Министерство образования и науки Украины

Харьковский национальный университет радиоэлектроники

Кулак Эльвира Николаевна

# 1 Основы дисциплины

Харьков

2010

## Содержание

## Теория

### 1.1 Из истории развития прикладной теории цифровых автоматов

Достижения в создании и применении средств вычислительной техники определяют не только уровень развития промышленности и организацию различных процессов управления, но и возможности фундаментальных исследований в различных отраслях науки. Именно поэтому цифровая техника стала наиболее важным фактором технического прогресса и существенной частью производительных сил в ведущих странах мира.

Перед Второй Мировой Войной (1939-1945 гг.) и 10 лет спустя, цифровые устройства были главным образом основаны на релейных схемах. Строгое доказательство того, что Булева алгебра может использоваться для анализа релейных схем, было выдвинуто советским физиком Шестаковым В. И. в 1935 г.. Американский инженер Шеннон К. Э. и японский ученый Накашима привели аналогичные доказательства в 1936-1938 г.. Шестаков В. И. показал, что релейные схемы способны моделировать функции алгебры логики. Причем истинность и ложность высказываний моделируется замкнутыми или открытыми контактами электрической цепи.

Развитие электроники и микроэлектроники стимулировало создание и развитие теории цифровых автоматов. Наиболее важным достижением 60-х годов было создание эффективной модели цифрового автомата с памятью. Модель конечного автомата стала фундаментальной частью методов синтеза в теории автоматов с памятью. Таким образом, теория релейных устройств переросла в теорию дискретных (цифровых) автоматов.

Автоматы, широко применяемые в практике, подразделяются на два класса: автоматы Мили и автоматы Мура, названные так в честь американских ученых, исследовавших впервые эти типы автоматов.

Существенное влияние на теорию цифрового автоматов оказало изобретение компьютера. Академик Глушков В. М. внес большой вклад в развитие теории автоматов, особенно в применении к ЭВМ. Многочисленные теоретические работы в области автоматического синтеза устройств ЭВМ связаны с его именем.

Практика проектирования поставила новые сложные задачи не только в области теории цифровых автоматов, но и в теории алгоритмов, теории информации и теории систем. Все это привело к дальнейшему развитию цифровых автоматов и перерастанию её в раздел технической кибернетики.

Следует отметить, что советские ученые занимают ведущие позиции в мире в области теории конечных автоматов. Так, первая монография по релейно-контактным схемам написана Гавриловым М. А., первая машина для автоматического анализа и синтеза схем разработана Пархоменко П. П. и Рогинским В. Н., логический язык для проектирования алгоритмов синтеза впервые в мире разработан Закревским А. Д. К подобным работам относятся исследования по синтезу программных автоматов Лазарева В. Г., исследования по синтезу асинхронных автоматов Якубайтиса Э. А., методы синтеза микропрограммных автоматов Баранова С. И..

## 1.2 Представление информации в цифровых устройствах

В повседневной жизни мы обычно пользуемся десятичной системой счисления. Это позиционная система счисления, которая имеет 10 арабских цифр от 0 до 9 и, следовательно, основание 10. Т. е. число 6235,89 в десятичной системе счисления представляется выражением

$$6*10^{3}+2*10^{2}+3*10^{1}+5*10^{0}+8*10^{-1}+9*10^{-2},$$

или в общем виде

$$a_3*10^{3}+a_2*10^{2}+a_1*10^{1}+a_0*10^{0}+a_{-1}*10^{-1}+a_{-2}*10^{-2},$$

где $a_i$ , - коэффициенты, принимающие значения 0, 1, 2 … 9.

Цифровые системы оперируют двумя значениями сигнала - логические 0 и 1. Следовательно, для математического представления значений дискретных переменных в цифровых системах используется двоичная система счисления - позиционная система счисления, которая имеет 2 арабских цифры 0 и 1 и, следовательно, основание 2.

Т. е. двоичное число 1001,01 в двоичной системе счисления представляется выражением

$$1*2^{3}+0*2^{2}+0*2^{1}+1*2^{0}+0*2^{-1}+1*2^{-2} \quad (1.1)$$

или в общем виде

$$a_3*2^{3}+a_2*2^{2}+a_1*2^{1}+a_0*2^{0}+a_{-1}*2^{-1}+a_{-2}*2^{-2}, \quad (1.2)$$

где $a_i$ , - коэффициенты, принимающие значения 0, 1.

