCLK	5	-	мкс
t_{ST2CK}	Время между переключением nSTATUS в состояние высокого уровня и первым передним фронтом DCLK	1	-	мкс
t_{psu}	Время предустановки данных перед передним фронтом DCLK	10	-	нс
t_{DH}	Время удержания данных после переднего фронта DCLK	0	-	нс
t_{CH}	Длительность импульса высокого уровня DCLK	50	-	нс
t_{CL}	Длительность импульса низкого уровня DCLK	50	-	нс
t_{CLK}	Период DCLK	100	-	нс
f_{max}	Максимальная частота DCLK	-	10	МГц

2.6 JTAG конфигурирование

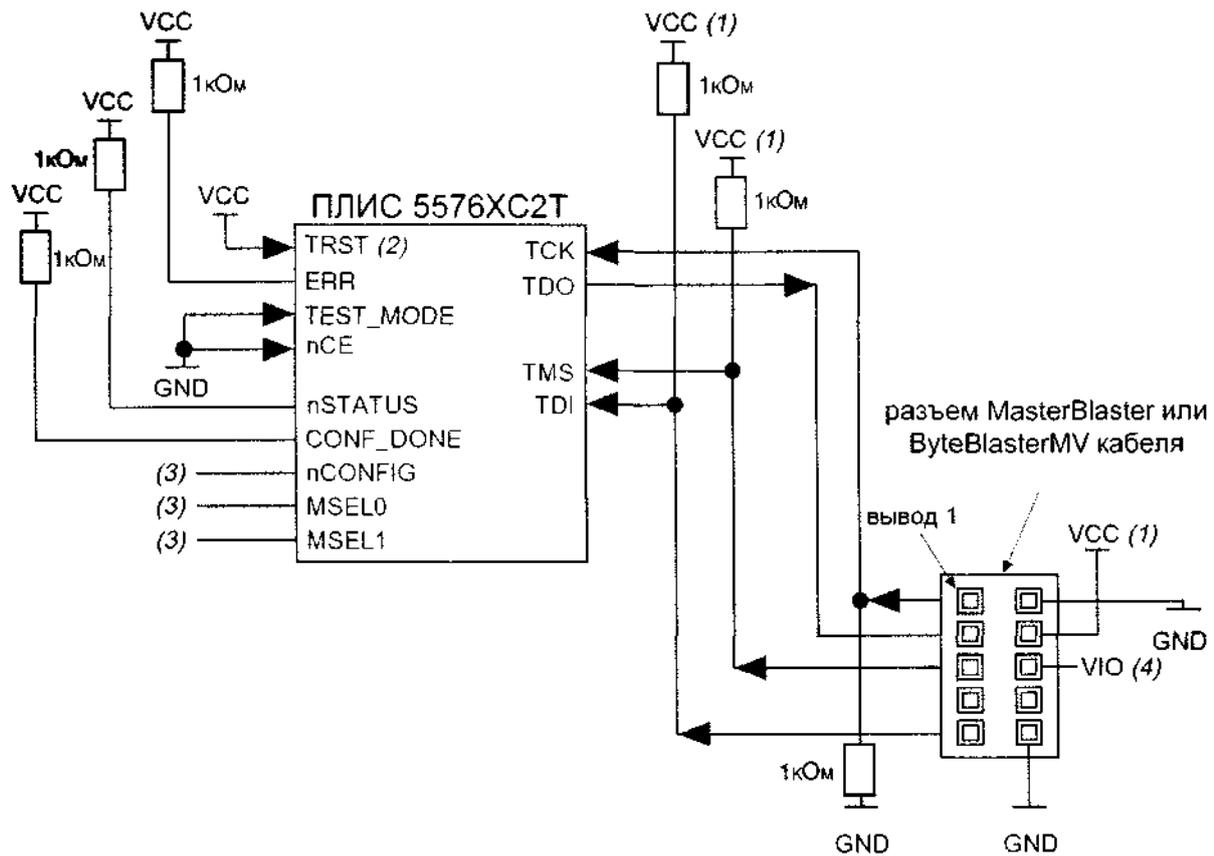
Система Joint Test Action Group (JTAG) разработана специально для тестирования микросхем. Порт JTAG также может быть использован для конфигурирования ПЛИС 5576XC2T.

Для работы с JTAG предназначены четыре вывода: TDI, TDO, TMS, TCK. В тестовом режиме может также использоваться дополнительный вывод TRST. В процессе JTAG конфигурирования остальные выводы находятся в третьем состоянии. Не следует начинать JTAG конфигурирование, пока все другие процессы конфигурирования не завершены. В таблице 7 указано назначение JTAG выводов.

Таблица 7 - Назначение JTAG выводов

Вывод	Описание	Назначение
TDI	Вход данных JTAG	Последовательный вход данных для тестирования и программирования. Данные захватываются по переднему фронту сигнала TCK.
TDO	Выход данных JTAG	Последовательный выход данных для тестирования и программирования. Данные выводятся по заднему фронту сигнала TCK. Когда выход не используется, он переключается в третье состояние.
TMS	Управление состоянием JTAG	Вход управляющего сигнала, который определяет переходы автомата состояний TAP - контроллера. Сигнал TMS должен быть установлен до срабатывания переднего фронта сигнала TCK.
TCK	Тактовый вход JTAG	Вход тактового сигнала. Некоторые операции совершаются по переднему фронту этого сигнала, некоторые - по заднему.
TRST	Асинхронный сброс JTAG	Вход асинхронного сброса в режиме тестирования с активным низким уровнем.

Во время JTAG конфигурирования данные загружаются в ПЛИС через кабель MasterBlaster или ByteBlasterMV. Конфигурирование устройства через кабель похоже на системное программирование устройства за исключением вывода TRST, который следует подключить к VCC. Это гарантирует, что TAP контролер не будет сброшен во время конфигурирования. Схема JTAG конфигурирования приведена на рисунке 19.



Примечания

1 Pull-up резисторы должны быть подключены к тому же напряжению питания, что и загрузочный кабель.

2 Вывод TRST следует подключить к VCC.

3 Если используется конфигурирование только по JTAG порту, вывод nCONFIG следует подключить к VCC, а выводы MSEL0 и MSEL1 - к GND. Если требуется поддерживать конфигурирование схемы не только по JTAG, выводы nCONFIG, MSEL0 и MSEL1 должны быть подключены согласно соответствующей схеме конфигурирования.

4 Напряжение на выводе VIO должно соответствовать напряжению питания ПЛИС.

