1. История развития микроэлектроники и схемотехники
2. Ламповый этап. На этом этапе эвм строились на основе ламп, которые были большими, потребляли много энергии и были не надёжными.
3. Транзисторный этап. На этом этапе эвм строились на основе транзисторов, которые были более компактные, энергоэффективные и надёжные.
4. Интегральный этап. На этом этапе были разработаны так называемые микросхемы, где на одном кристалле полупроводника может быть расположено множество транзисторов и других элементов, что привело к минитюриализации, удешевлению и повышению производительности.
5. Современный этап. Характеризуется развитием интегральных систем и развитием нанотехнологий.
6. Основные определения (сигнал, виды сигналов)

Сигнал – это физический процесс, который несёт информацию. В электронике он обычно представляет собой изменение напряжения или тока во времени.

Виды сигналов:

Аналоговый сигнал – непрерывный сигнал, значение которого находится в любом диапазоне значений (звук, температура).

Дискретный – это сигнал, который может принмать значения только определённое конечное число значений.

Цифровой – это частный случай дискретного сигнала, который обычно принмает только 2 устойчивых состояния, обозначаемых как логический 0 или логическая 1.

Импульсный – это кратковременное изменение напряжения или тока.

Периодический – это сигнал, который повторяется через определённый интервал времени

1. Система условных обозначений ИМС и их конструктивное оформление.

ИМС классифицируются по технологи изготовления, по функциональному назначению и по конструктивному исполнению. В России маркировка имеет следующий формат: XXX.YYY.ZZZ. Первые буквы указывают на серию и технологию, После буквенного кода идёт порядковый номер серии и вариант исполнения, который указывает на температурный диапазон и тип корпуса.

Конструктивное оформление ИМС включает различные типы корпусов, которые выбираются в зависимости от способа монтажа, количества выводов и условий эксплуатации.

1. Основные устройства цифровых ИМС

Устройства цифровых ИМС можно разделить на разных типы.

Самые простые – это логические элементы, которые выполняют логические операции такие как И ИЛИ НЕ, а также их комбинации.

Более сложные – это триггеры, они являются элементарными ячейками памяти. Они могут запоминать своё состояние(0 или 1) и сохранять его до изменения входных данных.

Для работы с данными используются регистры – они могут временно хранить информацию или последовательно сдвигать биты.

Более сложную функцию выполняют шифраторы и дешифраторы, которые преобразовывают одни виды кодов в другие. Они часто используются в устройствах управления, где необходимо выбрать один из вариантов ответов

Отдельно можно рассказать про запоминающие устройства, которые хранят данные и программы.

1. Корпуса цифровых микросхем

Микросхемы выпускаются в разных корпусах, которые предназначены для разных задач. У них разный тип крепления, размер и кол-во контактов.

Есть DIP корпуса, которые выглядят как маленькие прямоугольники с 2 рядами металлических ножек. Такие корпуса удобно вставлять в плату руками, без пайки и тд. Такие корпуса удобны для обучения и ремонта, но сейчас они почти не используются.

Более современными микросхемами являются SMD корпуса, которые ещё называют плоскими. Монтаж у них поверхностный. Самый простой из них SOIC, который больше похож на DIP корпуса, но гораздо меньше.

Также есть корпуса BGA – это микросхемы у которых вместо ножек используются маленькие шарики припоя. Также их называют самыми компактными. Их нельзя вставить руками, только пайка. Зато их можно разместить в большом количестве.

Есть QFN – совсем плоские микросхемы без ножек. Они имеют хороший теплоотвод, а также они очень маленькие, поэтому их часто используют в телефонах и планшетах.

1. Основные характеристики логических элементов (ЛЭ)

Логические элементы – это так называемые базовые кирпичики из которых строятся процессоры, память и так далее.