Для определения десятичного эквивалента данного числа достаточно подсчитать сумму произведений в выражении (1.1).

$$1*2^3 + 0*2^2 + 0*2^1 + 1*2^0 + 0*2^{-1} + 1*2^{-2} = 8 + 0 + 0 + 1 + 0 + 0{,}25 = 9{,}25$$

Для перевода числа из десятичной системы счисления в двоичную можно воспользоваться делением десятичного числа на основание 2 и записывать остатки от деления в обратном порядке, а можно воспользоваться другим способом, вытекающим из выражения (1.2). В таблице 1.1 представлен пример перевода числа $(171{,}75)^{10}$ в двоичную систему счисления.

Таблица 1.1 - Перевод десятичного числа в двоичное по степеням двойки

| Степени двойки | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ | $2^{-1}$ | $2^{-2}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 | 0,5 | 0,25 |
| Двоичное число | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |

Способ состоит в подборе тех степеней двойки, которые в сумме дают исходное десятичное число. В позиции этих степеней подставляются единицы (голубые столбцы), остальные позиции заполняются нулями (белые столбцы). Старший разряд расположен слева, как и в других позиционных системах счисления. В нашем примере $(171{,}75)^{10} = (10101011{,}11)^2$ . Поскольку далее будут использоваться небольшие десятичные целые числа, такой способ перевода их в двоичные эквиваленты является наиболее удобным. Аналогичным образом выполняется обратное преобразование. Т. е. для перевода двоичного числа в десятичное необходимо складывать степени двойки, соответствующие позициям с единичными значениями. В нашем примере

$$(10101011{,}11)_2 = 128 + 32 + 8 + 2 + 1 + 0{,}5 + 0{,}25 = (171{,}75)_{10}.$$

При работе с цифровыми системами часто приходится иметь дело с восьмиричной и шестнадцатиричной системами счисления. Так как $8 = 2^2$ и $16 = 2^4$ , каждая восьмиричная цифра соответствует трем бинарным, а каждая шестнадцатиричная цифра соответствует четырем бинарным цифрам. Переход от двоичной к восьмиричной системе счисления выполняется разделением двоичной последовательности на группы из трех цифр (триад), начиная от запятой влево и вправо от нее, и заменой каждой из групп одной восьмиричной цифрой 0, 1, …, 7. Следующий пример иллюстрирует данное правило.

$$(\underbrace{10}_{2}\,\underbrace{110}_{6}\,\underbrace{001}_{1}\,\underbrace{101}_{5}\,\underbrace{011}_{3} \; * \; \underbrace{111}_{7}\,\underbrace{100}_{4}\,\underbrace{000}_{0}\,\underbrace{110}_{6})_2 = \left(26153.7460\right)_8$$

Конвертация двоичного кода в шестнадцатиричный аналогична, за исключением того, что двоичный код разделяется на группы из четырех единиц (тетрады), которые заменяются шестнадцатиричными знаками

$$0,\ 1,\ \ldots,\ 9,\ A,\ B,\ C,\ D,\ E,\ F.$$

Следующий пример иллюстрирует данное правило.

$$(\underbrace{10}_{2}\,\underbrace{1100}_{C}\,\underbrace{0110}_{6}\,\underbrace{1011}_{B} \; * \; \underbrace{1111}_{F}\,\underbrace{0010}_{2})_2 = (2C6B.F2)_{16}$$

Процедура получения двоичного кода из восьмиричного (шестнадцатиричного) обратна выше упомянутой. Каждая цифра восьмиричного (шестнадцатиричного) кода заменяется на двоичную триаду (тетраду). Это иллюстрирует следующий пример:

$$(673.124)_8 = (\underbrace{110}_{6}\,\underbrace{111}_{7}\,\underbrace{011}_{3} \; * \; \underbrace{001}_{1}\,\underbrace{010}_{2}\,\underbrace{100}_{4})_2$$

$$(306.D)_{16} = (\underbrace{0011}_{3}\,\underbrace{0000}_{0}\,\underbrace{0110}_{6} \; * \; \underbrace{1101}_{D})_2$$

Человеку трудно работать с двоичными числами, потому что они в три, четыре раза длиннее десятичных эквивалентов. Например, двоичное число 111111111111 имеет десятичный эквивалент 4095. Однако цифровые компьютеры используют двоичные числа, а человеку приходится иногда связываться с компьютером посредством бинарных чисел (через переключатели, световые индикаторы или посредством программ, написанных на машинно-ориентированных языках). В этом случае с помощью специальных преобразователей человек может получать двоичную информации в восьмиричном или шестнадцатиричном коде. Подразумевается, что в случае необходимости, пользователь сам перейдет к двоичному коду. Это делается весьма просто и к тому же размер восьмиричного (шестнадцатиричного) кода гораздо меньше двоичного - $(111111111111)_2$ = $(7777)_8$ =$(FFF)_{16}$ - 3, 4 символа воспринимаются человеком комфортнее, чем 12.

