Программируемые логические интегральные схемы (ПЛИС) отличаются тем, что, позволяют создавать произвольную архитектуру из базовых логических элементов. Это позволяет сочетать высокое быстродействие с гибкостью микросхемы, что даёт возможность, используя один чип реализовать целый ряд проектов.

Обобщённо внутреннее устройство ПЛИС можно разделить на три основных группы:

* массив из логических элементов (макроячеек, логических блоков);
* блоки входа-выхода (IO);
* линии связи между ними и устройство, управляющее этими связями.

Программирование ПЛИС осуществляется соединением элементов подобно тому, как бы это делается, при реализации устройства (его схемы) из отдельных логических элементов, соединяя их входы и выходы проводниками.

При выборе ПЛИС, для реализации некоторого устройства, основным критерием является достаточность количества программируемых блоков. В зависимости от конкретного ПЛИС количество блоков может изменяться в широких пределах.

**Классификация ПЛИС**

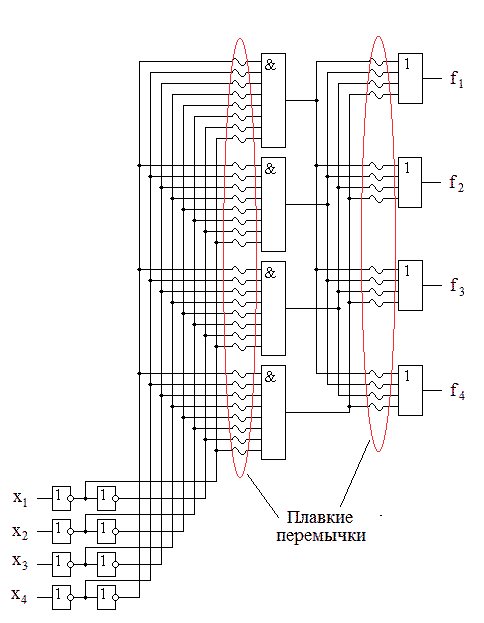
* **[](https://ru.bmstu.wiki/%D0%A4%D0%B0%D0%B9%D0%BB:%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_%D0%9F%D0%9B%D0%9C.gif)PLA** (англ. Programmable Logic Arrays) — программируемая логическая матрица. Основная идея работы ПЛМ заключается в реализации логической функции, представленной в СДНФ — совершенной дизъюнктивной нормальной форме. В программируемой логической матрице обе матрицы логических элементов "И" и "ИЛИ" программируемы.

Рис. 1. Обобщенная структура программируемых логических матриц (ПЛМ)

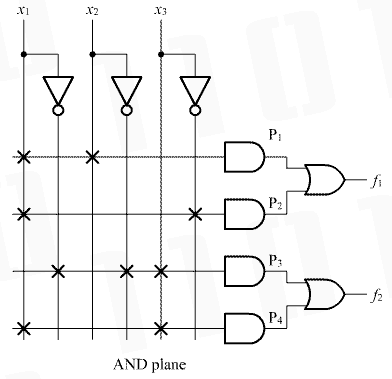
* **[](https://ru.bmstu.wiki/%D0%A4%D0%B0%D0%B9%D0%BB:%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_%D0%9F%D0%9C%D0%9B.gif)PAL** (англ. Programmable Array Logic) — программируемая матрица логики. В отличии от ПЛМ в ПМЛ матрица логических элементов "И" является программируемой, а матрица логических элементов "ИЛИ" фиксированной. Поскольку изготовить такие устройства проще, чем ПЛМ, они имеют меньшую стоимость и обладают улучшенными характеристиками, что привело к их высокой популярности.

Рис. 2. Обобщенная структура программируемых матриц логики (ПМЛ)

Наиболее актуальными на сегодняшний день являются два основных вида ПЛИС:

* **CPLD** (англ. Complex Programmable Logic Device – Программируемая Логическая Интегральная Микросхема, собственно это и есть ПЛИС в её классическом понимании). В ней обычно есть встроенная энергонезависимая память, в которую загружается прошивка.

