

# ПЛИС

## СОДЕРЖАНИЕ

НАЗНАЧЕНИЕ

АРХИТЕКТУРА

ТИПЫ

ПРОГРАММИРОВАНИЕ

СИСТЕМА НА КРИСТАЛЛЕ

ПРОИЗВОДИТЕЛИ

# ПЛИС

## НАЗНАЧЕНИЕ

**Программируемая логическая интегральная схема** (Programmable Logic Device, PLD, ПЛИС) — электронный компонент (интегральная микросхема), используемый для создания конфигурируемых цифровых электронных схем. В отличие от обычных цифровых микросхем, логика работы ПЛИС не определяется при изготовлении, а задаётся посредством программирования (проектирования). Для программирования используются программатор и IDE (отладочная среда), позволяющие задать желаемую структуру цифрового устройства в виде принципиальной электрической схемы или программы на специальных языках описания аппаратуры (Verilog, VHDL, AHDL и других).

### Внутренняя логическая структура (связи между транзисторами, архитектура)

ПЛИС	Микроконтроллер
<p>Задается конечным пользователем путем программирования / конфигурирования внутренних (программируемых) блоков.</p> <p>Можно получить микросхему, выполняющую на аппаратном уровне нужные действия (например, ШИМ, АЦП, ЦАП, сетевой драйвер и более сложные, в том числе и любой микроконтроллер).</p> <p>Главное, чтобы для реализуемой задачи хватило программируемых блоков. Число блоков зависит от модели ПЛИС и бывает достаточно большим.</p>	<p>Жестко определена на заводе производителе.</p>

### С точки зрения программиста

ПЛИС	Микроконтроллер
<p>Как «стадо муравьев», каждый из которых примитивен и может выполнять какие-то однообразные действия (часть большой задачи) различной сложности. Для выполнения общей задачи «муравьи» группируются. Таким образом, общая задача будет выполняться не последовательно, а параллельно — эффективно и быстро. Имея достаточное количество групп «муравьев» можно выполнять параллельно несколько общих задач.</p> <p>Например, группа из 10 «муравьев» выполняет задачу - срезают колос пшеницы и выковыривают зерно за одну минуту. Имея достаточное количество «муравьев» (объем внутренних блоков ПЛИС), можно убрать целое поле пшеницы за ту же минуту.</p>	<p>Как «человек».</p> <p>Может выполнять только одну задачу в один момент времени.</p> <p>При наличии нескольких задач, выполняется переключение между ними: определенное время работает часть одной задачи, далее исполняемая задача прерывается и работа передается на определенное время другой задаче и так далее. Переключение между задачами выполняет некая «ведущая» задача — планировщик. Исполнение задачи также может быть приостановлено по прерыванию.</p>

# ПЛИС

## НАЗНАЧЕНИЕ

### С точки зрения программиста (продолжение)

ПЛИС	Микроконтроллер
<p>Например, помогать светодиодом или несколькими светодиодами (с разными и некратными частотами): в ПЛИС создается блок управления светодиодом, созданные блоки копируются в количестве требуемых светодиодов и каждый из них настраивается под требуемые частоты и выводы. В итоге все эти блоки будут работать параллельно, никак не мешая друг другу, не снижая производительности.</p>	<p>Быстродействие здесь может быть достигнуто за счет повышения тактовой частоты процессора или передачи исполнения некоторых задач другим контроллерам или ядрам.</p> <p>В тоже время, может производить сложные действия (например, сложные вычисления), поддерживать работу с числами с плавающей точкой и пользоваться различной периферией.</p> <p>Но в любом случае, в один момент времени выполняется часть какой-то одной задачи и в некоторых случаях бывает довольно сложно получить требуемое быстродействие и качество.</p> <p>Например, помогать одним светодиодом на микроконтроллере легко (вкл.-пауза-выкл.). Попробуем добавить второй светодиод, который должен мигать с другой частотой, да еще не кратной первой. Задача резко усложняется вплоть до снижения общей производительности.</p>

На ПЛИС можно реализовать различные по сложности и возможности цифровые устройства, которые сложно сделать на микроконтроллере:

- устройства с большим количеством портов ввода-вывода (контроллер ШИМ, АЦП, ЦАП)
- устройства аудио и видео (аудио и видео контроллеры)
- устройства, выполняющие цифровую обработку сигнала
- устройства, выполняющих передачу данных (контроллер USB, PCI-шины, высокоскоростной Ethernet)
- устройства, выполняющие криптографические операции, систем защиты информации
- устройства, выполняющие роль мостов (коммутаторов) между системами с различной логикой и напряжением питания

# ПЛИС

## АРХИТЕКТУРА

ПЛИС могут быть построены на основе разных встроенных (элементарных) элементов:

- логические блоки (logical cell, logical element, LE)
- макроячейки (macrocell)

Поэтому, разные ПЛИС не всегда можно сравнивать по «объему» логических блоков.

Логические блоки (logical cells, logical elements, LE) обычно состоят из таблицы поиска на 4 входа и триггера, но бывают и с 6 входами, для увеличения производительности.