Каждый логический элемент выполняет простую логическую операцию с нулями и единицами. Например: И, ИЛИ, НЕ, И-НЕ, ИСКЛ ИЛИ(выдаст 1 если на входах разные сигналы)

Логические элементы понимают сигналы по напряжению: 0 – это напряжение близкое к 0 В, 1 – более высокое напряжение.

При изменении сигнала микросхема работает почти мгновенно, но не сразу(наносекунды).

Старые ЛЭ потребляли даже в просто включённом состоянии много энергии, современные в покое почти ничего не потребляют.

У каждого элемента есть ограниченное кол-во элементов которые можно к нему подключить.

1. Инверторы, повторители и буферы

Инверторы – это самый простой логический элемент, который инвертирует сигнал, который подаётся на него (из 1 делает 0 и наоборот).

Повторитель – это как громкоговоритель, он никак не меняет смысл самого сигнала, но при этом усиливает его, чтобы он мог управлять большим количеством других элементов.

Буфер работает как повторитель, но у него есть 3 состояния, он может передавать 1, 0 и молчать.

1. Схема и назначение элементов базового логического элемента 2ИЛИ/2ИЛИ-НЕ (2И/2И-НЕ)

2И – Лампочка будет гореть только если одновременно включены 2 включателя

2И-НЕ – это тот же элемент, что и первый, но лампочка будет гореть только тогда, когда один выключатель включен, а другой выключен

2ИЛИ – лампочка горит тогда, когда включен один из выключателей или сразу оба

2ИЛИ-НЕ – горит только тогда, когда оба выключателя выключены.

1. Логический элемент Исключающее ИЛИ

Логический элемент исключающее ИЛИ работает так, что лампочка будет гореть только тогда, когда включен один из выключателей, но стоит включить их обоих, как она гаснет.

1. Триггер. Классификация и основные параметры триггеров.

Триггер – это маленькая ячейка памяти, соответственно он может запомнить своё состояние и при прекращении подачи сигнала условная лампочка продолжит гореть.

RS-триггер – самый простой, есть 2 кнопки, “запомни 1” и “запомни 0” и при их нажатии соответственно он запоминает либо 1, либо 0. Нельзя нажимать 2 кнопки одновременно, иначе триггер не поймёт что ему запоминать.

D-триггер – более умный. У него есть вход данных(D) и кнопка записи (CLK). Он запоминает то, что было в D в момент нажатия на CLK.

JK-триггер – это улучшенная версия RS-триггера, которая обрабатывает все комбинации сигналов, которые поступают на него.

T-триггер – это самый простой триггер, который меняет состояние при каждом нажатии.

Параметры триггеров:

* Быстродействие
* Способ управления
* Нагрузочная способность
* Потребление энергии

1. Асинхронные RS-триггеры.

RS-триггер – самый простой, есть 2 кнопки, “запомни 1” и “запомни 0” и при их нажатии соответственно он запоминает либо 1, либо 0. Нельзя нажимать 2 кнопки одновременно,

иначе триггер не поймёт что ему запоминать.

1. Синхронные RS-триггеры

Если уходить в сравнения, то асинхронный RS-триггер – это невоспитанный ребёнок, который выхватывает конфету у родителей без разрешения(конфета – сигнал), а синхронный наоборот, ждёт команду и только тогда забирает конфету. Работает это за счёт сигнала с CLK, который и даёт разрешение на принятие сигнала.

1. D-триггер и DV-триггер

D-триггер – более умный. У него есть вход данных(D) и кнопка записи (CLK). Он запоминает то, что было в D в момент нажатия на CLK.

DV-триггер – это улучшенная версия D-триггера, у него помимо D и CLK есть флаг V, и если V=1, то данные с D запишутся, иначе нет.

1. T-триггер и триггер Шмидта

T-триггер – это самый простой триггер, который меняет состояние при каждом нажатии.

триггер Шмидта – своеобразный чистильщик сигналов, который убирает шумы и не реагирует на мелкие колебания.