CPLD используется в основном в схемах, где нужна высокая скорость и большое число выводов, при этом CPLD выполняют несложные задачи.

Внутренняя структура CPLD строится на матрице макроячеек или логических блоков, расположенных на одном кристалле. Количество этих макроячеек лежит в пределах сотен и тысяч штук. Каждая макроячейка соединена с блоками ввода-вывода, осуществляющими формирование необходимого вида входов или выходов для работы с внешними схемами. Кроме того, все макроячейки и блоки ввода-вывода связаны между собой внутренними параллельными шинами. Приведенная на рис. 3 упрощенная архитектура CPLD состоит из четырех макроячеек, которые связаны между собой внутренними шинами и соединяются с блоками ввода-вывода. Макроячейка построена подобно микросхеме ПЛМ, к которой на выходе подключен D-триггер, пример макроячейки приведен на рис.4. К недостаткам можно отнести то, что трудно обеспечить эффективное применение всех макроячеек. Всегда часть макроячеек остается неиспользуемой. Часто из макроячейки используется только триггер или логический элемент "2И" ("2ИЛИ"). Остальная часть схемы зря занимает площадь кристалла и потребляет ток от источника питания.

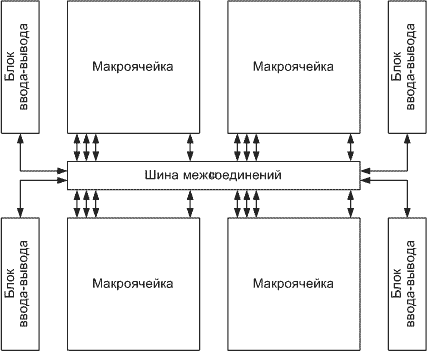
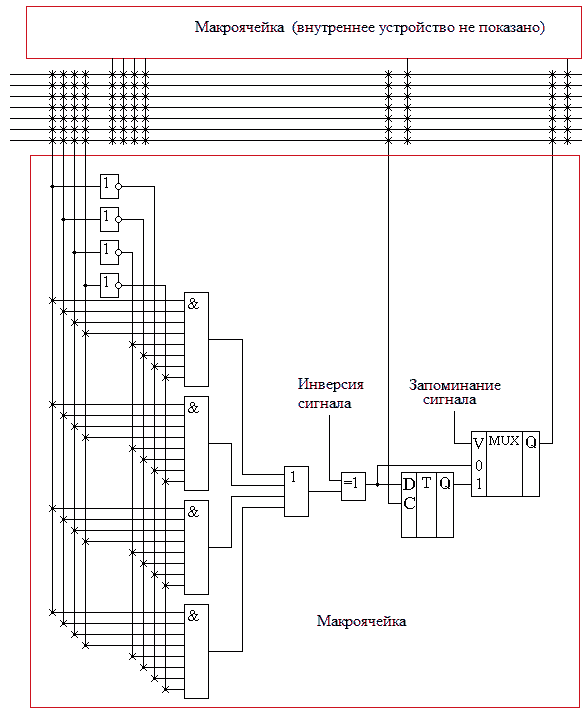


Рис. 3 Архитектура CPLD Рис. 4 Схема макроячейки

Макроячейка состоит из программируемых мультиплексоров, триггеров (одного или нескольких) и формирует группу выходных сигналов ФБ в нескольких их вариантах.

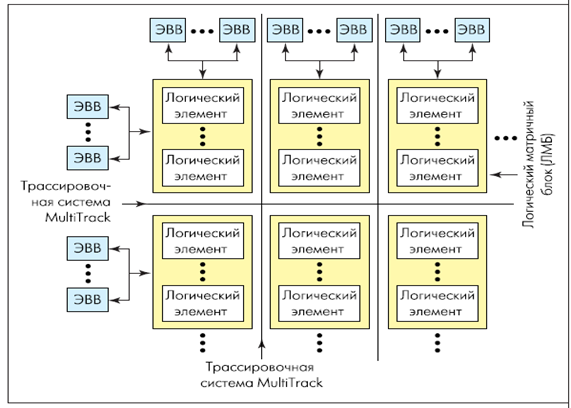
На рис. 5 приведен пример – блок-схема CPLD микросхемы семейства MAX II фирмы Altera.