В таблице 1.2. представлены десятичные, двоичные, восьмиричные и шестнадцатиричные коды некоторых чисел.

Таблица 1.2 - Числа различных систем счисления

| Decimal (base 10) | Binary (base 2) | Octal (base 8) | Hexadecimal (base 16) |
|---|---|---|---|
| 00 | 0000 | 00 | 0 |
| 01 | 0001 | 01 | 1 |
| 02 | 0010 | 02 | 2 |
| 03 | 0011 | 03 | 3 |
| 04 | 0100 | 04 | 4 |
| 05 | 0101 | 05 | 5 |
| 06 | 0110 | 06 | 6 |
| 07 | 0111 | 07 | 7 |
| 08 | 1000 | 10 | 8 |
| 09 | 1001 | 11 | 9 |
| 10 | 1010 | 12 | A |
| 11 | 1011 | 13 | B |
| 12 | 1100 | 14 | C |
| 13 | 1101 | 15 | D |
| 14 | 1110 | 16 | E |
| 15 | 1111 | 17 | F |

Часто в компьютерных системах используются двоично-десятичные коды (двоичное кодированное представление чисел - BCD (Binary coded decimal)), например в арифметических операциях. Для представления цифры в таком коде необходимо 4 двоичных разряда. В таблице 1.3 приведены примеры некоторых двоично-десятичных кодов десятичных цифр.

Таблица 1.3 - Примеры двоично-десятичных кодов

| Decimal digit | (BCD) 8421 | Excess-3 | 84-2-1 | 2421 |
|---|---|---|---|---|
| 0 | 0000 | 0011 | 0000 | 0000 |
| 1 | 0001 | 0100 | 0111 | 0001 |
| 2 | 0010 | 0101 | 0110 | 0010 |
| 3 | 0011 | 0110 | 0101 | 0011 |
| 4 | 0100 | 0111 | 0100 | 0100 |
| 5 | 0101 | 1000 | 1011 | 1011 |

| | | | | |
|---|---|---|---|---|
| 6 | 0110 | 1001 | 1010 | 1100 |
| 7 | 0111 | 1010 | 1001 | 1101 |
| 8 | 1000 | 1011 | 1000 | 1110 |
| 9 | 1001 | 1100 | 1111 | 1111 |

Десятичная цифра в двоично-десятичном коде с весами 8421 (взвешенный код) представляется обычным четырехзначным двоичным эквивалентом:

$$(7)_{10} = (0111)_{2\text{-}10} = (0111)_2 = 0*8 + 1*4 + 1*2 + 1*1 = 4 + 2 + 1.$$

Десятичная цифра в двоично-десятичном коде с весами 8421+3 (невзвешенный код) представляется четырехзначным двоичным эквивалентом, который увеличен на $3_{10}$ .

$$(7)_{10} = (1010)_{2\text{-}10} = (0111)_2,$$
$$(1010)_2 = (10)_{10}.$$

Двоично-десятичный код с весами 84-2-1 (взвешенный код с двумя отрицательных весами) цифры $(7)_{10}$ имеет вид 1001:

$$(7)_{10} = (1001)_{2\text{-}10} = (0111)_2 = 1*8 + 0*4 + 0*(-2) + 1*(-1) = 8 - 1.$$

При переводе десятичного числа в двоично-десятичное число каждая цифра десятичного числа заменяется ее четырехразрядным двоично-десятичным эквивалентом. Например, десятичное число 395 в двоично-десятичном коде с весами 8421 представляется как 12-разрядное число 0011 1001 0101. Цифре 3 соответствует тетрада 0011, цифре 9 - 1001, цифре 5 - 0101. Код содержит 12 бит в отличие от его двоичного эквивалента, который содержит 9 бит и имеет вид 110001011.

Также часто используется унитарный двоичный код. Это код, содержащий одну единицу и остальные нули или, содержащий один нуль и остальные единицы. Для представления десятичных цифр 0, 1, … 9 необходим десятиразрядный унитарный код. Каждый разряд такого кода соответствует определенной десятичной цифре. Например, нулю будет соответствовать код 0000000001, единице - 0000000010, а девятке - 1000000000.

Для представления отрицательных чисел используются дополнительный и обратный коды.

Существую специальные форматы для представления двоичных кодов с фиксированной и плавающей точкой. При получении помехоустойчивых кодов используют разного рода избыточности.

