Логические основы ЭВМ, элементы и узлы

Практическая работа на 60 минут. Изучите короткое объяснение в каждом разделе и сразу выполните задание. Записывайте ответы прямо на странице. В конце нажмите «Отправить работу — сохранить PDF».

ФИО

Фахертдинова Анастасия Родионовна

Группа

ИС 25/9-2

Дата

2026-09-28

1

Двоичная логика и логические сигналы

Цифровая вычислительная техника работает с двумя устойчивыми состояниями, которые условно записывают как `0` и `1`.

```
0 → ложь / нет сигнала
1 → истина / есть сигнал
```

Логическая переменная может принимать только одно из двух значений: 0 или 1. Например, сигнал A может описывать состояние кнопки:

```
A = 0 → кнопка не нажата
A = 1 → кнопка нажата
```

На физическом уровне нулю и единице соответствуют разные электрические состояния. Для изучения цифровых схем важнее не конкретное напряжение, а то, как устройство обрабатывает два логических состояния.

> Главная идея: сложные цифровые устройства строятся из большого количества простых элементов, которые принимают и выдают только 0 или 1.

Задание 1 · 6 минут

1. Сколько разных значений может принимать логическая переменная?
2. Пусть сигнал **D** показывает состояние датчика двери: 0 — дверь закрыта, 1 — дверь открыта. Что означает **D = 1**?
3. Придумайте ещё один пример устройства, состояние которого можно описать значениями 0 и 1.

Ваш ответ

1) Две: 0 и 1
2) D = 1 - Дверь открыта
3) Лампа: 0 - Лампа выключена, 1 - Лампа включена

2

Базовые логические элементы: НЕ, И, ИЛИ, XOR

НЕ (NOT) инвертирует входной сигнал: 0 превращается в 1, а 1 — в 0.

```
NOT 0 = 1
NOT 1 = 0
```

И (AND) выдаёт 1 только тогда, когда оба входа равны 1. **ИЛИ (OR)** выдаёт 1, если хотя бы один вход равен 1. **XOR** выдаёт 1, когда входы различаются.

A	B	AND	OR	XOR
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

Например, дверь должна открыться только при наличии карты **И** правильного кода:

```
OPEN = CARD AND CODE
```

> XOR удобно понимать как «ровно один из двух». Если входы одинаковые, XOR выдаёт 0; если разные — 1.

Задание 2 · 8 минут

1. Пусть **A = 1**, **B = 0**. Вычислите A AND B, A OR B, A XOR B и NOT A.
2. Какой элемент нужен, если лампа должна включаться только при одновременном нажатии двух кнопок?
3. Какой элемент нужен, если сигнал должен появиться, когда сработал ровно один из двух датчиков?

Ваш ответ

1) AND = 0; OR = 1; XOR = 1; NOT A = 0
2) (И) AND
3) XOR

3

Таблицы истинности и логические выражения

Таблица истинности показывает результат логической функции для всех возможных наборов входных значений.

```
Количество наборов для n входов:
N = 2ⁿ
```

Для двух входов существует 4 комбинации. Для трёх входов — уже 8.

Рассмотрим выражение:

```
F = A AND (NOT B)
```

A	B	NOT B	F
0	0	1	0
0	1	0	0
1	0	1	1
1	1	0	0

В более сложном выражении удобно вычислять части по порядку:

```
F = (A OR B) AND (NOT C)
```

```
A = 0, B = 1, C = 0
A OR B = 1
NOT C = 1
F = 1 AND 1 = 1
```

Задание 3 · 8 минут

1. Сколько наборов входных значений будет у схемы с тремя входами?
2. Вычислите **F = (A OR B) AND (NOT C)**, если A = 1, B = 0, C = 1.
3. Вычислите **F = (A XOR B) OR C**, если A = 1, B = 1, C = 0.

Ваш ответ

1) 2^3=8

2) F = (A OR B) AND (NOT C)

A = 1, B = 0, C = 1
A OR B = 1
NOT C = 0
F = 1 AND 0 = 0

3) F = (A XOR B) OR C

A = 1, B = 1, C = 0
A XOR B = 0
C = 0
F = 0 OR 0 = 0

4

Комбинационные схемы. Полусумматор

Комбинационная схема формирует выход только по текущим значениям входов. Она не запоминает предыдущее состояние.

Один из важнейших примеров — **полусумматор**. Он складывает два одноразрядных двоичных числа A и B и выдаёт два результата:

- **S** — бит суммы;
- **C** — бит переноса.

```
S = A XOR B
C = A AND B
```

A	B	S — сумма	C — перенос
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

```
1 + 1 = 10₂
S = 0
C = 1
```

> Полный сумматор дополнительно учитывает перенос из предыдущего разряда. Из нескольких полных сумматоров строят многоразрядный сумматор.

Задание 4 · 8 минут

1. Полусумматор складывает A = 1 и B = 0. Чему равны S и C?
2. Полусумматор складывает A = 1 и B = 1. Чему равны S и C?
3. Какие два логических элемента используются в формулах полусумматора?

