Вариант 1

1. Количество всевозможных функций N от n аргументов выражается зависимостью: N=2^(2n).

2.

Функция эквивалентности (XNOR) представляет собой логическую операцию, результат которой истинен (равен 1) только в том случае, если оба входа равны (либо оба истинны, либо оба ложны). В математической форме, функция эквивалентности определяется следующим образом:

$$A \leftrightarrow B = \neg(A \oplus B)$$

3.

Рассмотрим функции "равнозначность" (XNOR) и "стрелка Пирса" (NAND) и проверим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

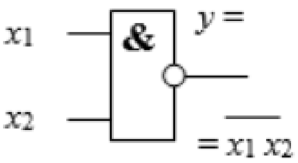
1. Функция "Равнозначность" (XNOR):
   - Удовлетворяет законам коммутативности: $A \oplus B = B \oplus A$
   - Удовлетворяет законам ассоциативности: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
   - Удовлетворяет закону поглощения: $A + A \oplus B = A$
   - Не удовлетворяет законам дистрибутивности и де Моргана.
2. Функция "Стрелка Пирса" (NAND):
   - Удовлетворяет законам де Моргана: $\overline{A \cdot B} = \overline{A} + \overline{B}$
   - Удовлетворяет законам дистрибутивности: $A \cdot (B + C) = A \cdot B + A \cdot C$
   - Не удовлетворяет законам коммутативности и ассоциативности.

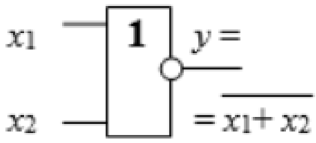
Обратите внимание, что функция "Равнозначность" удовлетворяет закону поглощения, а функция "Стрелка Пирса" удовлетворяет законам де Моргана и дистрибутивности, но не удовлетворяет законам коммутативности и ассоциативности.

4. Проблема минимизации логических функций решается на основе применения законов склеивания и поглощения с последующим перебором получаемых дизъюнктивных форм и выбором из них оптимальной (минимальной длины).

Существует большое количество методов минимизации логических функций. Все они отличаются друг от друга спецификой применения операций склеивания и поглощения, а также различными способами сокращения переборов.

5.





$$Output = \overline{(A \cdot B) + (A \cdot C) + (B \cdot C)}$$

Соответствующая схема в базисе «И-НЕ» будет выглядеть следующим образом:

lua Copy cod

```lua
A -----\
        AND ---- NOT ---- Output
B -----/
C -----/
```

6. Узлы обеспечивают одновременную обработку группы сигналов - информационных слов.

7. Дешифраторы (ДШ) — это комбинационные схемы с n входами

и m =2n выходами. Единичный сигнал, формирующийся на одном

из т выходов, однозначно соответствует комбинации входных

сигналов.

8.

Вариант 2

1.

При n=0 можно определить две основные функции *(N*=2*)*, не зависящие от каких-либо переменных:

y_0 , тождественно равную нулю (**y_0=0**), и

y_1 , тождественно равную единице (**y_1=1**).

Технической интерпретацией функции y_1=1 может быть **генератор импульсов**. При отсутствии входных сигналов на выходе этого устройства всегда имеются импульсы (единицы). Функция $y_0=0$ может быть интерпретирована **отключенной схемой**, сигналы от которой не поступают ни к каким устройствам.

2.

Функция Шеффера (означается как $NAND$ или ↑) в алгебре логики определяется как отрицание конъюнкции двух переменных. Аналитически она определяется следующим образом:

$$A \uparrow B = \neg(A \wedge B)$$

где:

- A и B - две логические переменные.
- $\wedge$ - операция логического умножения (логическое И, AND).
- $\neg$ - операция логического отрицания (логическое НЕ, NOT).

Функция Шеффера принимает значение 1 только тогда, когда оба входа равны 0. Во всех остальных случаях она принимает значение 0. Эта функция используется в алгебре логики и теории вычислений, а также в цифровых схемах, где является одним из базисных логических элементов.

3.

Рассмотрим функции "стрелка Пирса" (NAND) и "импликация" и проверим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

1. Функция "Стрелка Пирса" (NAND):
   - Удовлетворяет законам де Моргана: $\overline{A \cdot B} = \overline{A} + \overline{B}$
   - Удовлетворяет законам дистрибутивности: $A \cdot (B + C) = A \cdot B + A \cdot C$
   - Не удовлетворяет законам коммутативности и ассоциативности.
2. Функция "Импликация":
   - Удовлетворяет закону поглощения: $A + (A \rightarrow B) = A + B$
   - Удовлетворяет закону склеивания: $A \rightarrow A = 1$
   - Не удовлетворяет законам коммутативности и дистрибутивности.

