

1. Аналоговые и цифровые сигналы

Основы цифровой схемотехники

Цифровая схемотехника – важнейшая дисциплина, которую изучают во всех высших и средних учебных заведениях, готовящих специалистов в электронике. Большинство книг и учебных пособий написаны очень сложным для понимания языком, и начинающему электронщику (возможно, школьнику) будет тяжело освоить новую информацию. Постараемся о сложных понятиях рассказать доступным образом.

1.1. Аналоговые и цифровые сигналы

Сначала надо разобраться, чем вообще аналоговая схемотехника отличается от цифровой. И главное отличие – в сигналах, с которыми работают эти схемы. Все сигналы можно разделить на два основных вида: аналоговые и цифровые.

Аналоговые сигналы

Аналоговые сигналы наиболее привычны для нас. Можно сказать, что весь окружающий природный мир вокруг нас – аналоговый. Наши зрение и слух, а также все остальные органы чувств воспринимают поступающую информацию в аналоговой форме, то есть непрерывно во времени. Передача звуковой информации – речь человека, звуки музыкальных инструментов, рёв животных, звуки природы и т.п. – также осуществляется в аналоговом виде.

Чтобы лучше понять этот вопрос, изобразим аналоговый сигнал (см. рис.1.):



Рис.1. Аналоговый сигнал

Мы видим, что аналоговый сигнал непрерывен во времени и по амплитуде. Для любого момента времени можно определить точное значение амплитуды аналогового сигнала.

Цифровые сигналы

Будем анализировать амплитуду сигнала не постоянно, а дискретно, через фиксированные промежутки времени. Например, раз в секунду, или чаще: десять раз в секунду. То, как

часто мы будем это делать, называется частотой дискретизации: один раз в секунду – 1 Гц, тысячу раз в секунду – 1000 Гц или 1 кГц.

Для наглядности нарисуем графики аналогового (вверху) и цифрового (внизу) сигналов (рис.2.):

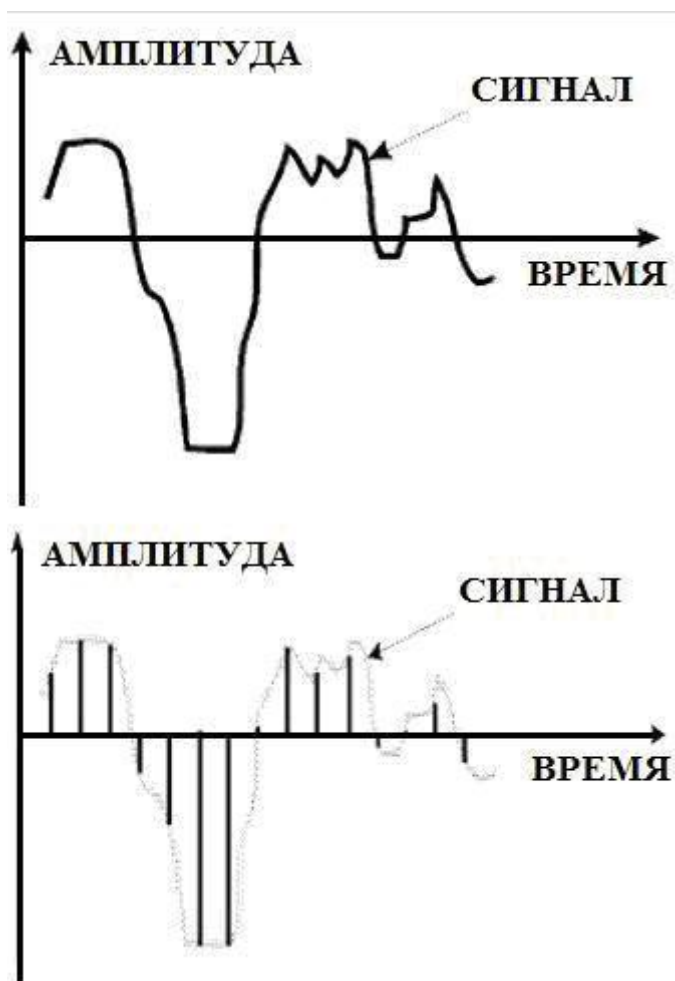


Рис.2. Аналоговый сигнал (вверху) и его цифровая копия (внизу)

