ЛЕКЦИЯ 2-11

GA, FPGA, FLEX

Кафедра КИТС Кораблев Н.М.

Базовые матричные кристаллы (GA)

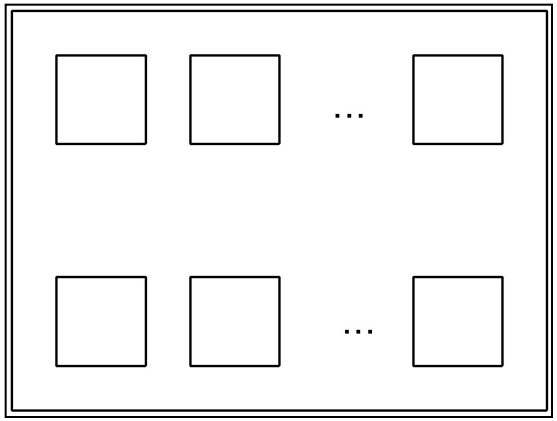
БМК – это кристалл, на прямоугольной поверхности которого размещены внутренние и периферийные области.

Во внутренней области по строкам и столбцам в виде матицы расположены базовые ячейки – это группы некоммутируемых схемных элементов (транзисторов, резисторов как вентилей И-НЕ, ИЛИ-НЕ)

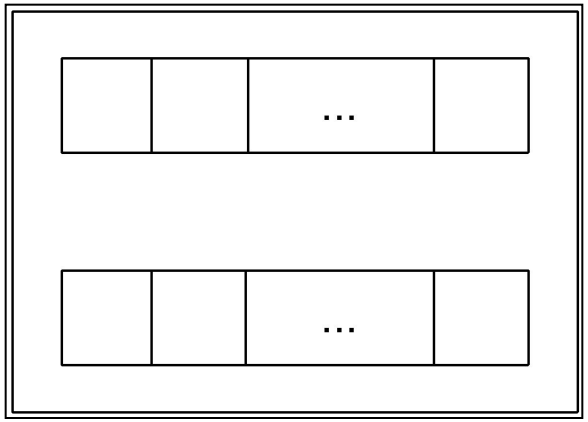
Элементарный состав базовой ячейки при разных вариантах межсоединений элементов допускает реализацию некоторого множества схем базового класса, каждая из которых соответствует определенной функциональной ячейке.

В зависимости от степени интеграции БМК могут иметь: канальную, бесканальную и блочную архитектуры.

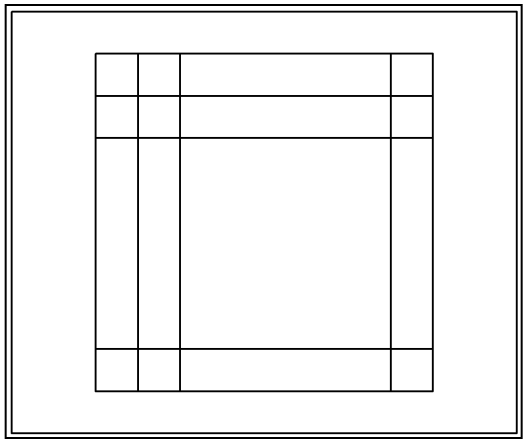
В БМК канальной структуры между строками и столбцами базовых ячеек оставляются вертикальные и горизонтальные свободные зоны для соединений.



или



Для снижения потерь площади кристалла затрачиваемой на трассировочные каналы используется бесканальная структура типа:



На основе КМОП-транзисторов.

Для организации связи сами базовые ячейки.

Из элементов базовой матрицы ячеек может быть сформирован один логический элемент, тогда для реализации более сложных функций используется несколько базовых ячеек.

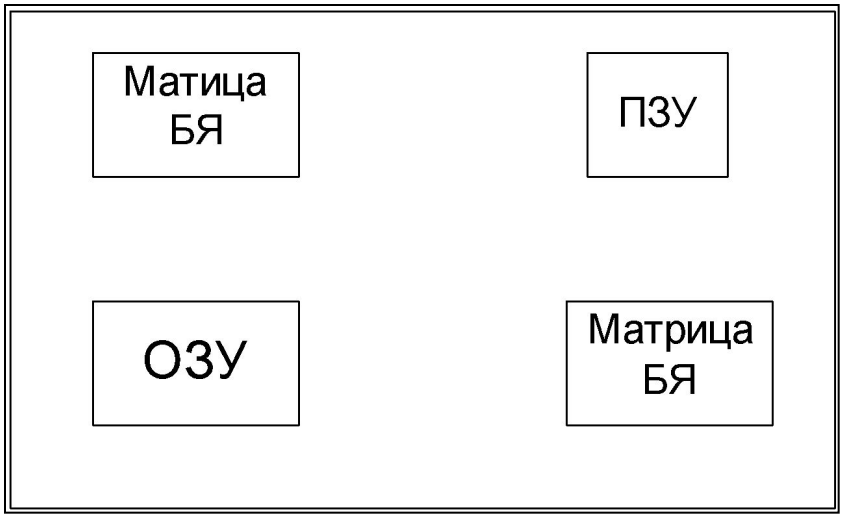
Из элементов базовой матицы ячеек может быть сформирован функциональный узел, а состав элементов ячейки определяется схемой самого сложного узла.