Макроячейки (macrocells) представляет собой небольшую схему из некоторого количества вентилей. Схема макроячеек может отличаться у разных производителей.

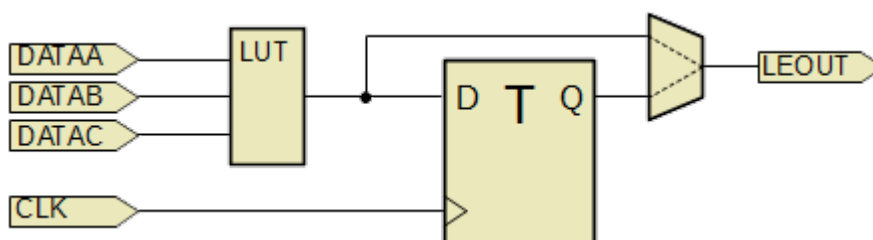
$$1 \text{ macrocell} = 1.3 \text{ LE}$$

Элементарный элемент представляют собой блок с несколькими входами и выходами — логический вентиль (gate), реализующий какую-то элементарную логическую операцию: AND, NAND, OR, NOR, XOR или (обычно) более сложную. Все вентили соединены между собой. Логика работы схемы определяется путем изменения (программирования) коммутаций между определенными вентилями. Коммутация выполняется битовыми ключами. Логика работы схемы может быть модифицирована практически в любой момент времени в процессе работы схемы. Достигается это за счет того, что вся микросхема пронизана ячейками конфигурационной памяти (STATIC RAM). Каждый бит этой памяти либо управляет каким-то ключом коммутации, либо является аргументом какой-то логической функции.

Элементарные элементы довольно эффективны. Несколько десятков блоков хватит для решения задачи малой сложности. На 700 блоках можно реализовать полноценный микроконтроллер, а на 3000 это уже будет микроконтроллер с конвейером инструкций, кэш-памятью, блоком управления памятью и производительностью в пару сотен DMIPS.

### **Базовый логический элемент**

В ПЛИС в качестве простейшего логического элемента используется сложная структура:



Элемент имеет три логических входа DATAA, DATAB и DATAC, вход тактовых импульсов CLK и один выход LEOUT. В случае если от ячейки требуется работа в качестве только комбинационного устройства, то выходной мультиплексор коммутирует выход элемента LUT на выход всей ячейки, если выход должен быть регистровым, то сигнал с LUT защелкивается по сигналу синхронизации в D-триггер, выход которого через мультиплексор соединяется с LEOUT. Управляющий вход мультиплексора (на рисунке не показан) подключен к соответствующему биту конфигурационной памяти.

# ПЛИС

## ТИПЫ

Есть разные типы ПЛИС.

Наиболее актуальные из них CPLD и FPGA.

**CPLD**, собственно и есть ПЛИС (Complex Programmable Logic Device, Программируемая Логическая Интегральная Схема). Содержат встроенную энергонезависимую память, которая хранит прошивку при отключении питания. Строятся традиционно на матрице макроячеек, хотя сейчас всё популярнее основанные на логических блоках. Обычно имеют сравнительно небольшое число элементов (сотни или тысячи). Они дешевле, проще в применении, требуют минимальной обвязки, быстро запускаются. Могут использоваться там, где не требуется сложная логика работы, но не хватает скорости или выводов у обычного микроконтроллера. Например, для обработки сигналов, к примеру, как контроллер USB, VGA или PCI-шины.

**FPGA** (Field-Programmable Gate Array, Программируемая Пользователем Вентильная Матрица, ППВМ) — дальнейшее развитие ПЛИС. Построены на логических блоках с гибкой коммутацией. Могут иметь большое число элементов (десятки и сотни тысяч). Как правило требуют внешнюю энергонезависимую память для хранения прошивки, с которой «загружается» при включении питания. Часто содержат, помимо логики, отдельные блоки, например, для умножения. FPGA дороже, сложнее, зато на них можно реализовать, например, полноценный процессор или сложную обработку сигналов. Можно проверить сложную микросхему, прежде чем начать производить её в кремнии.

# ПЛИС

## ПРОГРАММИРОВАНИЕ

Программирование ПЛИС выполняется на языке HDL (Hardware Description Language) — язык описания аппаратуры, используемый при создании (синтезировании) микросхем, как тех, что производятся в кремнии (аппаратные), так и виртуальных (программные), что прошиваются в ПЛИС. Помимо синтеза, эти языки программирования используются для моделирования.

Наиболее популярные языки HDL:

- Verilog (Си-подобный)
- VHDL (Паскаль-подобный)

Программирование выполняется в специальных средах разработки (IDE), которые помимо использования языков описания аппаратуры позволяют «рисовать» схемы в графическом виде. Все эти способы можно комбинировать, используя модульный подход.

Написанный код можно прошить в ПЛИС и проверить, а можно воспользоваться симулятором (обычно встроен в среду разработки).

# ПЛИС

## СИСТЕМА НА КРИСТАЛЛЕ

Система на кристалле (СНК, однокристальная система; System-on-a-Chip, SoC) — электронная схема, выполняющая функции целого устройства (например, компьютера) и размещённая на одной интегральной схеме.