Рисунок 1

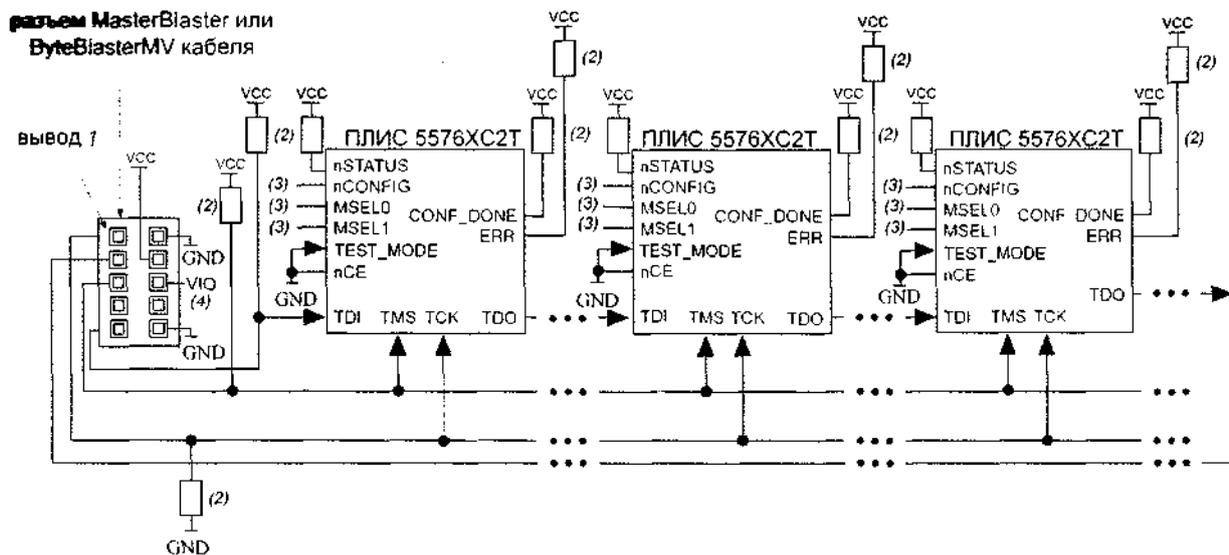
мировать или верифицировать ПЛИС 5576XC2T. Конфигурационные данные, отправленные в устройство, появляются на выводе TDO спустя один период тактового сигнала.

JTAG тестирование может быть выполнено на ПЛИС 5576XC2T как до, так и после конфигурирования, но не во время него. Выводы глобального сброса (DEV_CLRn) и разрешения третьего состояния (DEV_OE) в ПЛИС 5576XC2T не влияют на JTAG операции периферийного сканирования или программирования. Когда планируется конфигурирование ПЛИС 5576XC2T по JTAG порту, следует учитывать подключение стандартных конфигурационных выводов. В таблице 8 показано, как эти выводы должны быть подключены при JTAG конфигурировании.

Таблица 8 - Подключение конфигурационных выводов при JTAG конфигурировании

Обозначение вывода	Подключение вывода
nCE	Должен быть подключен к GND напрямую, через резистор или управляться некоторой управляющей схемой.
nSTATUS	Должен быть подключен к VCC через pull-up резистор 1кОм. Когда конфигурируются сложные устройства по JTAG схеме, каждый вывод nSTATUS должен быть подключен к VCC через отдельный pull-up резистор ¹⁾
CONF_DONE	Должен быть подключен к VCC через pull-up резистор 1кОм. Когда конфигурируются сложные устройства по JTAG схеме, каждый вывод CONF_DONE должен быть подключен к VCC через отдельный pull-up резистор ²⁾ .
nCONFIG	Должен быть подключен к VCC через pull-up резистор 1кОм или управляться некоторой управляющей схемой.
MSEL0, MSEL1	Эти выводы не должны оставаться неподключенными. Они используются во всех конфигурационных схемах, кроме JTAG. Если используется только JTAG конфигурирование, то оба вывода следует подключить к GND.
DCLK	Не должен оставаться неподключенным. Должен иметь низкий или высокий уровень.
DATA0	Не должен оставаться неподключенным. Должен иметь низкий или высокий уровень.
TRST	Этот JTAG вывод не подключен к загрузочному кабелю и должен иметь высокий уровень.
¹⁾ Низкий уровень вывода nSTATUS во время JTAG конфигурирования указывает на то, что произошла ошибка. ²⁾ Высокий уровень вывода CONF_DONE в конце JTAG конфигурирования указывает на то, что конфигурирование прошло успешно.	

Когда программируется цепочка JTAG устройств, к разъему кабеля MasterBlaster или ByteBlasterMV подключают несколько программируемых устройств. Число устройств в JTAG цепочке ограничивается только возможностями загрузочного кабеля. Однако когда в JTAG цепочке соединяется более пяти устройств, рекомендуется буферизировать сигналы TCK, TDI и TMS. На рисунке 20 показана схема такого конфигурирования.



Примечания

1 Несколько ПЛИС 5576XC2Т могут быть размещены в одной JTAG цепочке для программирования и конфигурирования.

2 Все pull-up, pull-down резисторы имеют номинал 1кОм.

3 Если используется конфигурирование только по JTAG порту, вывод nCONFIG следует подключить к VCC, а выводы MSEL0 и MSEL1 - к GND. Если требуется поддерживать конфигурирование схемы не только по JTAG, выводы nCONFIG, MSEL0 и MSEL1 должны быть подключены согласно соответствующей схеме конфигурирования.

4 Напряжение на выводе VIO должно соответствовать напряжению питания ПЛИС.

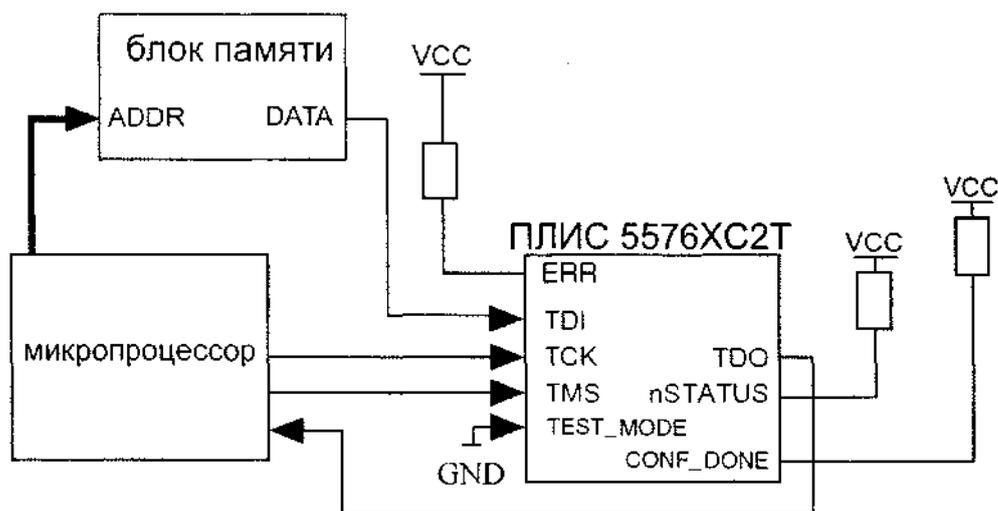
Рисунок 20 - JTAG конфигурирование нескольких ПЛИС 5576XC2Т