Рост уровня интеграции привел к возможности реализации на одном кристалле более сложных устройств, что стало возможным благодаря созданию БМК блочного типа (с блочной структурой)

В таких БК на кристалле выделяются специализированные области – подматрицы для выполнения заранее определенных функций:

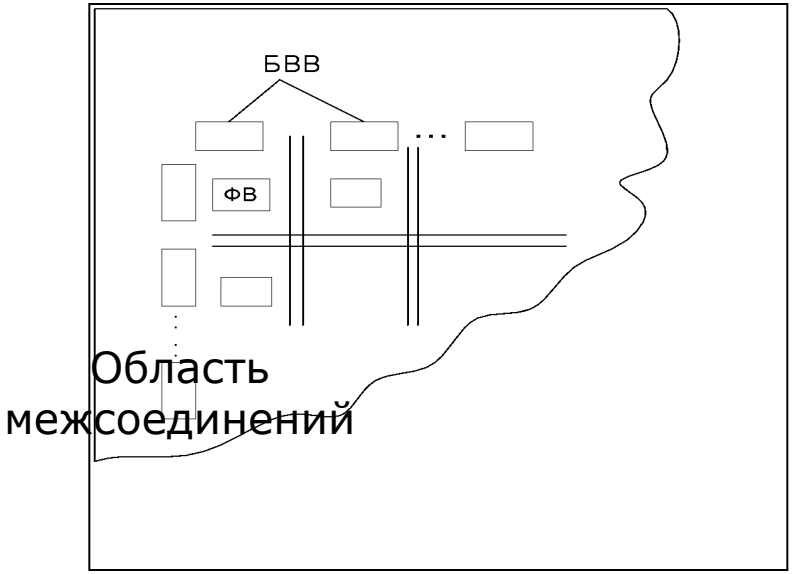
- Логическое образование данных
- Постоянной или переменной памяти и др.

Между подматрицами реализуются специальные трассировочные каналы. На периферии подматриц имеются внутренние периферийные ячейки для передачи сигналов.



Программируемые пользователем вентильные матрицы FPGA

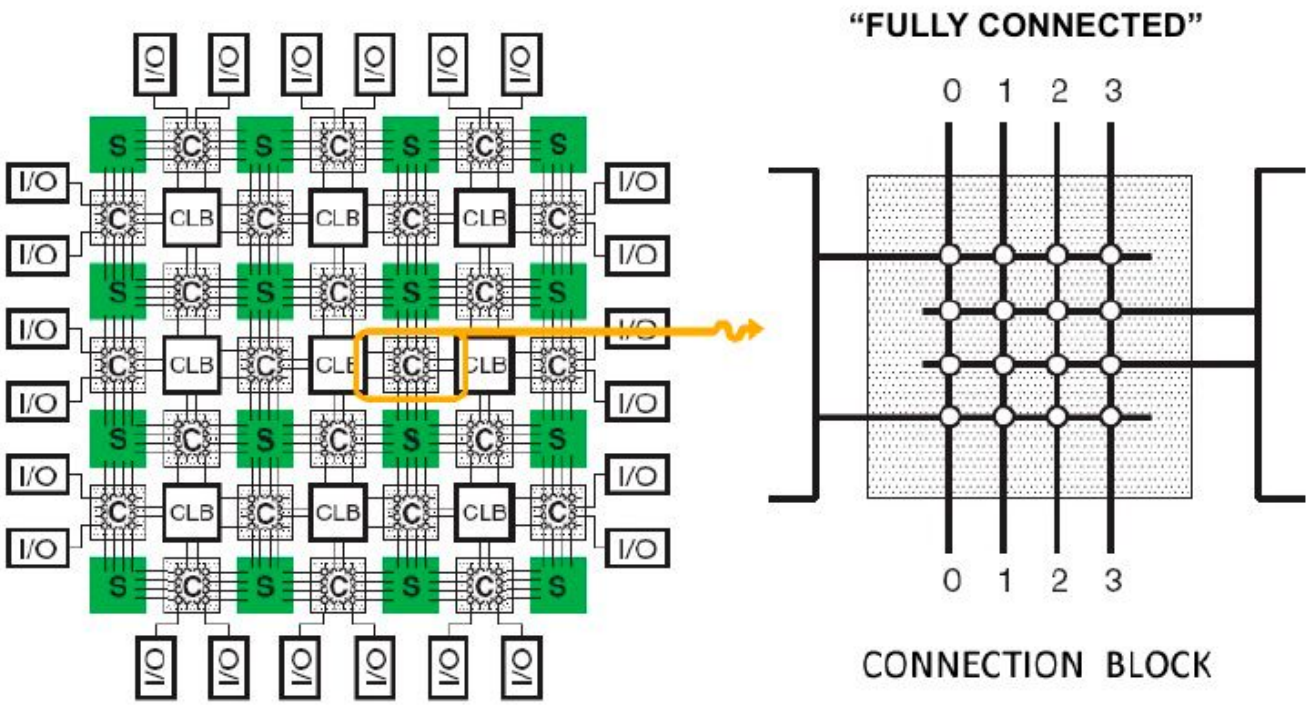
FPGA представляют собой микросхемы высокого уровня интеграции, содержащие во внутренней области матрицу идентичных функциональных блоков (ФБ) систему их межсоединений, размещенную между строками и столбцами матрицы, а в периферийной области блоки ввода-вывода (БВВ).