Ваш ответ

1) S = 1; C = 0
2) S = 0; C = 1
3) элементы XOR и AND

5

Шифраторы, дешифраторы, мультиплексоры и компараторы

Шифратор преобразует номер активной входной линии в двоичный код. В простом варианте предполагается, что активна только одна линия.

```
Шифратор 4 → 2:
вход 0 → 00
вход 1 → 01
```

```
вход 2 → 10
вход 3 → 11
```

Дешифратор выполняет обратное преобразование: двоичный код выбирает один из выходов.

```
Дешифратор 2 → 4:
00 → Y0
01 → Y1
10 → Y2
11 → Y3
```

Мультиплексор (MUX) — цифровой переключатель. Управляющий код определяет, какой из входных сигналов попадёт на общий выход.

```
MUX 4 → 1:
00 → I0
01 → I1
10 → I2
11 → I3
```

Компаратор сравнивает два двоичных числа и формирует один из результатов: A = B, A > B или A < B.

```
A = 1010₂ = 10₁₀
B = 0111₂ = 7₁₀
Результат: A > B
```

> Не путайте назначения: шифратор кодирует, дешифратор выбирает выход, мультиплексор выбирает вход, компаратор сравнивает два значения.

Задание 5 · 9 минут

1. У шифратора 4→2 активен вход № 3. Какой код он выдаст?
2. На дешифратор 2→4 поступил код 10. Какой выход станет активным?
3. На MUX 4→1 подан код 01. Какой вход выбран?
4. Сравните A = 1001_2 и B = 1010_2. Переведите оба числа в десятичную систему.

Ваш ответ

1) 11
2) Y2
3) I1
4) A = 9^10; B = 10^10; A < B

6

Последовательностные схемы: триггеры, регистры и счётчики

В отличие от комбинационных схем, **последовательностные схемы** могут хранить информацию о предыдущем состоянии.

Триггер — простейший элемент памяти. Он способен хранить один бит: 0 или 1.

```
1 триггер → 1 бит
```

Регистр — группа триггеров, предназначенная для хранения нескольких битов одновременно.

```
8 триггеров  → 8-битный регистр
16 триггеров → 16-битный регистр
32 триггера  → 32-битный регистр
```

Счётчик изменяет своё состояние при поступлении импульсов. Например, двухбитный счётчик проходит четыре состояния:

```
00 → 01 → 10 → 11 → 00 → ...
```

Регистры нужны для временного хранения операндов, адресов и промежуточных результатов. Счётчики применяют для подсчёта событий и формирования последовательностей состояний.

Задание 6 · 8 минут

1. Сколько триггеров нужно для регистра на 16 бит?
2. Двухбитный счётчик находится в состоянии 00. Запишите его состояние после 1-го, 2-го, 3-го и 4-го импульса.
3. Какой узел лучше использовать для временного хранения 8-битного числа?
4. Какой узел нужен для подсчёта числа нажатий кнопки?

Ваш ответ

1) 16 триггеров
2) После 1-го 01; После 2-го 10; После 3-го 11; После 4-го 00
3) 8-битный регистр
4) Счётчик

7

Как логические элементы превращаются в узлы процессора

Процессор состоит из огромного количества цифровых схем, но каждая сложная функция строится из более простых элементов и узлов.

Элемент или узел	Что делает
AND, OR, XOR, NOT	Выполняют базовые логические операции над битами.
Сумматор	Выполняет двоичное сложение и участвует в арифметике.
Компаратор	Сравнивает числа и коды.
Мультиплексор	Выбирает, какой сигнал или данные передать дальше.
Регистр	Временно хранит данные внутри процессора.
Счётчик	Изменяет состояние по импульсам и помогает организовывать последовательность работы.

Комбинации этих узлов образуют более крупные части процессора. Например, **АЛУ** выполняет арифметические и логические операции, а регистры хранят данные, с которыми АЛУ работает.

```
Данные в регистрах
        ↓
мультиплексоры выбирают входы
        ↓
АЛУ выполняет операцию
        ↓
результат снова записывается в регистр
```

> Главная мысль темы: процессор не является «одним сложным устройством». Он построен из большого количества простых логических элементов, объединённых в узлы.

Задание 7 · 8 минут

1. Какой узел временно хранит данные?
2. Какой узел выбирает один из нескольких входных сигналов?
3. Какой узел сравнивает два числа?
4. Объясните своими словами в 2–3 предложениях, как простые логические элементы связаны с работой процессора.

Ваш ответ

1) Регистр
2) Мультиплексор
3) Компаратор
4) Из них собирают более сложные узлы: сумматоры и мультиплексоры.
Эти узлы объединяются в крупные блоки, например в АЛУ, которое выполняет вычисления, или в регистры, которые хранят промежуточные данные.

Перед сдачей · 5 минут вместе с оформлением

✓ Указаны ФИО и группа.

✓ Выполнены все семь заданий, ответы записаны по пунктам.

✓ В вычислениях показаны промежуточные действия там, где они требуются.

После нажатия кнопки откроется окно печати. Выберите «Сохранить как PDF». В PDF попадут все ваши ответы полностью.