Рис. 5. CPLD семейства MAX II фирмы Altera.

На рис. 6 показан структурный план микросхем этого же семейства.

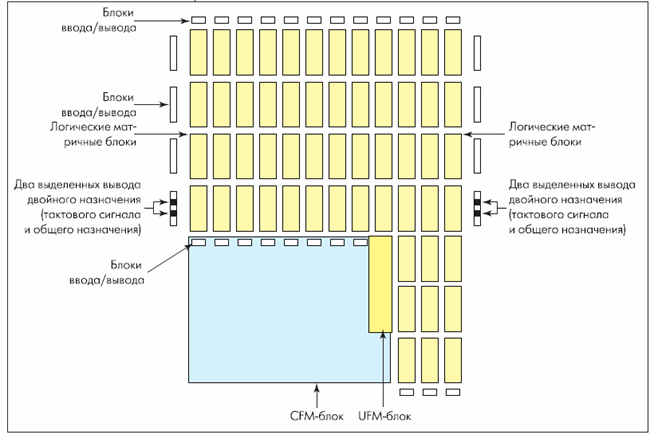


Рис. 6. структурный план CPLD семейства MAX II.

* **FPGA** (англ. Field-Programmable Gate Array). Принцип работы FPGA существенно отличаются от принципа работы CPLD. FPGA является наследником комбинационных схем, реализованных на постоянных запоминающих устройствах (ПЗУ). Обобщенная структура микросхем FPGA приведена на рис. 7.

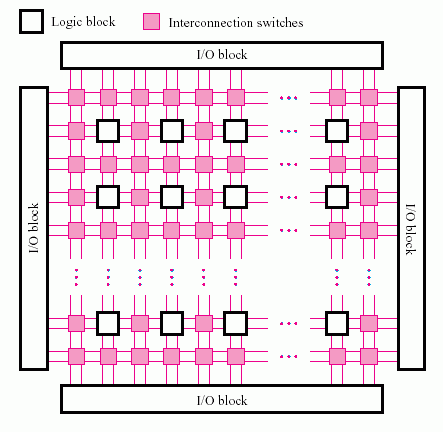
[](https://ru.bmstu.wiki/%D0%A4%D0%B0%D0%B9%D0%BB:%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_FPGA.png)

Рис. 7. Обобщенная структура микросхем FPGA

Главным отличием между большими CPLD и малыми FPGA до недавних пор было наличие внутренней [энергонезависимой](https://ru.wikipedia.org/wiki/%D0%AD%D0%BD%D0%B5%D1%80%D0%B3%D0%BE%D0%BD%D0%B5%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) конфигурационной памяти в CPLD. Это отличие становится уже не столь значимым, поскольку ряд последних моделей FPGA также включают такую внутреннюю память. Тем не менее, наличие такой внутренней энергонезависимой конфигурационной памяти, наряду с такой важной характеристой, как устойчивость показателей, делают CPLD незаменимыми для современных цифровых схем в качестве устройства для [инициализации](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D0%B8%D1%86%D0%B8%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) схемы, перед тем, как передать управление другим микросхемам, не обладающим такой способностью. В качестве примера можно привести использование CPLD для загрузки данных конфигурации FPGA из энергонезависимой памяти.

По мере развития технологии, различия между CPLD и FPGA продолжают размываться (например, иногда некоторые CPLD фирмы [Intel](https://ru.wikipedia.org/wiki/Intel) называют FPGA). В то же время, сравнивая эти два семейства и учитывая развитие самих CPLD и их возможностей, такие их архитектурные преимущества, как цена, энергонезависимая конфигурация, макроячейки с предсказуемыми характеристиками параметров, меньшее энергопотребление, можно предположить, что CPLD будут иметь в обозримом будущем устойчивую нишу в задании начальных параметров цифровых схем, мобильной технологии, расширении числа входов/выходов для более сложных микросхем, предобработке сигналов и в других применениях.