## 1.3 Двоичные сигналы

Бинарные сигналы в современной цифровой технике представляются потенциальным способом, т. е. постоянным напряжением высокого и низкого уровня. Высокий уровень напряжения соответствует логической 1, низкий уровень соответствует логическому 0. Например, определенные цифровые системы могут определят логическую 1 как сигнал с номинальным значением 3 В, а логический 0 как сигнал с номинальным значением 0 В.

Как показано на рис. 1.1, каждый уровень напряжения имеет допустимое отклонение от номинала. Промежуточная область между разрешенными границами пересекается только в процессе перехода от одного уровня к другому. Входы цифровых схем получают двоичные сигналы, напряжение которых лежит в областях допустимого отклонения, равно как и значения, формируемые на выходах схем.

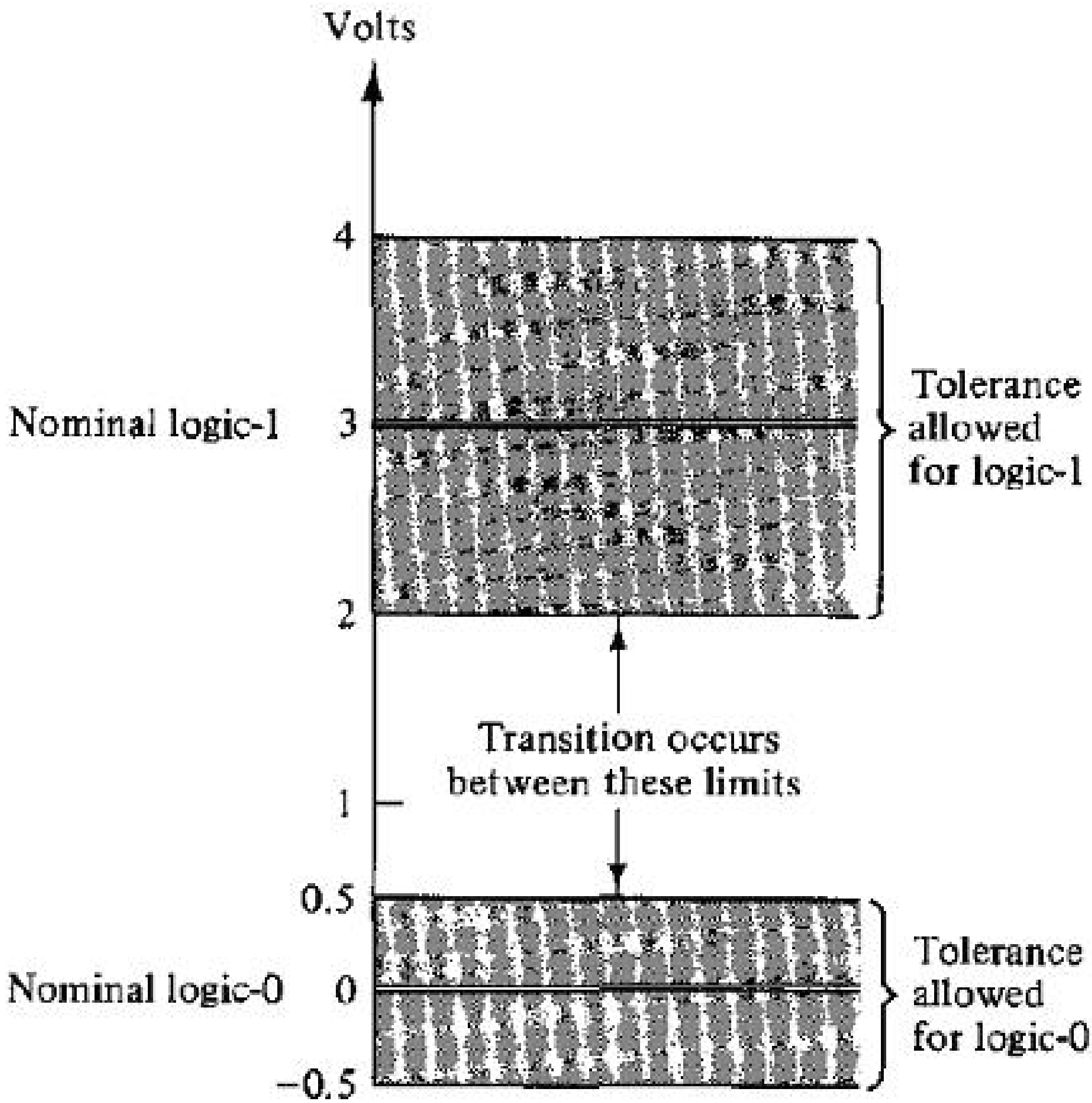


Рисунок 1.1 - Пример представления двоичного сигнала

В цифровых устройствах переменные и соответствующие им сигналы изменяются не непрерывно, а лишь в дискретные моменты времени, обозначаемые целыми неотрицательными числами: 0, 1, 2, …, i, … Временной интервал между соседними моментами дискретного времени называется тактом. Во многих случаях цифровые устройства содержат специальный блок, вырабатывающий синхронизирующие сигналы (синхроимпульсы), отмечающие моменты дискретного времени. На рис. 1.2 изображен пример временной диаграммы для некоторого сигнала А.

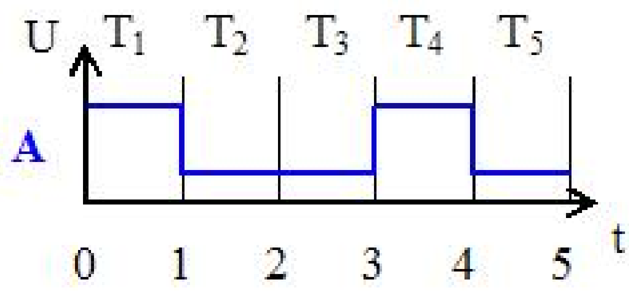


Рисунок 1.2 - Пример временной диаграммы

Изменение сигнала А показано на протяжении пяти тактов $(X_1\text{-}X_5)$ . Сигнал А принимает значение логического 0 на тактах $T_2$, $T_3$, $T_5$ и логической 1 на тактах $T_1$, $T_4$. U - напряжение, t - время.

Здесь не указывается значение напряжения (и далее мы будем абстрагироваться от него), поскольку для разных схем оно может быть разным. Это зависит от технологии, по которой выполнена интегральная схема и от ее типа.

Известны следующие интегральные технологии.

RTL (Resistor-transistor logic) - РТЛ (Резисторно-транзисторная логика).

DTL (Diode-transistor logic) - ДТЛ (Диодно-транзисторная логика).

TTL (Transistor-transistor logic) - ТТЛ (Транзисторно-транзисторная логика).