Успешное JTAG конфигурирование проверяется автоматически программным обеспечением MAX+PLUS II, которое проверяет состояние вывода CONF_DONE через JTAG порт в конце конфигурирования. Если вывод CONF_DONE находится в состоянии низкого уровня, то программное обеспечение MAX+PLUS II выдает сообщение, что конфигурирование не завершено.

Если вывод CONF_DONE находится в состоянии высокого уровня, программное обеспечение выдает сообщение, что конфигурирование завершено успешно.

Не допускается одновременно запускать конфигурирование по JTAG порту и по какой-либо иной схеме.

На рисунке 21 показано JTAG конфигурирование ПЛИС 5576XC2T с микропроцессором.



Примечание - Все pull-up резисторы имеют номинал 1 кОм.

Рисунок 21 - JTAG конфигурирование ПЛИС 5576XC2T с микропроцессором

На рисунке 22 показаны временные диаграммы при JTAG конфигурировании.

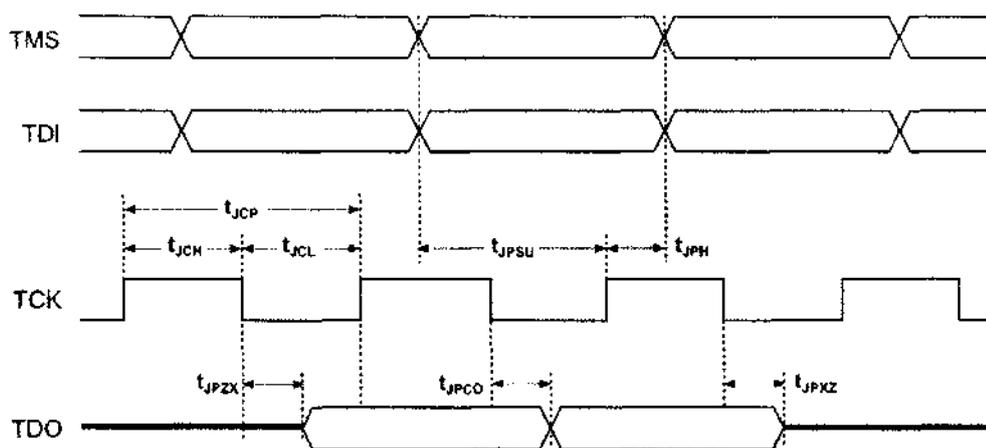


Рисунок 22 - Временные диаграммы при JTAG конфигурировании

Таблица 9 содержит информацию о временных характеристиках ПЛИС 5576XC2T при JTAG конфигурировании.

Таблица 9 - Временные параметры ПЛИС 5576XC2T при JTAG конфигурировании

Обозначение	Параметр	Значение		Единица измерения
		не менее	не более	
t_{JCP}	Период тактового сигнала ТСК	100	-	нс
t_{JCH}	Длительность импульса высокого уровня ТСК	50	-	нс
t_{JCL}	Длительность импульса низкого уровня ТСК	50	-	нс
t_{JPSU}	Время предустановки данных TMS и TDI перед передним фронтом ТСК	20	-	нс
t_{JPH}	Время удержания данных TMS и TDI после переднего фронта ТСК	45	-	нс
t_{JPCO}	Время между задним фронтом ТСК и переключением TDO	-	25	нс
t_{JPZX}	Время между задним фронтом ТСК и переключением TDO в активное состояние	-	25	нс
t_{JPXZ}	Время между задним фронтом ТСК и переключением TDO в третье состояние	-	25	нс

3 Программирование расширенной функциональности ПЛИС 5576XC2T

Для задания дополнительных и изменения установленных с помощью САПР MAX+PLUS II ф. Altera режимов работы ПЛИС 5576XC2T, а также для программирования расширенной функциональности пользовательских элементов ввода-вывода используется комплекс программ «5576XC2T Configuration Edit Tools».

Данный пакет предоставляет удобный графический интерфейс, а также инструменты для автоматизации процесса внесения изменений: консольную программу, управляемую сценариями, и программный интерфейс, позволяющий встраивать работу с данными проекта в пользовательское приложение. В качестве входных и выходных данных используются файлы прошивок в форматах .sof и .rbf, обычно используемых в САПР MAX+PLUS II. Экспорт данных также возможен в формат .prof для ПЗУ типа EPC2.

3.1 Режимы устройства

Режимы устройства:

- User-supplied start-up clock - позволяет использовать в качестве входа тактового сигнала для вывода микросхемы в рабочий режим пользовательский вывод CLKUSR (I/O33). Этот сигнал может быть использован для обеспечения синхронной инициализации нескольких устройств. Если данный режим не активен, то в пассивных режимах загрузки в качестве входа тактового сигнала используется DCLK, в активном режиме устройство использует внутренний генератор тактового сигнала для выхода в рабочий режим. При включении этого режима вывод CLKUSR должен быть зарезервирован как вход;

- Auto-restart configuration on frame error - данный режим влияет на поведение ПЛИС в случае возникновения ошибки в процессе конфигурирования или в режиме верификации конфигурационной памяти. В случае, если данный режим активен, при возникновении ошибки автоматически происходит перезагрузка конфигурационной памяти ПЛИС. В противном случае микросхема остается в состоянии ошибки;

- Release clears before tri-states - в процессе конфигурирования пользовательские выходы находятся в третьем состоянии. В случае, если данный режим неактивен, ПЛИС при выходе в рабочий режим отпускает сигнал, удерживающий пользовательские выходы в третьем состоянии прежде, чем будет переведен в неактивное состояние сигнал сброса на триггерах (как расположенных в логических элементах, так и в элементах ввода-вывода). Поэтому, при выходе в рабочее состояние все пользовательские выходы будут находиться в состоянии логического «0». Если режим активен, то после конфигурирования ПЛИС в процессе перехода в рабочий режим сначала переводит в неактивное состояние сигнал сброса на триггерах, а затем в неактивное состояние переводится сигнал, удерживающий пользовательские выходы в третьем состоянии. В этом случае появляется возможность вывести устройство в нужное состояние до перевода выводов в активное состояние;