Если разместить все необходимые цепи на одном полупроводниковом кристалле не удаётся, применяется схема из нескольких кристаллов, помещённых в единый корпус (System-in-Package, SiP).

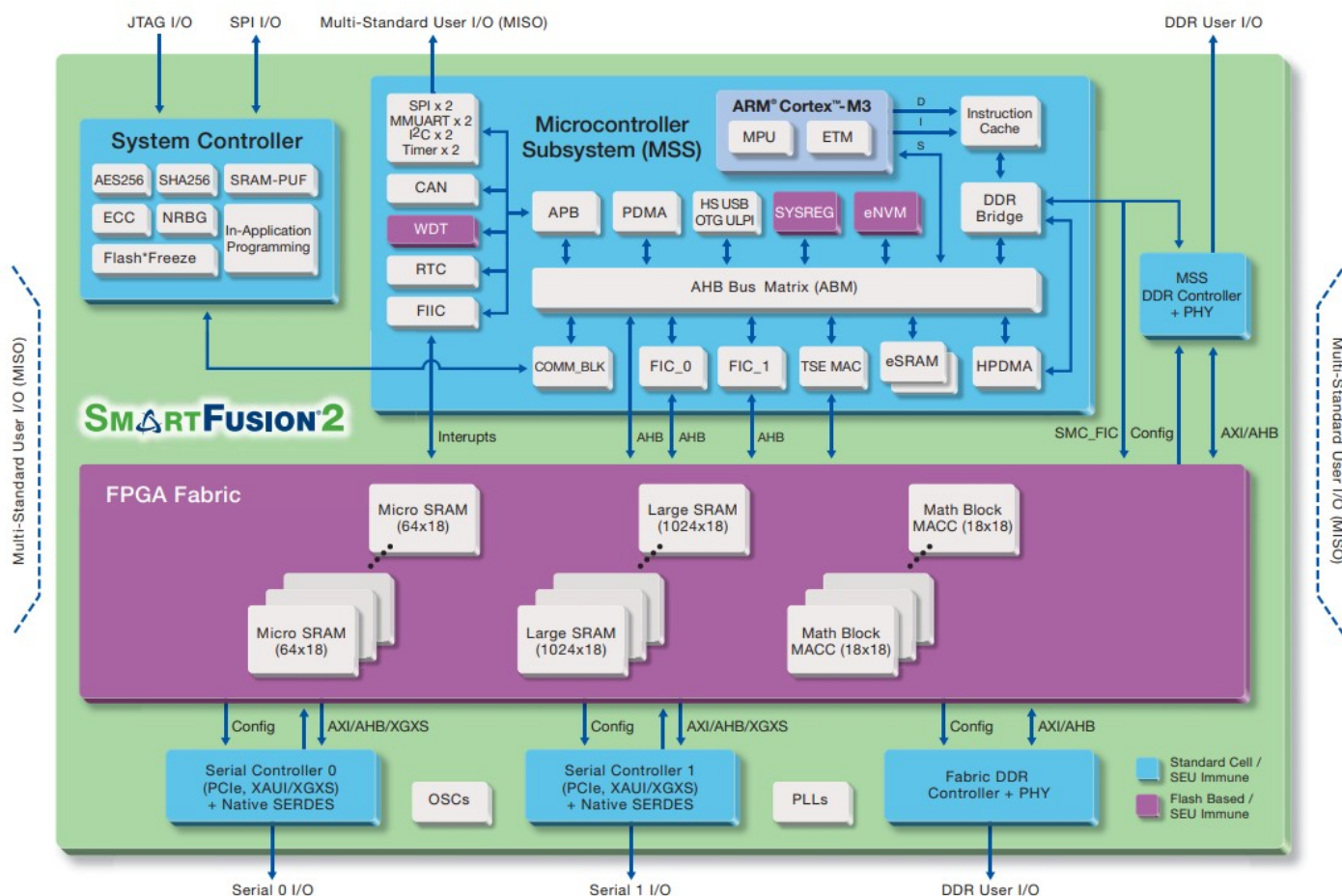
SoC считается более выгодной конструкцией, так как позволяет увеличить процент годных устройств при изготовлении и упростить конструкцию корпуса.

SoC, как правило, применяются в портативных и встраиваемых системах.

### Базовое устройство

- Микроконтроллер (одно или несколько аппаратных ядер различной архитектуры)
- ПЛИС
- Память (ОЗУ, FLASH)

Память и периферия (порты В/В, сетевые интерфейсы, таймеры, регистры), обычно, являются общими для ядер Микроконтроллера и ПЛИС. Связь между ПЛИС и Микроконтроллером, ПЛИС/Микроконтроллером и периферией осуществляется по общей шине или через прерывания.



# ПЛИС

## ПРОИЗВОДИТЕЛИ

- Xilinx
- Altera (Intel)
- Actel (Mocrochip)
- Atmel
- Lattice semiconductor
- Achronix
- TSMC

Наиболее известные производители ПЛИС: Altera и Xilinx — занимают около 80% мирового объема производства ПЛИС.