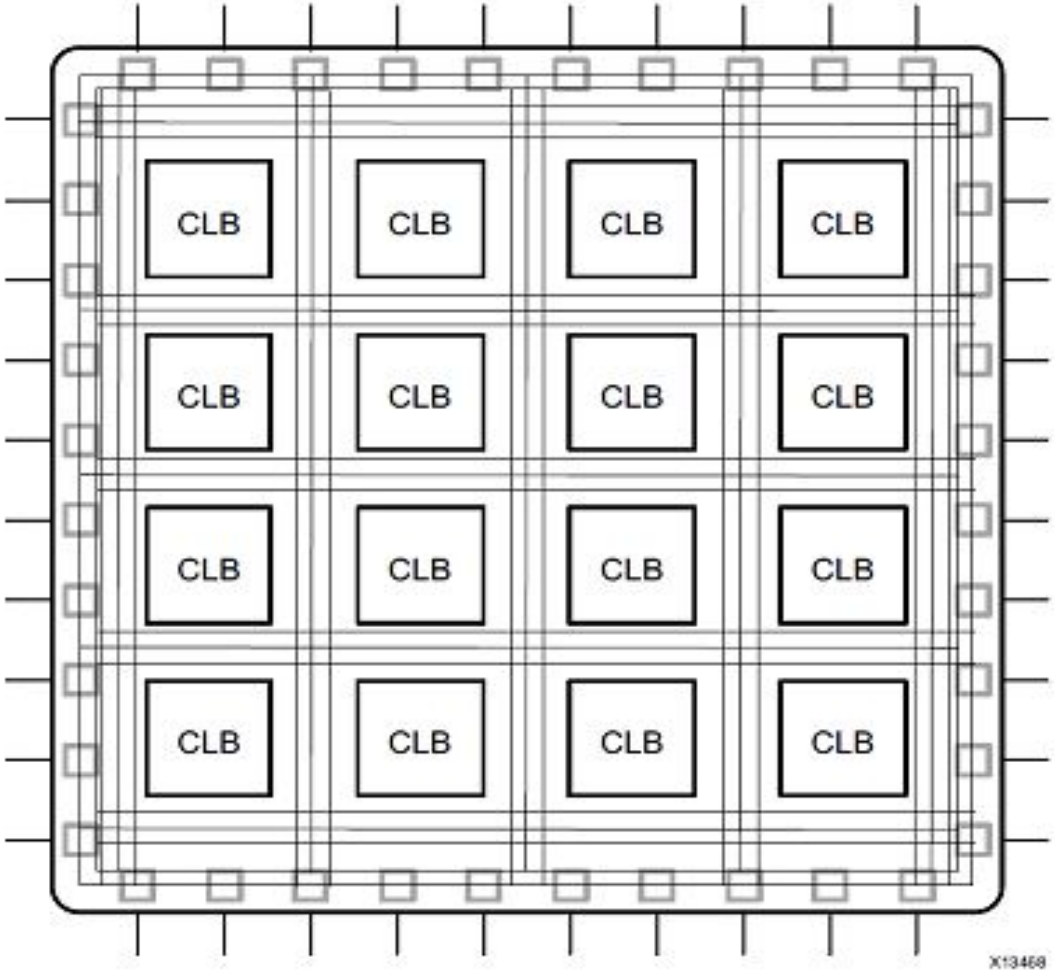
Все части FPGA являются конфигурируемыми (реконфигурируемыми) причем, в отличие от вычислительных матриц, средствами пользователей.

При конфигурировании FPGA ФБ настраиваются на выполнение необходимых операций преобразования данных, а система межсоединений на требуемые связи между ФБ.