- Enable DCLK output in user mode - если этот режим активен, то в рабочем режиме вывод DCLK является выходом и на него подается сигнал с внутреннего тактового генератора ПЛИС. Этот режим доступен только при активном режиме загрузки;

- Disable start-up time-out - данный режим не поддерживается ПЛИС 5576XC2T. Возможность изменения состояния активности этого режима оставлена для обеспечения совместимости с ПЛИС семейства EPF8282A ф. Altera;

- Enable JTAG support - данный режим не поддерживается ПЛИС 576XC2T. Возможность изменения состояния активности этого режима оставлена для обеспечения совместимости с ПЛИС семейства EPF8282A ф. Altera;

- Device clear - позволяет использовать в рабочем режиме в качестве глобального сигнала сброса всех пользовательских триггеров в ПЛИС (как расположенных в логических элементах, так и в элементах ввода-вывода) пользовательский вывод DEV_CLRn (I/O56, активен логическим «0»). При включении этого режима вывод DEV_CLRn должен быть зарезервирован как вход;

- Device output enable - позволяет использовать в рабочем режиме в качестве глобального сигнала управления пользовательскими элементами ввода-вывода пользовательский вывод DEV_OE (I/O55). При подаче на него логического «0» все пользовательские элементы ввода-вывода переводятся в третье состояние. При включении этого режима вывод DEV_OE должен быть зарезервирован как вход;

- Configuration over configuration - режим циклической перезаписи конфигурационной памяти. После штатной загрузки ПЛИС и выхода в рабочий режим снова происходит загрузка из ПЗУ. При этом контроль памяти не осуществляется и сигнал ошибки никогда не будет активен. Этот режим работает как с ПЗУ в активном и пассивном режимах загрузки, так и при работе с интеллектуальным хостом. При работе с ПЗУ частота обновления памяти зависит от частот внутренних генераторов ПЛИС (в активном режиме) или ПЗУ (в пассивном режиме) и по времени равна первоначальной загрузке. Включение этого режима автоматически переводит в активное состояние режим Auto-restart on frame error;

- Configuration check - режим верификации. В этом режиме при загрузке ПЛИС происходит подсчет контрольной суммы для конфигурационной памяти, которая при выходе в рабочий режим сохраняется как эталон. После выхода в рабочий режим ПЛИС циклически считывает конфигурационную память, вычисляет контрольную сумму и сравнивает ее с эталоном. При несовпадении контрольных сумм или изменении регистра режима становится активным сигнал ошибки ERR (переходит в состояние логического «0»). Далее, в зависимости от состояния режима Auto-restart on frame error либо происходит перезагрузка конфигурационной памяти, либо микросхема остается в состоянии индикации ошибки. Для уменьшения токов потребления частота проверки памяти может быть снижена в 2, 4 или 8 раз от частоты загрузки. При возникновении ошибки ПЛИС остается в рабочем режиме.

В режимах циклической перезаписи и верификации содержимого памяти происходит обновление конфигурационной памяти без выхода из рабочего режима. Поэтому необходимо, чтобы прошивка, загружаемая поверх существующей,

полностью совпадала с первоначальной, иначе возможны необратимые нарушения в ПЛИС. Это критично при разработке интеллектуального хоста загрузки ПЛИС. При загрузке ПЛИС из ПЗУ это выполняется автоматически. Режимы циклической перезаписи конфигурационной памяти и верификации содержимого памяти являются взаимоисключающими;

- режимы делителя частоты внутреннего генератора тактового сигнала: MODE_FREQ1 - делитель частоты внутреннего генератора тактового сигнала выключен; MODE_FREQ2 - уменьшение в 2 раза частоты внутреннего генератора тактового сигнала; MODE_FREQ4 - уменьшение в 4 раза частоты внутреннего генератора тактового сигнала; MODE_FREQ8 - уменьшение в 8 раз частоты внутреннего генератора тактового сигнала.

3.2 Дополнительные режимы элементов ввода-вывода

Дополнительные режимы элементов ввода—вывода:

- Bus hold — программируемая функция удержания шины в последнем состоянии. В случае, если управляющая шина находится в высокоимпедансном состоянии, при состоянии на неуправляемых выводах вблизи порогового напряжения шум на выводах может служить причиной непреднамеренного высокочастотного переключения. Цепь удержания шины в последнем состоянии доопределяет состояние на неуправляемых выводах до верхнего или нижнего логических уровней, что предотвращает «плавающее» состояние на входах и, во многих случаях, освобождает от необходимости наличия резисторов доопределени до верхнего или до нижнего логических уровней;

- Pull-up - каждый пользовательский элемент ввода-вывода имеет программируемый резистор доопределения до верхнего логического уровня в рабочем режиме. Данный режим включает этот резистор;

- Pull-down - каждый пользовательский элемент ввода-вывода имеет программируемый резистор доопределения до нижнего логического уровня в рабочем режиме. Данный режим включает этот резистор;

- Programmable Drive Strength - программируемая нагрузочная способность выходных буферов (12/24 мА) по высокому и низкому уровням сигнала. Позволяет использовать ПЛИС с устройствами, имеющими различные стандарты входов-выходов. При использовании режима повышенной нагрузочной способности (24 мА) автоматически задействуется режим низкой скорости нарастания выходного напряжения (Slew Rate).

Функции удержания шины в последнем состоянии и доопределения до верхнего или до нижнего логических уровней являются взаимоисключающими.

ConfEdit5576XC2T - программа, предназначенная для активизации дополнительных режимов работы ПЛИС 5576XC2T, а также для программирования расширенной функциональности элементов ввода—вывода (повышенная до 24 мА нагрузочная способность выходных буферов по высокому и низкому уровням сигнала, функции «Bus hold», «Pull-up», «Pull-down») на основе графического интерфейса.

Ниже (рисунок 23) приведен внешний вид главного окна программы.