Мы видим, что в каждый мгновенный промежуток времени можно узнать мгновенное цифровое значение амплитуды сигнала. Что происходит с сигналом (по какому закону он меняется, какова его амплитуда) между интервалами «проверки», мы не знаем, эта информация потеряна для нас. Чем реже мы проверяем уровень сигнала (чем ниже частота дискретизации), тем меньше имеем информации о сигнале. Разумеется, справедливо и обратное: чем выше частота дискретизации, тем лучше качество представления сигнала. В пределе, увеличивая частоту дискретизации до бесконечности, мы получаем практически тот же аналоговый сигнал.

Значит ли это, что аналоговый сигнал в любом случае качественнее цифрового? В теории, пожалуй, да. Но на практике современные аналого-цифровые преобразователи (АЦП) работают с такой высокой частотой дискретизации (до нескольких миллионов выборок в секунду), так качественно описывают аналоговый сигнал в цифровой форме, что органы чувств человека (глаза, уши) уже не могут почувствовать разницу между оригинальным сигналом и его цифровой моделью. Цифровой сигнал обладает очень существенным

достоинством: его легче передавать по проводам или радиоволне, помехи не оказывают на такой сигнал существенного влияния. Поэтому вся современная мобильная связь, теле- и радиовещание - цифровая.

Нижний график на рис. 2 легко представить и в другом виде – как длинную последовательность пары цифр: время/амплитуда. А цифры – это как раз то, что нужно цифровым схемам. Правда, цифровые схемы предпочитают работать с цифрами в особом представлении, но об этом мы поговорим в следующем уроке.

Сделаем важные выводы:

- цифровой сигнал дискретен, его можно определить только для отдельных моментов времени;
- чем выше частота дискретизации – тем лучше точность представления цифрового сигнала.

2. Двоичная логика

Логические уровни и двоичная система счисления

Мы познакомились с цифровыми сигналами. Теперь мы знаем, как можно преобразовать привычный нам аналоговый сигнал в цифровой в виде последовательности чисел.

Но оказывается, цифровые схемы предпочитают работать не с любыми числами, а только с «0» либо «1». Рассмотрим процесс перевода произвольного числа в двоичный код.

Начнём с определения понятия «логический уровень».

Цифровые схемы понимают только два состояния: «нет сигнала» или «есть сигнал». Другими словами, состояние «нет сигнала» - это низкий логический уровень, или логический «0», состояние «есть сигнал» - высокий логический уровень, или логическая «1». Условились считать, что в электронике логической единице соответствует сигнал высокого напряжения, а логическому нулю – сигнал низкого напряжения. На графике (рис.1.) это можно представить так:

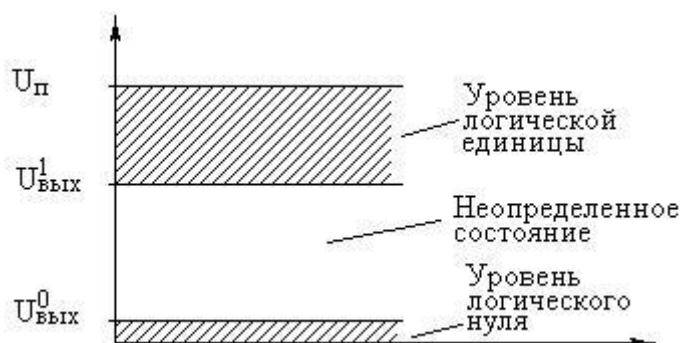


Рис.1. Высокий и низкий логический уровни

Обозначения на рисунке: $U_{п}$ – это уровень напряжения питания; $U_{1вых}$ - уровень логической единицы, $U_{0вых}$ – уровень логического нуля.

Мы видим, что сигнал логической «1» может иметь напряжение не строго равное напряжению питания ($U_{п}$) или 0В, а находящееся в некотором диапазоне, ширина которого ($U_{п} - U_{1вых}$). Так условились специально по причине того, что КПД микросхем не может быть 100% (из-за потерь на рассеивание тепла напряжение на выходе микросхемы всегда ниже её напряжения питания).

Эти же рассуждения справедливы и для уровня логического «0», который принимается равным чуть выше 0 Вольт.