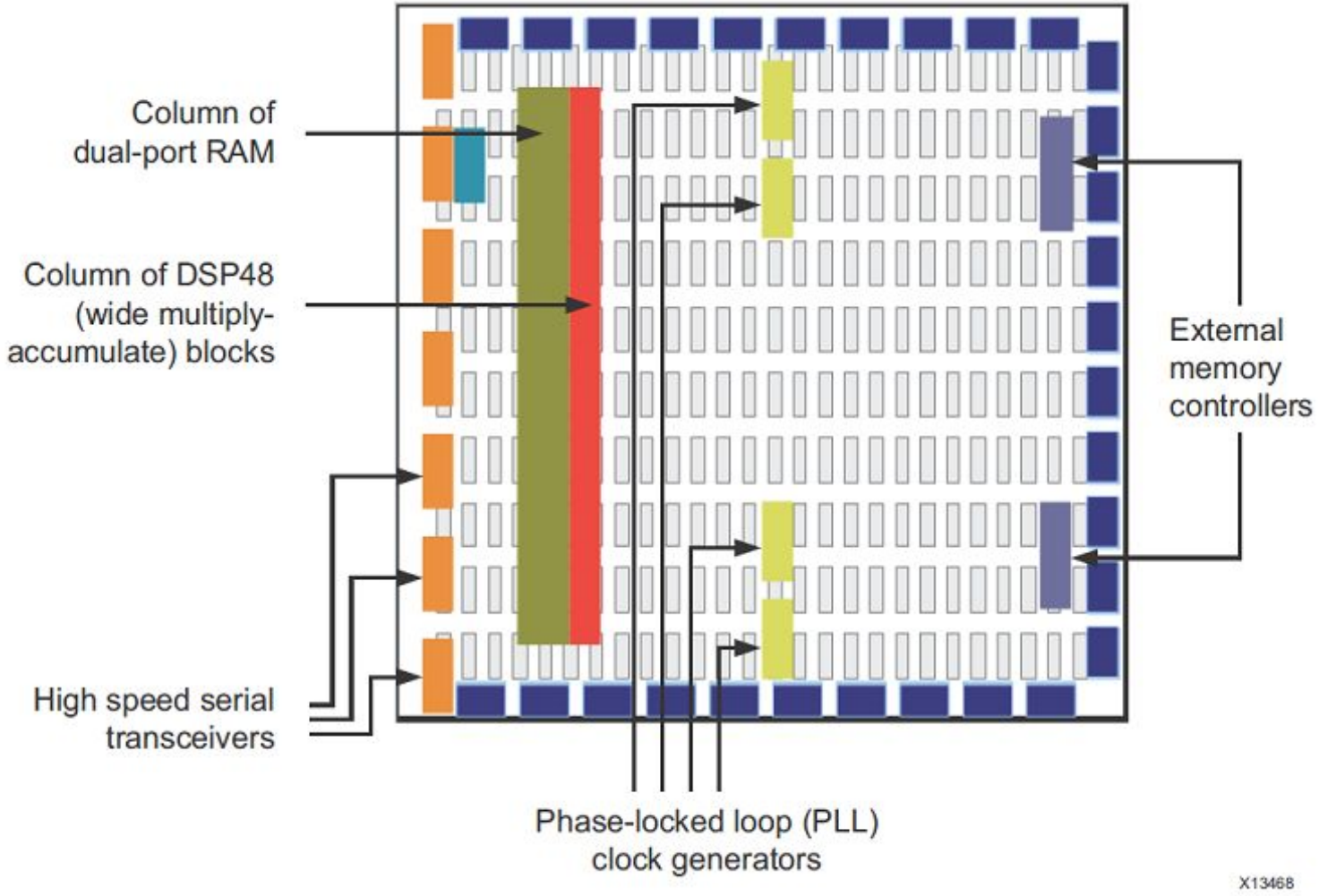
FPGA Basics



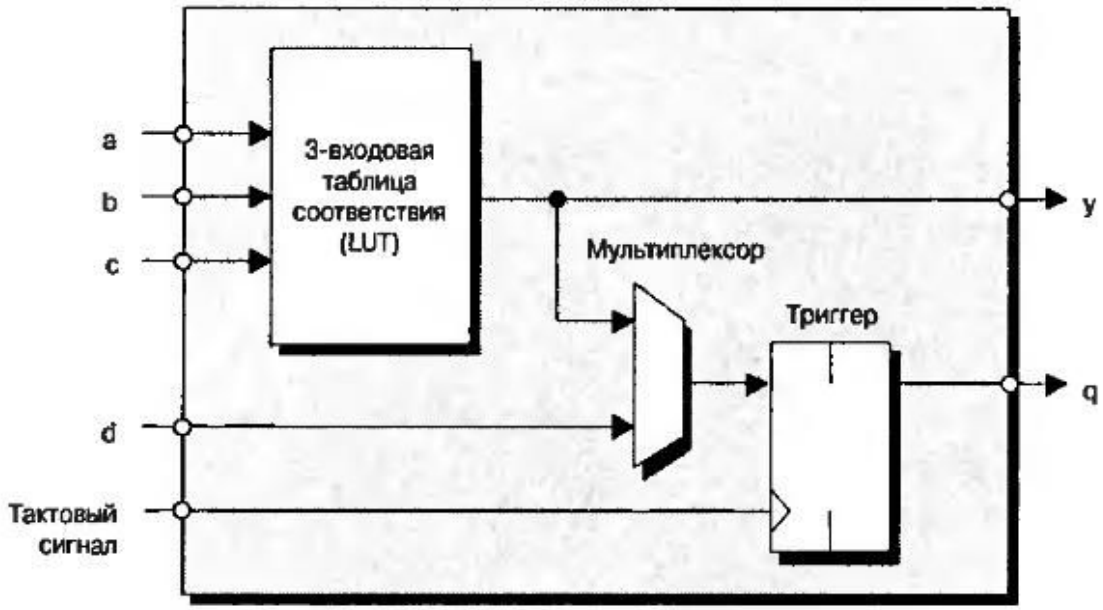
Архитектура ПЛИС FPGA: CLB и Interconnect