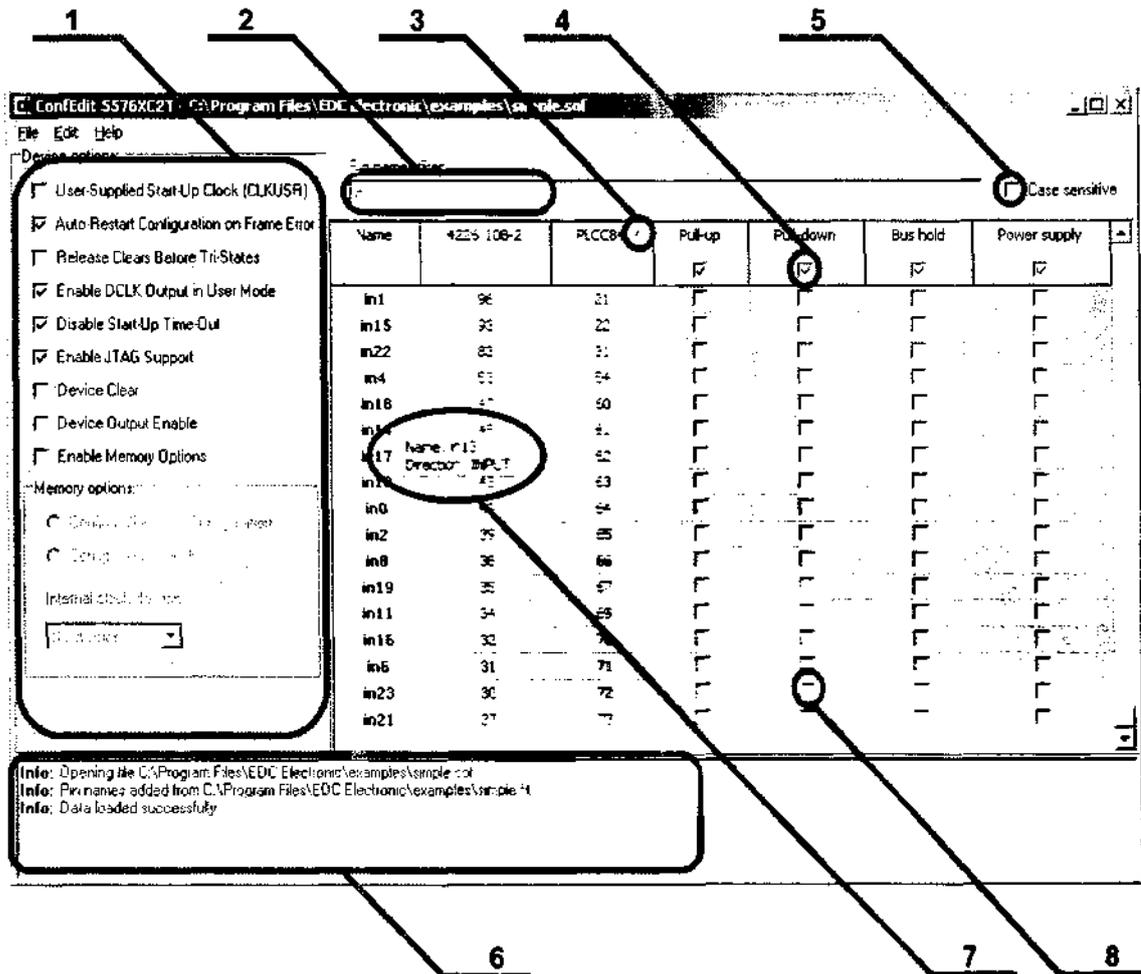


Рисунок 23 - Внешний вид главного окна программы ConfEdit5576XC2T

Окно программы содержит 4 основных элемента: панель состояния режимов устройства; таблицу, содержащую сведения об элементах ввода/вывода; поле вывода сообщений; меню.

4.1 Панель состояния режимов устройства (1)

Позволяет изменять состояния режимов устройства. Установка флага Enable Memory Options разрешает режимы верификации и циклической перезаписи конфигурационной памяти. В выпадающем списке Internal clock division задается режим делителя частоты внутреннего генератора тактового сигнала:

- No division - делитель частоты внутреннего генератора тактового сигнала выключен;
- Divide by 2 - уменьшение в 2 раза частоты внутреннего генератора тактового сигнала;
- Divide by 4 - уменьшение в 4 раза частоты внутреннего генератора тактового сигнала;
- Divide by 8 - уменьшение в 8 раз частоты внутреннего генератора тактового сигнала.

4.2 Таблица, содержащая сведения об элементах ввода—вывода

Над таблицей расположено поле фильтра имен выводов (2). В таблице показываются только те выводы, имена которых удовлетворяют регулярному выражению в данном поле. Не используемые в проекте пользовательские выводы получают имя «!». В случае, если не был подключен файл с именами выводов (автоматически создается MAX+PLUS II при компиляции проекта, имеет расширение .fit), все выводы считаются неиспользуемыми. Если поле фильтра имен выводов пусто, в таблице отображаются данные для всех выводов. Чувствительность к регистру символов для регулярного выражения, по которому фильтруется список выводов, задается флагом (5).

Таблица содержит следующие поля:

1 Name - в данном поле отображаются имена пользовательских выводов. Имена выводов, не используемых в проекте, отображаются как "RESERVED" и окрашены в серый цвет.

2 4226.108-2 - поле содержит номера выводов в корпусе 4226.108-2.

3 PLCC84 - поле содержит номера выводов в корпусе PLCC84.

4 Поля Pull-up, Pull-down, Bus hold, Power supply содержат флаги состояния режимов Pull-up, Pull-down, Bus hold и Programmable Drive Strength элементов ввода-вывода.

Заголовок поля таблицы содержит индикатор сортировки (3) и флаг состояния режима для выделенной группы выводов (4), позволяющий изменять состояние режима для группы выводов. Ячейки колонок Pull-up, Pull-down,

Bus hold, Power supply содержат флаг состояния соответствующего режима элемента ввода-вывода (8). Состояние флага изменяется двойным щелчком кнопки мыши. При наведении указателя мыши на ячейку с именем вывода появляется всплывающая подсказка (6), содержащая информацию об имени вывода и его конфигурации (вход, выход или вход-выход). Комбинации клавиш при работе с таблицей:

- Ctrl + Z - отменить действие;
- Ctrl + Y - вернуться к отмененному действию;
- Ctrl + A - выделить все строки в таблице;
- Ctrl + D - снять выделение;
- Ctrl + I - инвертировать выделение.

Удерживание клавиши Ctrl при выборе строк таблицы позволяет добавлять в выделение произвольные строки. Удерживание клавиши Shift позволяет выделять диапазон ячеек.

4.3 Поле вывода сообщений (6)

В данном поле выводиться информация о работе программы и сообщениях об ошибках.