Для разных микросхем ширина интервала логических «1» и «0» разная, поэтому на этом графике намеренно не указаны конкретные цифровые значения напряжений: для первого знакомства с цифровой схемотехникой этого не требуется.

А теперь поговорим о том, как преобразовывать любые числа в двоичный код, то есть в последовательность «нулей» и «единиц».

Есть простое правило перевода: для того, чтобы преобразовать десятичное число в двоичное, его нужно разложить по степеням двойки.

Возьмём, например, десятичное число 9 и разложим его по степеням двойки (здесь хорошо бы знать степени двойки на память):

$$9 = 8 + 1 = 2^3 + 2^0 = 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 1001_2$$

Последовательность нулей и единиц в правой части уравнения – это и есть двоичный код нашей девятки. Таким образом: $9_{10} = 1001_2$

Можете потренироваться сами, а для проверки приведём таблицу соответствия десятичных и двоичных чисел от нуля до десяти (рис.2.):

Десятичное число	Двоичный код
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010

Рис.2. Таблица соответствия десятичных чисел и двоичного кода

Таким же образом можно перевести любое десятичное число в двоичный вид. Например, число 245 в двоичной системе будет записано так: 11110101.

Процесс перевода чисел из одной системы в другую — это рутинная механическая работа, поэтому можно воспользоваться специальными калькуляторами, переводящими числа из одной системы счисления в другую.

Конечно, мы рассмотрели двоичную систему счисления только в самом общем виде, но для первого понимания основ этой информации достаточно. В следующих уроках мы продолжим погружение в увлекательный мир цифровой схемотехники и поговорим о простейших логических элементах.

3. Логические элементы

Логические элементы

Абсолютно все электронные компоненты, обрабатывающие цифровые сигналы, состоят из небольшого набора одинаковых «кирпичиков». В микросхемах малой степени интеграции могут быть единицы и десятки таких элементов, а в современных процессорах их может быть очень и очень много. Они называются логические элементы.

Логическим элементом называется электрическая схема, предназначенная для выполнения какой-либо логической операции с входными данными.

Логический элемент — элемент, осуществляющий определенную логическую операцию над входными сигналами. Входные данные представляются здесь в виде напряжений различных уровней, и результат логической операции на выходе — также получается в виде напряжения определенного уровня. Логические элементы обычно используются для построения логических схем вычислительных машин, дискретных схем автоматического контроля и управления.

Тем не менее, принцип работы цифровой логики остается неизменным — на входе логического элемента (входов может быть несколько) должен быть цифровой сигнал (сигналы, если входов несколько), который однозначно определяет сигнал на выходе логического элемента.

Конечно, логические элементы строятся, в свою очередь, из уже рассмотренных в предыдущих уроках резисторов, транзисторов и других электронных компонентов, но с точки зрения разработки цифровых схем именно логический элемент является их «элементарной» частицей.

При анализе работы логических элементов используется так называемая булева алгебра. Начала этого раздела математики было изложено в работах Джорджа Буля — английского математика и логика 19-го века, одного из основателей математической логики. Основами булевой алгебры являются высказывания, логические операции, а также функции и законы. Для понимания принципов работы логических элементов нет необходимости изучать все тонкости булевой алгебры, мы освоим ее основы в процессе обучения с помощью таблиц истинности.

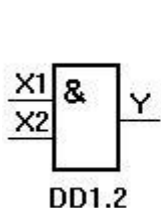
Еще несколько замечаний. Логические элементы (как, впрочем, и другие элементы электронных схем) принято обозначать так, чтобы входы были слева, а выходы справа. Число входов может быть, вообще говоря, любым, отличным от нуля. Реальные цифровые микросхемы могут иметь до 8 входов, но мы ограничимся двумя – этого достаточно для понимания. Условные обозначения соответствуют отечественному ГОСТу, в других стандартах они могут быть иными.

Какие же бывают логические элементы?

Логические элементы имеют один или несколько входов и один или два (обычно инверсных друг другу) выхода. Значения «нулей» и «единиц» выходных сигналов логических элементов определяются логической функцией, которую выполняет элемент, и значениями «нулей» и «единиц» входных сигналов, играющих роль независимых переменных. Существуют элементарные логические функции, из которых можно составить любую сложную логическую функцию.