1. JK-триггер

JK-триггер – это улучшенная версия RS-триггера, которая обрабатывает все комбинации сигналов, которые поступают на него.

1. Счетчики. Основные определения и классификация

Счётчики работают также как и обычные электронные счёты, которые умеют автоматически прибавлять или отнимать единичку при каждом сигнале.

Счётчики бывают суммирующими, вычитающими и реверсивными

1. Асинхронный суммирующий счетчик

Асинхронный суммирующий счётчик – это цепочка T-триггеров, где каждый последующий триггер переключается от выхода предыдущего. Он считает импульсы вверх и обнуляется при достижении максимума.

1. Асинхронный вычитающий счетчик

Это цепочка триггеров, которая уменьшает своё значение при каждом входном импульсе на 1. Когда доходит до минимума, счётчик автоматически переходит

1. **Реверсивный счетчик**

Тоже самое, но умеет считать и вверх и вниз.

1. **Синхронный суммирующий счетчик**

Синхронный суммирующий счётчик — это цифровое устройство, которое:

* **Считает вверх** (0 → 1 → 2 → ...)
* Все его внутренние триггеры переключаются **одновременно** по общему тактовому сигналу (**CLK**).
* Не имеет проблем с задержками (в отличие от асинхронных).

1. **Вычитающий синхронный счетчик**

Вычитающий синхронный счётчик — это цифровая схема, которая:

* **Считает вниз** (например, от 7→6→5→...→0→7)
* Все триггеры переключаются **одновременно** по тактовому сигналу (CLK)
* Не имеет проблем с задержками (в отличие от асинхронных аналогов)

1. **Счетчик-делитель**

**Счетчик-делитель** — это цифровое устройство, которое уменьшает частоту входного сигнала в заданное число раз. Он является разновидностью **счетчика импульсов** и используется в цифровой электронике для управления частотой синхросигналов, временными задержками и другими задачами, где требуется деление частоты.

1. **Регистры. Основные определения, классификация**

Регистры – это элементы цифровой электроники, которые предназначены для временного хранения, преобразования и передачи двоичных данных. Обычно они состоят из триггеров(обчно это D-триггеры, либо JK-триггеры) и могут включать дополнительные схемы для управления вводом и выводом информации

Классификация регистров:

1. По способу загрузки и вывода данных
2. По направлению сдвига
3. По назначению
4. **Регистры хранения**

Регистр хранения – это цифровое устройство, которое предназначено для временного хранения двоичных данных. Регистр хранения не выполняет преобразование или перемещение данных, а просто хранит их до момента обновления.

1. **Регистры сдвига**

Регистр сдвига – это цифровое устройство, которое предназначено для последовательного ввода/вывода данных с возможностью их хранения и побитого сдвига. В отличии от регистра хранения он не просто сохраняет информацию, а ещё и сдвигает её вправо или влево

1. **Реверсивные регистры**

Это универсальный сдвиговый регистр, который даёт возможность выбирать направление сдвига данных. Он сочетает функции обычного сдвигового регистра и регистра хранения, что делает его особенно полезным в сложных цифровых схемах.

1. **Кольцевой регистр сдвига**

Это замкнутый регистр сдвига, в котором выход последнего разряда подключен ко входу первого, что образует замкнутую петлю и позволяет бесконечно циркулировать данным пока подаётся тактовый сигнал.

1. **Универсальные регистры**

Универсальные регистры – это программируемые цифровые устройства, сочетающие функции регистра хранения, сдвигового регистра и реверсивного регистра в одном корпусе. Они являются наиболее гибким решением для сложных цифровых систем.

1. **Мультиплексор**

Мультиплексор – это электронное устройство, которое принимает один из нескольких входных сигналов и передаёт его на общий выход. Его основной задачей является переключение между несколькими источниками данных в зависимости от управляющего сигнала.

У мультиплексора есть n входов данных, m управляющих входов (S0, S1 и тд. – селекторы) 1 выход (Y).