4.4 Меню программы

Меню программы имеет следующую структуру:

1 File:

- Open - вызывает диалог открытия файла, содержащего прошивку;
- Save - сохраняет изменения в файл;
- Save As - вызывает диалог сохранения файла;
- Create script - генерирует сценарий для консольной версии программы;
- Exit - выход из программы.

2 Edit:

- Undo - отмена действия;
- Redo - возврат к отмененному действию;
- Select All - выбор всех строк в таблице;
- Select Inverse - инвертирование выделения в таблице;
- Deselect - снятие выделения в таблице.

3 Help:

- Contents - вызов документации;
- About - информация о версии программы.

5 Консольная программа

ScriptEx5576XC2T — консольная программа, предназначенная для задействования дополнительных режимов работы ПЛИС 5576XC2T, а также для программирования расширенной функциональности элементов ввода—вывода (повышенная до 24 мА нагрузочная способность выходных буферов по высокому и низкому уровням сигнала, функции «Bus hold», «Pull-up», «Pull-down») на основе сценариев. В таблице 10 приведены параметры командной строки этой программы.

Таблица 10 - Параметры командной строки программы ScriptEx5576XC2T

Параметр командной строки	Описание параметра
--input-file	Имя файла содержащего прошивку ПЛИС. Формат файла определяется по его расширению.
--output-file	¹ Имя файла, в который будут сохранены результаты работы программы. Формат файла определяется по его расширению.
--script-file	Имя файла, содержащего исполняемый сценарий.
--fit-file	Имя файла, содержащего имена выводов. Автоматически создается MAX+PLUS II при компиляции проекта, имеет расширение .fit.
--verbose -v	В случае, если этот флаг указан, программа выводит максимум информации. Этот режим имеет приоритет больший, чем режим --quiet.
--quiet 1 -q	Указывает программе выводить как можно меньше информации. При этом на консоль выводятся только сообщения об ошибках.

В процессе выполнения программы сообщения об ошибках выводятся на стандартное устройство вывода ошибок, предупреждения и другая информация выводятся на стандартное устройство вывода.

Пример:

```
ScriptEx5576XC2T --input-file=weaving.sof --script-file=test.script  
--output-file=out.sof --fit-file=weaving.fit
```

5.1 Описание языка сценариев

Сценарий представляет собой последовательность секций. Допустимы три типа секций: DEVICE_OPTIONS, PIN и ENUMERATION. Тело секции заключается в фигурные скобки. Операторы присваивания параметрам значений разделяются символом «;».

5.1.1 Секция DEVICE_OPTIONS

В секции DEVICE_OPTIONS выставляются режимы работы устройства. Если в теле секции отсутствует значение для режима, то состояние этого режима не изменится.

Список допустимых режимов:

- USER_SUPPLIED_START_UP_CLOCK;
- AUTORESTART_ON_FRAME_ERROR;
- RELEASE_CLEAR_BEFORE_TRI_STATES;
- ENABLE_DCLK_OUTPUT_IN_USER_MODE;
- DISABLE_START_UP_TIME_OUT;
- ENABLE_JTAG_SUPPORT;
- DEV_CLRN;
- DEV_OE;
- CONF_OVER;
- CONF_CHECK;
- MODE_FREQ1;
- MODE_FREQ2;
- MODEJFREQ4;
- MODE_FREQ8.

Пример:

```
DEVICE_OPTIONS  
{  
    DEV_OE = on;  
    DEV_CLRN = on;  
    USER_SUPPLIED_START_UP_CLOCK = on;  
    MODE_FREQ4 = on;  
    DISABLE_START_UP_TIME_OUT = on ;  
}
```

Режимы **MODE_FREQ1**, **MODE_FREQ2**, **MODE_FREQ4**, **MODE_FREQ8**, а также режимы **CONF_OVER** и **CONF_CFFICK** являются взаимоисключающими. При попытке установить в активное состояние взаимоисключающие режимы будет выдано сообщение об ошибке. Для предотвращения появления ошибки в ситуации, когда, например, во входном файле активен режим **CONF_OVER** и необходимо установить режим **CONF_CFFICK**, достаточно явным образом указать, что режим **CONF_OVER** выключен. Следующий код приведет к ошибке, если во входном файле включен режим **CONF_OVER**:

```
DEVICE_OPTIONS  
{  
    CONF_CHECK = on;  
}
```

Решение:

```
DEVICE_OPTIONS  
{  
    CONF_OVER = off;  
    CONF_CHECK = on;  
}
```

Порядок задания режимов в теле секции имеет значение. Следующий код так же приведет к ошибке в случае, когда во входном файле активен режим **CONF_OVER**.

DEVICE_OPTIONS

```
{  
    CONF_CHECK = on;  
    CONF_OVER  = off;  
}
```

5.1.2 Секция PIN

После ключевого слова PIN в скобках указывается регулярное выражение для имени вывода. Выражение заключается в апострофы. Указанные в теле секции режимы будут выставлены на всех выводах, имена которых удовлетворяют регулярному выражению. Не используемые в проекте выводы получают имя «!». В случае если не задан файл с именами выводов, все выводы считаются неиспользуемыми. Не указанные в теле секции режимы считаются неактивными.

Список допустимых режимов:

- PULLUP;
- PULLDOWN;
- BUSHOLD;
- PWRSUPPLY.

Пример:

PIN('.*')

```
{  
    PWRSUPPLY = on/  
    BUSHOLD   = on;  
}
```

5.1.3 Секция ENUMERATION

После ключевого слова ENUMERATION в скобках перечисляются номера выводов (в корпусе 4226.108-2). Номера выводов указываются через запятую. Допустимы также диапазоны вида «37-40». Диапазон не должен содержать номеров выводов, не являющихся пользовательскими. Указанные в теле секции режимы будут выставлены на всех перечисленных выводах. Не указанные в теле секции режимы считаются неактивными.

Список допустимых режимов:

- PULLUP;
- PULLDOWN;
- BUSHOLD;
- PWRSUPPLY.

Пример:

```
ENUMERATION(4,5,7,13,50-53)
```

```
{  
    PULLUP = on;  
    PULLDOWN = off;  
    BUSHOLD = off;  
    PWRSUPPLY = off;  
}
```

5.2 Синтаксис языка сценариев

Пробельные символы, находящиеся вне строковых литералов, а также символы перевода строки игнорируются. Поддерживаются однострочные комментарии, начинающиеся последовательностью символов «//» и продолжающиеся до конца строки. Язык сценария является регистронезависимым. Сценарий представляет собой последовательность секций.

сценарий:

[секция]...