Элемент «И» (AND), он же конъюнктор, выполняет операцию логического умножения:

Условное обозначение - Таблица истинности



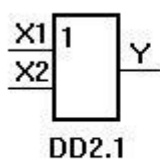
X2	X1	Y
0	0	0
0	1	0
1	0	0
1	1	1

Здесь изображен логический элемент «2И» (цифра перед буквой «И» означает число входов). Знак & (амперсант) в левом верхнем углу прямоугольника указывает, что это логический элемент «И». Первые две буквы обозначения DD1.2 указывают на то, что это цифровая микросхема (Digital), цифра слева от точки указывает номер микросхемы на принципиальной схеме, а цифра справа от точки – номер логического элемента в составе данной микросхемы. Одна микросхема может содержать несколько логических элементов.

Состояние входов в таблице обозначаются «0» и «1» («ложь» и «истина»). Из таблицы видно, что выход «Y» будет иметь состояние «1» только в том случае, когда на обоих входах «X1» и «X2» будут «1». Это легко запомнить: умножение на «0» всегда дает «0».

Элемент «ИЛИ» (OR), он же дизъюнктор, выполняет операцию логического сложения:

Условное обозначение - Таблица истинности

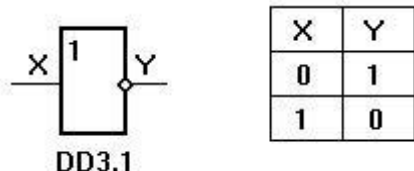


X2	X1	Y
0	0	0
0	1	1
1	0	1
1	1	1

Состояние «1» на выходе будет всегда, пока есть хотя бы одна «1» на входах.

Элемент «НЕ» (NOT), он же инвертор, выполняет операцию логического отрицания:

Условное обозначение - Таблица истинности



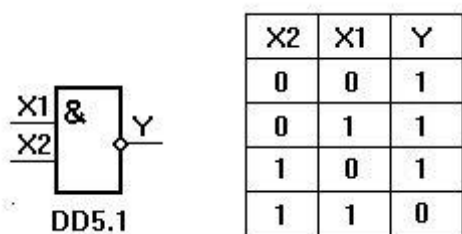
Состояние на выходе обратно состоянию на входе.

Вот из этих трех элементов строятся все цифровые устройства!

Рассмотрим еще три логических элемента, которые можно получить, комбинируя уже рассмотренные. В силу исторически сложившихся схемотехнических решений эти скомбинированные схемы тоже считаются логическими элементами.

Элемент «И-НЕ» (NAND), конъюнктор с отрицанием:

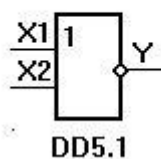
Условное обозначение - Таблица истинности



Элемент И-НЕ работает точно так же как «И», только выходной сигнал противоположен. Там где у элемента «И» на выходе должен быть «0», у элемента «И-НЕ» будет единица. И наоборот.

Элемент «ИЛИ-НЕ» (NOR), дизъюнктор с отрицанием:

Условное обозначение - Таблица истинности

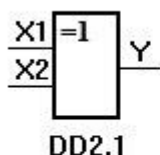


X2	X1	Y
0	0	1
0	1	0
1	0	0
1	1	0

Элемент работает так же как и «ИЛИ», но с инверсией выхода.

Элемент «Исключающее ИЛИ» (XOR), сумматор по модулю 2:

Условное обозначение - Таблица истинности



X2	X1	Y
0	0	0
0	1	1
1	0	1
1	1	0

В этом элемента «1» на выходе будет только тогда, когда на входах разные состояния.

На таких элементах строят сумматоры двоичных многоразрядных чисел. Для этого используется еще один дополнительный выход, на котором при появлении на входах двух «1» появляется сигнал переноса разряда.

Мы рассмотрели логические элементы, которые применяются в цифровой технике для построения логических схем любого уровня сложности, но рассмотренные нами элементы не могут делать одну крайне важную работу – они не умеют хранить информацию.

Для хранения используется более сложный класс устройств, называемый элементами с памятью или конечными автоматами. В этот класс входят триггеры, регистры, счетчики, шифраторы (дешифраторы), мультиплексоры (демультиплексоры) и сумматоры. Некоторые из этих устройств мы рассмотрим в следующих уроках.