1. **Демультиплексор**

Демультиплексор – это устройство обратное мультиплексору. У него может быть не один выход и один вход. В зависимости от управляющего сигнала он берёт данные со входа и перенаправляет их на один из выходов.

1. **Шифратор**

Шифратор – это электронное устройство, которое преобразует сигнал одного из множества входов в двоичный код на выходе.

Для примера можно взять шифратор у которого 8 входов и, соответственно, 3 выхода(так как больше не надо для шифрования 8 символов) и он преобразует каждый из символов в двоичный код: D0 – 000; D1 – 001; D2 – 010; и тд.

1. **Дешифратор**

Дешифратор работает немного по-другому, нежели шифратор. Если шифратор кодирует входные сигналы, то дешифратор расшифровывает то, что подаётся на входах.

1. **Преобразователь кодов**

Преобразователь кодов – это электронное устройство, которое преобразует один код в другой на выходе(преобразование происходит по заданным правилам).

1. **Полусумматор**

Это простейшее электронное устройство, которое выполняет сложение двух однобитных чисел(0 или 1) без учёта переноса из предыдущего разряда.

1. **Полный сумматор**

**Полный сумматор** — это логическое устройство, которое **складывает три однобитных числа**: два основных бита (**A** и **B**) и **перенос из предыдущего разряда** (**Cin**, то есть Carry In). В отличие от **полусумматора**, он **учитывает перенос**, поэтому используется в реальных схемах для сложения многобитных чисел.

1. **Суммирующее устройство последовательного типа**

Это устройство, которое последовательно складывает многобитные числа поразрядно, один бит за один такт, начиная с младшего разряда и учитывая перенос между разрядами.

1. **Суммирующее устройство параллельного типа**

Это электронное устройство, которое складывает сразу все разряды за счёт того, что в устройстве параллельного типа находится 4 полных сумматора, которые складывают все разряды учитывая переносы.

1. **Устройство умножения**

**Устройство умножения** — это цифровая схема, которая выполняет **умножение двух двоичных чисел** и выдаёт результат в виде двоичного произведения. Оно может быть реализовано **разными способами** — от простого «школьного» умножения до быстрых аппаратных схем.

1. **Арифметико-логическое устройство\**

**Арифметико-логическое устройство (АЛУ)** — это основная часть процессора, которая выполняет **арифметические** и **логические** операции над двоичными числами.

Если представить процессор как мозг компьютера, то **АЛУ — это его "руки"**, которые умеют:

* Считать (сложение, вычитание и т. д.)
* Проверять условия (логические операции, сравнение)
* Работать с числами по правилам двоичной математики

**Что умеет делать АЛУ:**

**Арифметические операции:**

* Сложение
* Вычитание
* Иногда — умножение и деление (в простых процессорах — через отдельные блоки)

**Логические операции:**

* **AND (И)** — 1, если оба бита 1
* **OR (ИЛИ)** — 1, если хотя бы один бит 1
* **XOR (исключающее ИЛИ)** — 1, если биты разные
* **NOT (НЕ)** — инвертирует биты

**Другие функции:**

* Сдвиги (влево, вправо)
* Сравнение чисел (меньше, больше, равно)
* Установка флагов (например, флаг нуля, переполнения, переноса)

1. **Компараторы**

**Компаратор** — это цифровое устройство, которое **сравнивает два числа** и сообщает, **какое из них больше, меньше или равно другому**.

Он проверяет:

* A = B → числа равны
* A > B → первое число больше
* A < B → первое число меньше

**Какие бывают компараторы:**

**1. Однобитный компаратор:**

* Сравнивает по одному разряду (биту)
* Работает по простым логическим правилам

**2. Многобитный компаратор:**

* Сравнивает целые двоичные числа (например, 4-битные или 8-битные)
* Последовательно сравнивает каждый разряд, начиная со старшего