секция:

*идентификатор секции [параметры секции]
тело секции*

идентификатор секции:

PIN | ENUMERATION DEVICE_OPTIONS

параметры секции:

(строковый литерал\перечисление)

строковый литерал:

'любая последовательность символов'

перечисление:

диапазон \ число [, диапазон \ число]...

диапазон:

число — число

тело секции:

{присвоение значения параметру...}

присвоение значения параметру:

параметр = значение параметра;

параметр:

PULLUP | PULLDOWN | BUSHOLD |
PWRSUPPLY |
USER_SUPPLIED_START_UP_CLOCK
AUTORESTART_ON_FRAMEJERROR |
RELEASE_CLEAR_BEFORE_TRI_STATES
ENABLE_DCLK_OUTPUT_IN_USER_MODE|
DISABLE_START_UP_TIME_OUT |
ENABLE_JTAG_SUPPORT | DEV_CLRN DEV OE
| CONF_OVER CONF_CHECK MODE_FREQ1 |
MODE_FREQ2 | MODE_FREQ4 | MODE_FREQ8

значение параметра:

true | false | on off

6 Программный интерфейс

6.1 Заголовочные файлы

Предоставляются заголовочные файлы для языков программирования Pascal (core5576XC2T.pas; core5576XC2Tconst.pas) и C/C++ (core5576XC2T.h), содержащие определения функций, типов и константы.

Также доступны примеры использования программного интерфейса для каждого из этих языков.

6.2 Описание функций

6.2.1 Функция chip_open

Загружает прошивку.

Pascal:

```
function chip_open  
(  
    FileName : PChar;  
    Format : integer;  
    FitFileName : PChar  
): TChipHandle; stdcall;
```

C/C++:

```
TChipHandle ____stdcall chip_open  
(  
    char* FileName,  
    int Format,  
    char* FitFileName  
);
```

Параметры:

1 FileName - указатель на строку, содержащую имя файла с прошивкой;
2 Format - формат файла содержащего прошивку. Допустимые значения: fmt_SOF, fmt_RBF;

3 FitFileName - указатель на строку, содержащую имя файла с именами выводов. Этот файл создается MAX+PLUS II при компиляции проекта, имеет расширение .fit. Неиспользуемые выводы получают имя «!». В случае, если этот файл не используется, необходимо передать в качестве значения параметра nil (Pascal) или NULL (C/C++). При этом все выводы считаются неиспользуемыми.

Возвращаемое значение: в случае успешного выполнения функция возвращает дескриптор прошивки, в случае ошибки возвращаемое значение равно INVALID_CHIP_HANDLE. Полученный дескриптор должен быть впоследствии освобожден с помощью функции chip_close.

6.2.2 Функция chip_close

Закрывает прошивку, ранее открытую функцией chip_open.

Pascal:

```
function chip_close(ChipHandle: TChipHandle): integer;  
stdcall;
```

C/C++:

```
int __stdcall chip_close (TChipHandle  
ChipHandle)/
```

Параметр ChipHandle - дескриптор прошивки, возвращаемый функцией chip_open.

Возвращаемое значение: в случае успешного выполнения возвращаемое значение равно r_OK, в случае ошибки - r_ERR.

6.2.3 Функция chip_set_device_option

Устанавливает режим устройства.

Pascal:

```
function chip_set_device_option  
(  
  ChipHandle : TChipHandle;  
  DeviceOption : integer;  
  State : integer  
): integer; stdcall;
```

C/C++:

```
int __stdcall chip_set_device_option  
(  
    TChipHandle ChipHandle,  
    int DeviceOption,  
    int State  
);
```

Параметры:

1 ChipHandle - дескриптор прошивки, возвращаемый функцией chip_open;

2 DeviceOption - режим. Допустимые значения:

- dev_USER_SUPPLIED_START_UP_CLOCK;
- dev_AUTORESTART_ON_FRAME_ERROR;
- dev_RELEASE_CLEAR_BEFORE_TRI_STATES;
- dev_ENABLE_DCLK_OUTPUT_IN_USER_MODE;
- dev_DISABLE_START_UP_TIME_OUT;
- dev_ENABLE_JTAG_SUPPORT;
- dev_DEV_CLRN;
- dev_DEV_OE;
- dev_CONF_OVER;
- dev_CONF_CHECK;
- dev_MODE_FREQ1;
- dev_MODE_FREQ2;
- dev_MODE_FREQ4;
- dev_MODE_FREQ8;

3 State - устанавливаемое состояние режима. Значение «0» - режим выключен, любое другое значение - включен.

Возвращаемое значение: в случае успешного выполнения возвращаемое значение равно r_OK, в случае некорректного дескриптора - r_WRONG_HANDLE, в случае некорректного значения DeviceOption или если режим не может быть установлен (например, нужный вывод не зарезервирован как вход) - r_WRONG_OPTION.

6.2.4 Функция chip_get_device_option

Позволяет получить текущее состояние режима устройства.

Pascal:

```
function chip_get_device_option  
  (  
    ChipHandle : TChipHandle;  
    DeviceOption : integer;  
    State : PInteger  
  ): integer; stdcall/
```

C/C++:

```
int ___ stdcall chip_get_device_option  
  (  
    TChipHandle ChipHandle,  
    int DeviceOption,  
    int* State  
  );
```

Параметры:

1 ChipHandle - дескриптор прошивки, возвращаемый функцией chip_open;

2 DeviceOption - режим. Допустимые значения:

- dev_USER_SUPPLIED_START_UP_CLOCK;
- dev_AUTORESTART_ON_FRAME_ERROR;
- dev_RELEASE_CLEAR_BEFORE_TRI_STATES;
- dev_ENABLE_DCLK_OUTPUT_IN_USER_MODE;
- dev_DISABLE_START_UP_TIME_OUT;
- dev_ENABLE_JTAG_SUPPORT;
- dev_DEV_CLRN;
- dev_DEV_OE;
- dev_CONF_OVER;
- dev_CONF_CHECK;
- dev_MODE_FREQ1;
- dev_MODE_FREQ2;
- dev_MODE_FREQ4;
- dev_MODE_FREQ8;

3 State - указатель на переменную, в которую будет возвращено текущее состояние режима. Значение: «0» - режим выключен, «1» - включен.

Возвращаемое значение: в случае успешного выполнения возвращаемое значение равно `r_OK`, в случае некорректного дескриптора - `r_WRONG_HANDLE`, в случае некорректного значения `DeviceOption` - `r_WRONG_OPTION`, в случае иной ошибки - `r_JERR`.