ECL (Emitter-coupled logic) - ЭСЛ (Эмиттерно-связанная логика).

MOS (Metal-oxide semiconductor) - МОП (Логика типа металл-оксид-полупроводник).

CMOS (Complementary metal-oxide semiconductor) - КМОП (Комплементарная МОП логика).

Существуют биполярные транзисторы (bipolar junction transistor (BJT)) и униполярные или полевые транзисторы (field-effect transistor (FET)). Работа биполярного транзистора зависит от потока двух типов носителей: электроны и дырки. Работа униполярных транзисторов зависит от потока только одного типа основного носителя, который может быть электронами (n-канал) или дырками (p-канал).

РТЛ, ДТЛ, ТТЛ и ЭСЛ технологии используют биполярные транзисторы, а МОП и КМОП технологии используют униполярные (полевые) транзисторы, их еще называют МДП (металл-диэлектрик (оксид)-проводник) транзисторы.

РТЛ и ДТЛ технологии не производятся сегодня, они интересны как история. ТТЛ, ЭСЛ, КМОП технологии имеют широкий спектр схем МИС (Малой степени интеграции), СИС (Средней степени интеграции), БИС (Большой степени интеграции) и СБИС (Сверхбольшой степени интеграции). МОП технологии использовались только для производства БИС и СБИС схем. Новые ЭСЛ и МОП схемы не производятся сегодня, но уже выпущенные еще используются. ТТЛ и КМОП схемы широко применяются, развиваются и используются до сих пор (от МИС до СБИС). МИС и СИС схемы производятся до сих пор (используются, например, для обвязки микроконтроллеров).Напряжение питания варьируется от примерно 0.3 до 30 В (КМОП схемы), Такая же тенденция прослеживается и для максимального значения напряжения логической 1 (только на 0,1-0,3 В ниже). Максимальное значение напряжения 0 колеблется в пределах 0,1-0,5 В.

Напряжение зависит от технологи, кроме того зависит от типа схем. Чем выше напряжение 1, тем больше требуется времени и энергии для переключения транзистора, тем ниже быстродействие, но тем менее чувствительна схема к помехам. Для устройств, подверженных помехам, проектируются высоковольтные микросхемы, хотя сегодня стали появляться и низковольтные микросхем, устойчивые к помехам. Для схем, менее подверженных помехам, и для которых быстродействие - критичный параметр, проектируются низковольтные схемы.