Специализированное оборудование на ПЛИС FPGA

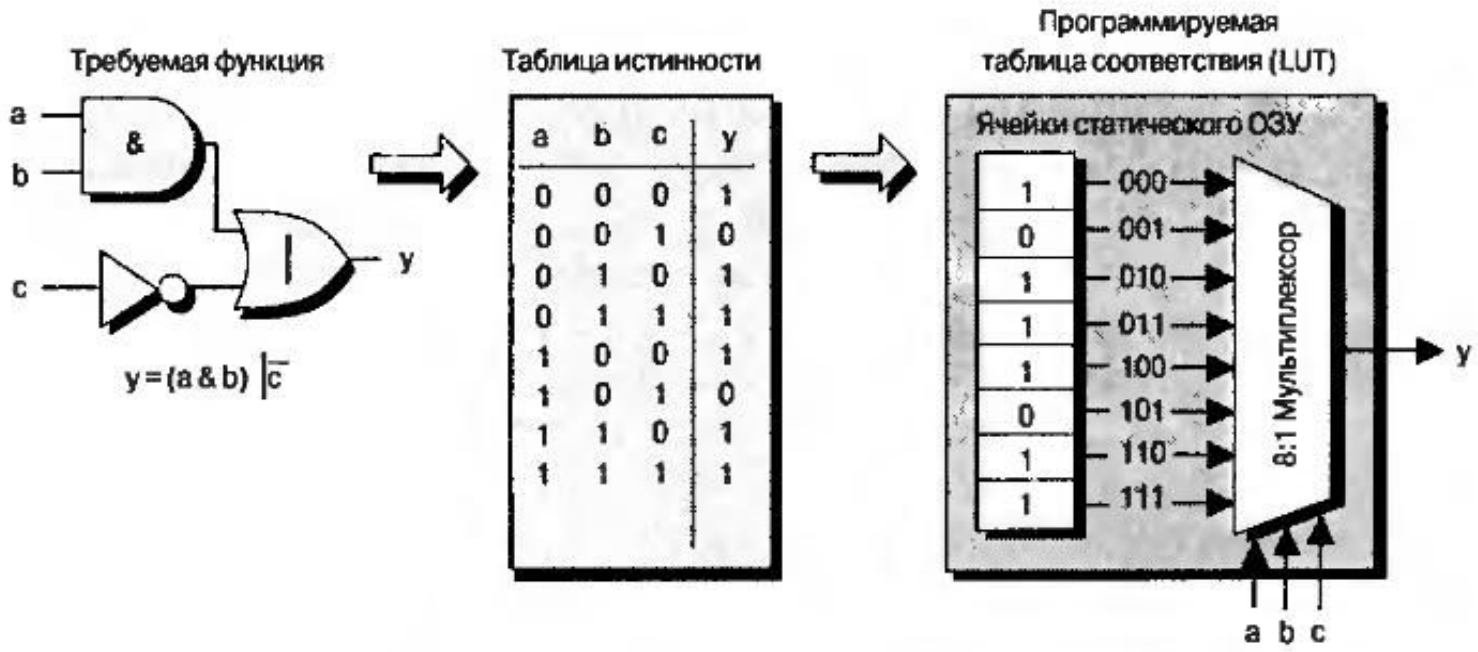


CLB упрощенно состоит из блока, задающего булеву функцию от нескольких аргументов (она называется таблицей соответствия — Up Table, LUT) и триггера (flip-flop, FF)



В современных FPGA LUT имеет шесть входов. Выход LUT подается на выход CLB либо асинхронно (напрямую), либо синхронно (через триггер FF, работающий на системной тактовой частоте).

Принцип реализации LUT



Пусть есть некоторая булева функция y = (a & b) | ~ c. Ее схемотехническое представление и таблица истинности показаны на рисунке.

Функциональные блоки

В состав типичного ФБ входят:

- Функциональный преобразователь (ФП)
- Триггер
- Мультиплексоры, которые используются для конфигурации ФБ

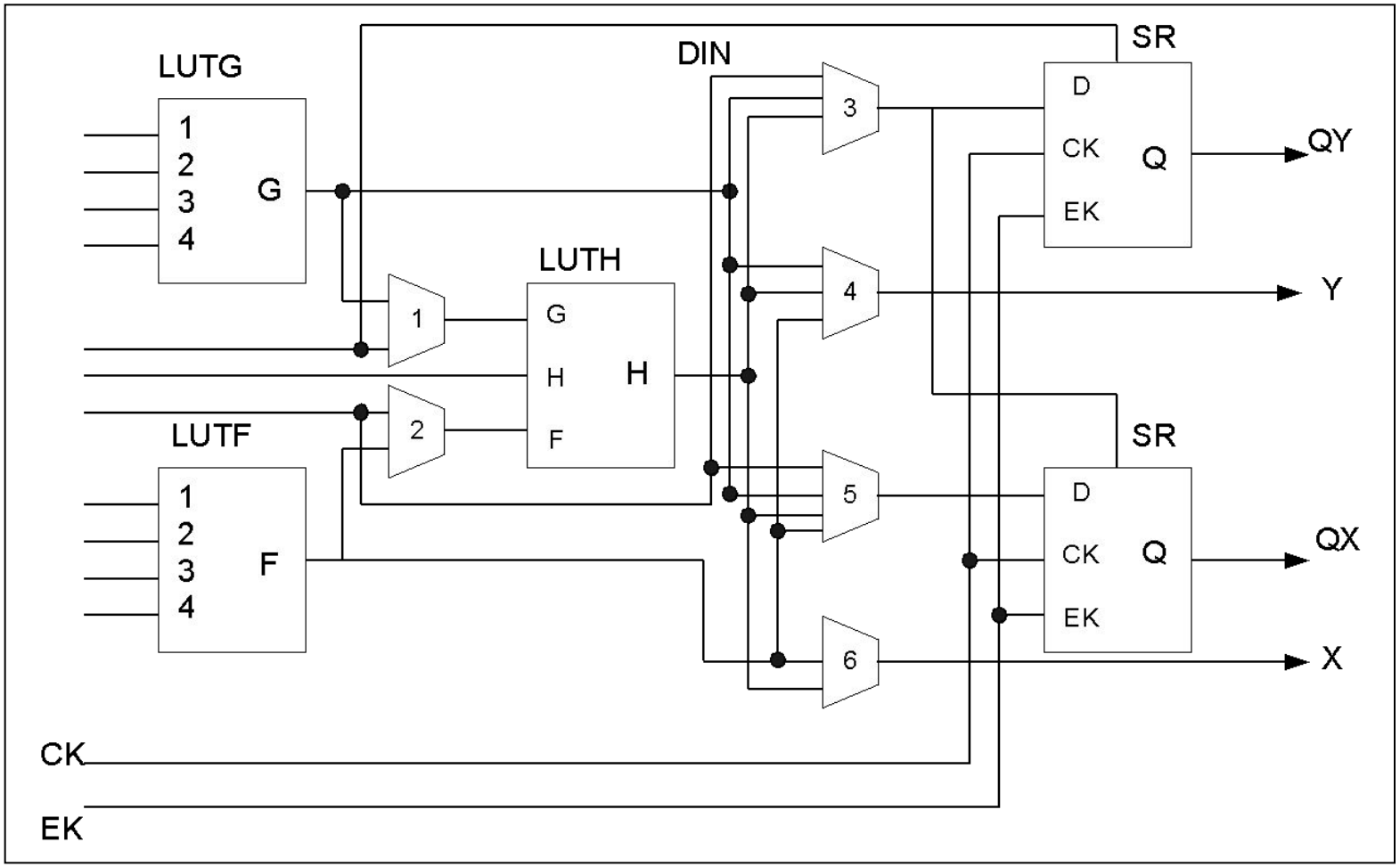
ФП могут быть выполнены в виде простых логических вентилей, логических модулей на основе мультиплексора, ПЗУ (LUT).