6.2.5 Функция `chip_set_pin_options`

Устанавливает режим работы элемента ввода/вывода.

Pascal:

```
function chip_set_pin_options
(ChipHandle : TChipHandle;
 index : integer/
 PinOptions : integer
): integer; stdcall; overload;
```

C/C++:

```
int __stdcall chip_set_pin_options
(TChipHandle ChipHandle,
 int index,
 int PinOptions
);
```

Параметры:

- 1 `ChipHandle` - дескриптор прошивки, возвращаемый функцией `chip_open`;
- 2 `Index` - номер вывода в корпусе 4226.108-2;
- 3 `PinOptions` - комбинация флагов `pin_BUSHOLD`, `pin_PULLDOWN`, `pin_PULLUP`, `pin_PWRSUPPLY`.

Возвращаемое значение: в случае успешного выполнения возвращаемое значение равно `r_OK`, в случае некорректного дескриптора - `r_WRONG_HANDLE`, в случае иной ошибки - `r_ERR`.

6.2.6 Функция chip_set_pin_options_by_name

Устанавливает режим работы элемента ввода-вывода.

Pascal:

```
function chip_set_pin_options  
  (  
    ChipHandle : TChipHandle;  
    NameExpression : PChar; PinOptions :  
    integer ): integer; stdcall; overload;
```

C/C++:

```
int ___ stdcall chip_set_pin_options_by_name  
  (  
    TChipHandle ChipHandle,  
    char* NameExpression,  
    int PinOptions  
  );
```

Параметры:

1 **ChipHandle** - дескриптор прошивки, возвращаемый функцией **chip_open**;

2 **NameExpression** - регулярное выражение для имени вывода. Информация об именах выводов передается при открытии прошивки функцией **chip_open**. В случае, если функции **chip_open** не был передан файл с именами выводов, все выводы считаются неиспользуемыми. Неиспользуемые выводы получают имя «!». Режимы будут установлены на все выводы, имена которых удовлетворяют регулярному выражению;

3 **PinOptions** - комбинация флагов **pin_BUSHOLD**, **pin_PULLDOWN**, **pin_PULLUP**, **pin_PWRSUPPLY**.

Возвращаемое значение: в случае успешного выполнения возвращаемое значение равно **r_OK**, в случае некорректного дескриптора - **r_WRONG_HANDLE**, в случае некорректного имени вывода - **r_WRONG_PIN_NAME**, в случае иной ошибки - **r_ERR**.

6.2.7 Функция chip_get_pin_options

Возвращает текущую конфигурацию элемента ввода-вывода.

Pascal:

```
function chip_get_pin_options  
(  
    ChipHandle : TChipHandle;  
    index      : integer;  
    PinOptions : PInteger  
): integer; stdcall; overload;
```

C/C++:

```
int __ stdcall chip_get_pin_options  
(  
    TChipHandle ChipHandle,  
    int         Index,  
    int*        PinOptions  
);
```

Параметры:

1 ChipHandle - дескриптор прошивки, возвращаемый функцией chip_open;

2 Index - номер вывода в корпусе 4226.108-2;

3 PinOptions - указатель на переменную, в которую будет возвращена информация о текущей конфигурации элемента ввода/вывода, являющаяся комбинацией флагов:

- pin_IN - элемент сконфигурирован как вход;
- pin_OUT - элемент сконфигурирован как выход;
- pin_IO - элемент сконфигурирован как вход-выход;
- pin_BUSHOLD - режим удержания шины в последнем активном состоянии:

нии:

- pin_PULLDOWN - разрешение резистора доопределения сигнала до нижнего логического уровня;

- pin_PULLUP - разрешение резистора доопределения сигнала до верхнего логического уровня;

- pin_PWRSUPPLY - режим повышенной нагрузочной способности (24 мА).

Возвращаемое значение: в случае успешного выполнения возвращаемое значение равно r_OK, в случае некорректного дескриптора - r_WRONG_HANDLE, в случае иной ошибки - r_ERR.

6.2.8 Функция chip_get_pin_options_by_name

Возвращает текущую конфигурацию элемента ввода-вывода.

Pascal:

```
function chip_get_pin_options  
  (  
    ChipHandle : TChipHandle;  
    Name       : PChar;  
    PinOptions : PInteger  
  ): integer; stdcall; overload;
```

C/C++:

```
int ___ stdcall chip_get_pin_options_by_name  
  (  
    TChipHandle ChipHandle,  
    char* Name,  
    int* PinOptions  
  );
```

Параметры:

1 ChipHandle - дескриптор прошивки, возвращаемый функцией chip_open;

2 Name - указатель на строку, содержащую имя вывода. Неиспользуемые выходы получают имя «!»;

3 PinOptions - указатель на переменную, в которую будет возвращена информация о текущей конфигурации элемента ввода-вывода, являющаяся комбинацией флагов:

- pin_IN — элемент сконфигурирован как вход;
- pin_OUT — элемент сконфигурирован как выход;
- pin_IO - элемент сконфигурирован как вход-выход;
- pin_BUSHOLD - режим удержания шины в последнем активном состоянии;
- pin_PULLDOWN - разрешение резистора доопределения сигнала до нижнего логического уровня;
- pin_PULLUP - разрешение резистора доопределения сигнала до верхнего логического уровня;
- pinJPWRSUPPLY - режим повышенной нагрузочной способности (24 мА).

Возвращаемое значение: в случае успешного выполнения возвращаемое значение равно `r_OK`, в случае некорректного дескриптора - `_WRONG_HANDLE`, в случае некорректного имени вывода - `r_WRONG_PIN_NAME`, в случае иной ошибки - `r_ERR`.

6.2.9 Функция `chip_save`

Сохраняет прошивку в файл.

Pascal:

```
function chip_save  
(  
    ChipHandle : TchipHandle;  
    FileName : Pchar;  
    Format : integer  
): integer; stdcall;
```

C/C++:

```
int __stdcall chip_save  
(  
    TChipHandle ChipHandle,  
    char* filename,  
    int format  
);
```

Параметры:

- 1 `ChipHandle` - дескриптор прошивки, возвращаемый функцией `chip_open`;
- 2 `FileName` - указатель на строку, содержащую имя файла, в который должна быть сохранена прошивка;
- 3 `Format` - формат файла содержащего прошивку. Допустимые значения: `fmt_SOF`, `fmt_RBF`, `fmt_POF`.

Возвращаемое значение: в случае успешного выполнения возвращаемое значение равно `r_OK`. в случае некорректного дескриптора - `r_WRONG_HANDLE`, в случае иной ошибки - `r_ERR`.