Типичный ФП типа LUT представляет собой ЗУ, хранящее значение искомых функций, m-адресов и n-входов (2mxn организация).

В случае реализации ФП на программируемом мультиплексоре выходная величина описывается некоторой порождающей функцией, соответствующей использованию всех входов схемы как информационных.

При реализации на простых логических вентилях (мелкозернистые ФП), блоки составляются чаще всего из транзисторных комплиментарных пар, из которых собираются КМОП логические элементы.

Рассмотрим типичную структуру ФБ на основе LUT



Имеющиеся ресурсы логической части ФБ позволяют воспроизводить:

1. Любую функцию с числом аргументов до четырех включительно, плюс вторую такую же функцию, плюс любую функцию с числом аргументов до трех.
2. Любую, но только одну функцию пяти аргументов.
3. Любую функцию четырех аргументов и одновременно некоторую функцию шести аргументов.
4. Некоторые функции с числом аргументов до девяти.

Блоки ввода-вывода

БВВ обеспечивают интерфейс между выводами корпуса и ее внутренними логическими схемами.

Каждому выводу придается БВВ, который может быть конфигурирован на:

- Вход
- Выход
- Двунаправленный обмен.

С этой целью в каждый БВВ входит:

- Входной буфер
- Выходной буфер
- Входной триггер
- Выходной триггер
- Мультиплексор для конфигурирования
- Схемы управления

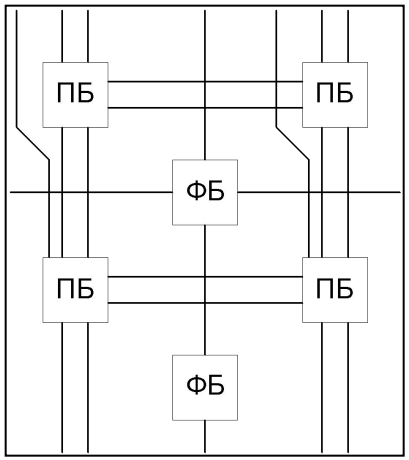
Система межсоединений

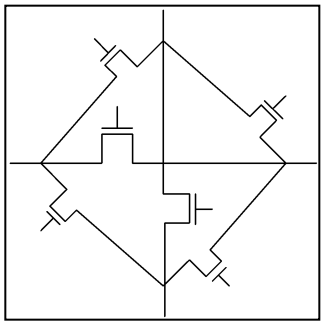
Для системы межсоединений в FPGA организация непрерывных связей затруднена. Для них характерны сегментированные линии связей, составленные из отдельных проводящих сегментов.

Сегменты соединяются в нужную цепь с помощью программируемых ключей.

Система межсоединений имеет иерархический характер, в ней сочетаются различные типы сегментов: основные связи, связи двойной длины, прямые связи для близлежащих ФБ, длинные линии, пересекающие кристаллпо всей его длине или ширине.

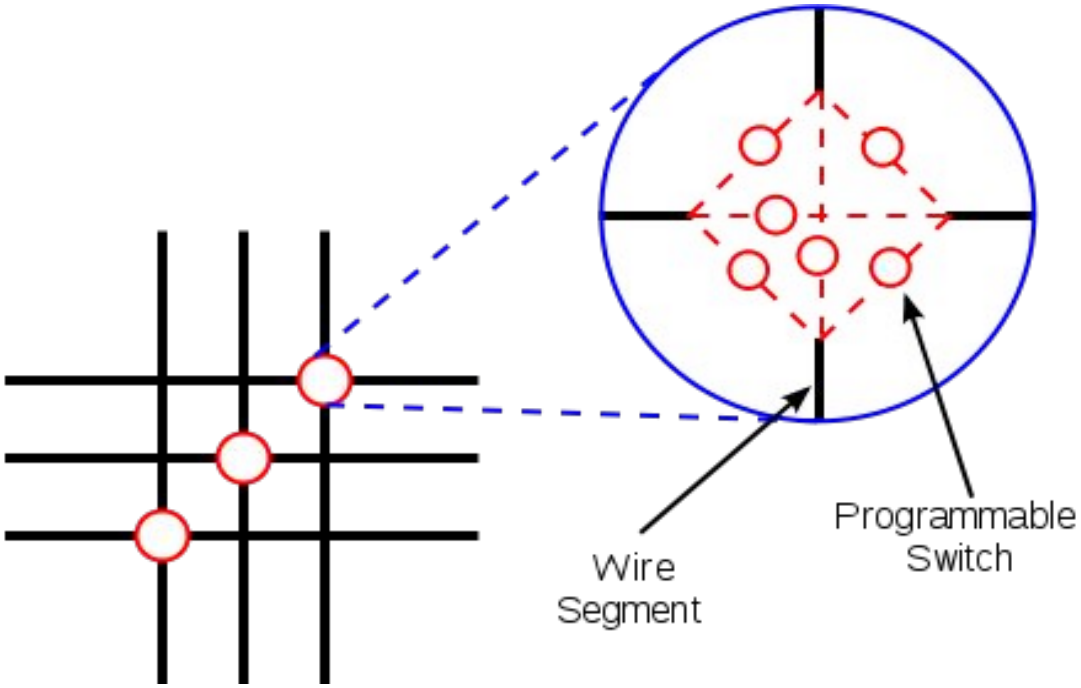
Система межсоединений образует сегментируемые линии и переключательные блоки, которые образованы на пересечении вертикальных и горизонтальных каналов.





В число FPGA по принятой классификации попадают микросхемы с числом эквивалентных вентилей от 100 до 15 000 с системными частотами от 50 до 85 МГц и числом пользовательских выводов 100-300.

Блоки CLB находятся в коммутационной матрице, которая задает соединения входов и выходов блоков CLB



На каждом пересечении проводников находится шесть переключающих ключей, управляемых своими ячейками конфигурационной памяти. Открывая одни и закрывая другие, можно обеспечить разную коммутацию сигналов между CLB

Сравнительный обзор CPLD и FPGA

№	CPLD	FPGA
1	Немедленное включение. CPLD начинают работать, как только они включены	Поскольку FPGA должна загрузить данные конфигурации из внешнего ПЗУ и настроить матрицу до того, как она сможет начать функционировать, между включением питания и FPGA начинает работать задержка. Задержка может составлять несколько десятков миллисекунд.
2	Энергонезависимая. CPLD остаются запрограммированными и сохраняют свою цепь после выключения. ПЛИС гаснут сразу после выключения.	ПЛИС использует хранилище конфигурации на основе SRAM. Содержимое памяти теряется, как только отключается питание.
3	Детерминированный анализ времени. Поскольку CPLD сравнительно проще по сравнению с FPGA, а количество межсоединений меньше, анализ времени можно выполнить гораздо проще.	Размер и сложность логики FPGA могут быть огромными по сравнению с CPLD. Это открывает возможность менее детерминированной маршрутизации сигнала и, таким образом, вызывает сложные временные сценарии. К счастью, инструменты реализации, предоставляемые поставщиками ПЛИС, имеют механизмы, помогающие добиться детерминированной синхронизации. Но для этого обычно необходимы дополнительные действия пользователя.
4	Более низкое энергопотребление на холостом ходу. Более новые CPLD, такие как CoolRunner-II, потребляют около 50 мкА в режиме ожидания.	Относительно выше энергопотребление на холостом ходу.
5	Может быть дешевле для реализации более простых схем	FPGA гораздо более эффективны по сравнению с CPLD, но могут быть и более дорогими.

Сравнительный обзор CPLD и FPGA

№	CPLD	FPGA
6	Более «безопасный» благодаря дизайну хранилища во встроенной энергонезависимой памяти.	ПЛИС, которые используют внешнюю память, могут выставлять IP извне. Многие поставщики ПЛИС предлагают такие механизмы, как шифрование, для борьбы с этим. Разработка конкретных механизмов защиты также может быть реализована.
7	Очень небольшое количество логических ресурсов.	Огромное количество логических и запоминающих элементов, с помощью которых могут быть разработаны невероятно сложные схемы. FPGA имеют в тысячи раз больше ресурсов! Уже один этот момент делает FPGA более популярными, чем CPLD.
8	Нет встроенных жестких IP-адресов для разгрузки обработки из логической структуры.	Разнообразие встроенных аппаратных средств, таких как блочное ОЗУ, блоки DSP, PLL, DCM, контроллеры памяти, мультигигабитные трансиверы и т. Д., Обеспечивают огромную гибкость. Это даже не мыслимо с CPLD.
9	Выключение питания и перепрограммирование всегда требуются для изменения функциональности дизайна.	ПЛИС могут менять свою схему даже во время работы! (Так как это просто вопрос обновления LUT с другим содержимым) Это называется частичной реконфигурацией, и она очень полезна, когда FPGA необходимо продолжать проектирование и в то же время обновлять его с другим дизайном согласно требованию. Эта функция широко используется в ускоренных вычислениях.

ПЛИС с комбинированной архитектурой FLEX

Стремление сочетать достоинства CPLD и FPGA привели к появлению микросхем с комбинированной архитектурой.

Микросхемы семейства FLEX имеют ФБ LAB’S (logic area blocks) с логическими элементами, содержащие ФП табличного типа LUT’S.

ФБ расположены в виде матрицы. Между их строками столбцами проходят горизонтальные и вертикальные трассировочные каналы.

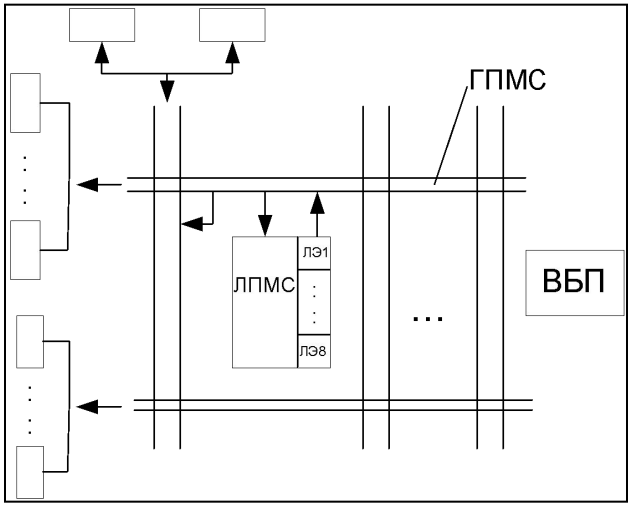
В то же время трассы в каналах не сегментированы, а непрерывны (как в CPLD).

Поскольку имеется большое количество ФБ, то реализация единой системы коммутации затруднена, поэтому она имеет два уровня межсоединений:

- Локальный
- Глобальный

Локальная программируемая матрица соединений обеспечивает межсоединение логических элементов, из которых соединяются ФБ.

В состав LAB’S входят 8 логических элементов, а соединение между ФБ обеспечивается глобальной программируемой матицей соединений, к концам которой подключаются БВВ.



В составе многих микросхем имеются встроенные блоки памяти, емкость которых может быть от 2 и более Кбит.

Путем конфигурирования встроенные блоки памяти могут быть:

2048x1

1024x2

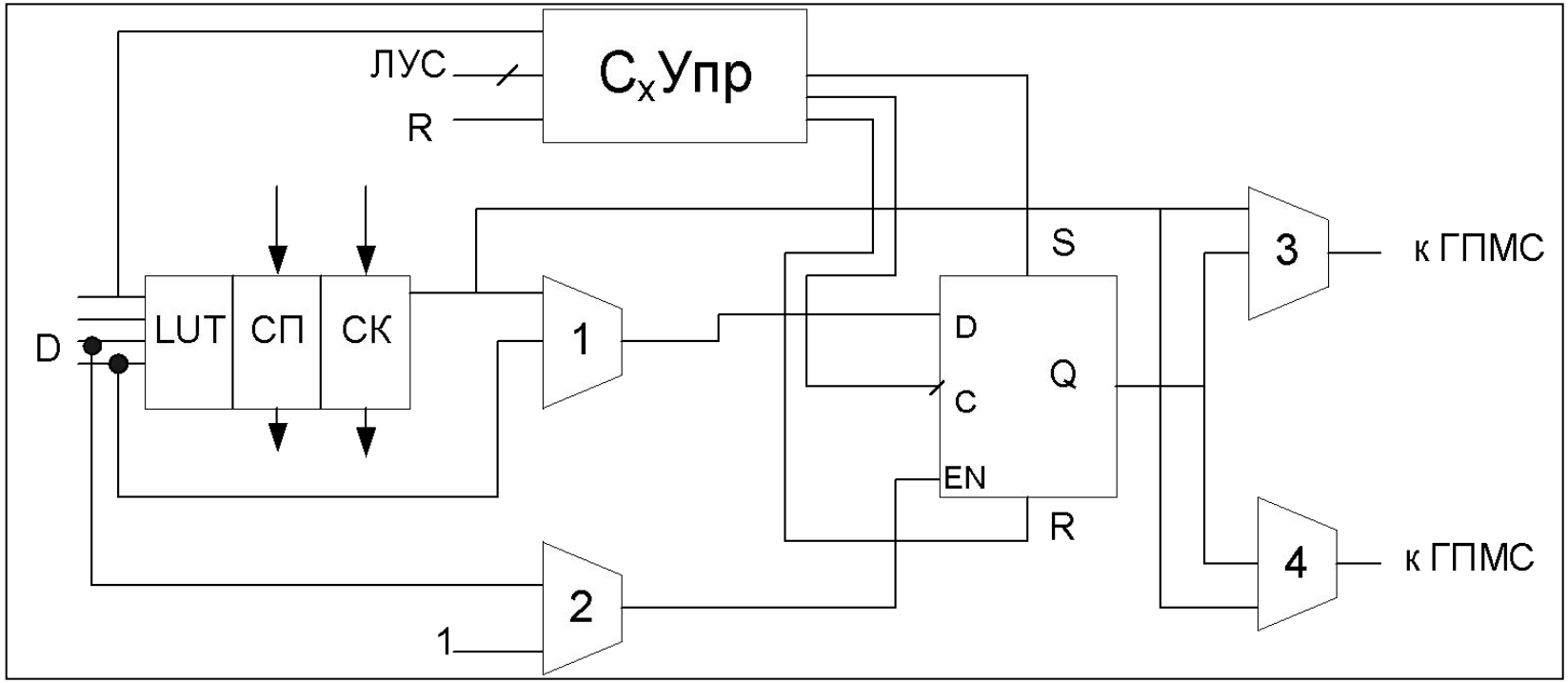
512x4

256x8

Логические элементы, которые являются основой LAB’S имеют в своей основе четырехвходовые ФП табличного типа LUT.

Особенностью схем, которые могут быть построены из этих логических элементов, является наличие специальных трактов переноса и каскадирования.

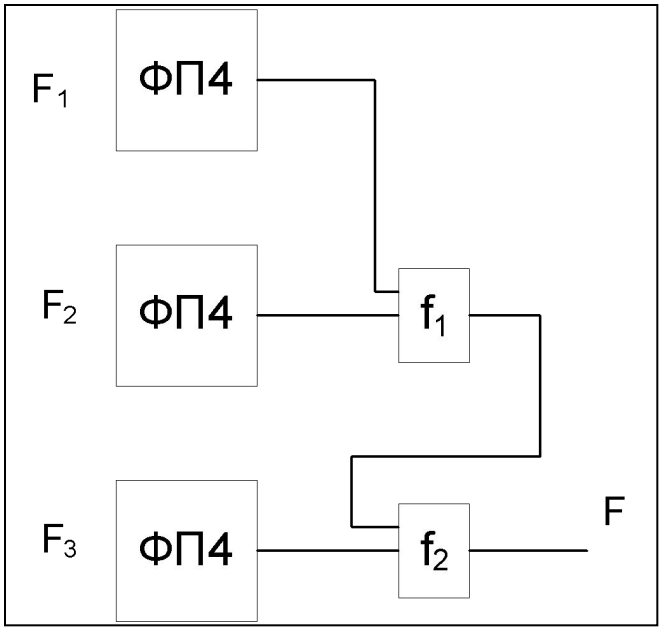
Рассмотрим упрощенную структуру такого логического элемента:



Цепи каскадирования используются для получения функций с числом аргументов больше четырех.

При этом функции многих переменных могут быть получены двумя способами:

1. F=f2(F3f1(F1F2))



2. Путем организации обратных связей с выходов ФП через локальную программируемую матрицу соединений.