Обратите внимание, что функция "Стрелка Пирса" удовлетворяет законам де Моргана и дистрибутивности, но не удовлетворяет законам коммутативности и ассоциативности. С другой стороны, функция "Импликация" удовлетворяет законам поглощения и склеивания, но не удовлетворяет законам коммутативности и дистрибутивности.

4.

- ***Правило де Моргана***

$$\overline{x_1 \vee x_2} = \overline{x}_1 \cdot \overline{x}_2 \qquad \overline{x_1 \cdot x_2} = \overline{x}_1 \vee \overline{x}_2.$$

5.

Для создания схемы в базисе «И-НЕ», фиксирующей наличие сигнала «1» на любом одном входе или сразу на всех входах в 3-входовой схеме, мы можем использовать операции И (AND) и НЕ (NOT).

$$Output = (A + B + C)$$

Соответствующая схема в базисе «И-НЕ» б

```lua
A -------\
          AND ---- Output
B -------/
C -------/
```

6. Блоки реализуют некоторую последовательность в обработке информационных слов —

функционально обособленную часть машинных операций (блок выборки команд, блок

записи-чтения и др.).

7. Таким образом, главное отличие между ними заключается в том, что схемы с памятью имеют встроенные элементы памяти для хранения информации, в то время как комбинационные схемы зависят только от текущих входных значений и не сохраняют свое состояние между операциями.

8.

Вариант 3

1. Количество всевозможных функций N от n аргументов выражается зависимостью: N=2^(2n).

2.

Функция сложения по модулю 2, также известная как исключающее ИЛИ (XOR), определяет логическую операцию, возвращающую истину (1), если количество единиц на входах является нечётным, и ложь (0) в противном случае. Аналитически она определяется следующим образом:

$$A \oplus B = (A \wedge \neg B) \vee (\neg A \wedge B)$$

где:

- A и B - две логические переменные.
- $\wedge$ - операция логического И (AND).
- $\vee$ - операция логического ИЛИ (OR).
- $\neg$ - операция логического отрицания (NOT).

Функция сложения по модулю 2 (XOR) возвращает 1 только в том случае, когда количество входных единиц нечётно, и 0 во всех остальных случаях. Эта функция широко используется в цифровых схемах и криптографии, где играет важную роль в обработке информации.

3.

Функция "штрих Шеффера" (NOR) и функция "импликация" обе важны в алгебре логики. Рассмотрим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

1. Функция "Штрих Шеффера" (NOR):
   - Удовлетворяет законам де Моргана: $\overline{A+B} = \overline{A} \cdot \overline{B}$
   - Удовлетворяет законам поглощения: $A + A \cdot B = A$
   - Не удовлетворяет законам коммутативности и ассоциативности.
2. Функция "Импликация":
   - Удовлетворяет закону поглощения: $A + (A \rightarrow B) = A + B$
   - Удовлетворяет закону склеивания: $A \rightarrow A = 1$
   - Не удовлетворяет законам коммутативности и дистрибутивности.

Обратите внимание, что функция "Штрих Шеффера" удовлетворяет законам де Моргана и поглощения, но не удовлетворяет законам коммутативности и ассоциативности. С другой стороны, функция "Импликация" удовлетворяет законам поглощения и склеивания, но не удовлетворяет законам коммутативности и дистрибутивности.

4. Закон алгебры логики, на основании которого выполняется склеивание импликант при минимизации, называется законом склеивания (или законом объединения).

5.

Для построения схемы в базисе "И-НЕ", которая обрабатывает результаты решения трех задач (№1, №2 и №3) и возвращает истину в случае решения двух из них или всех трех задач, мы можем использовать операции И (AND) и НЕ (NOT) для каждой пары задач. Затем объединить результаты с помощью операции ИЛИ (OR).

$$Output = (A \wedge B) \vee (B \wedge C) \vee (A \wedge C) \vee (A \wedge B \wedge C)$$

Соответствующая схема в базисе "И-НЕ" будет выглядеть следующим образо

```lua
A ----\
        AND ----\
B ----/            OR ---- Output
C ----/
```

6. Функционально микросхемы могут соответствовать устройству, узлу или блоку, но каждая из них состоит из комбинации простейших логических элементов, реализующих функции формирования, преобразования, запоминания сигналов и т.д.

7. Сумматоры накапливающего типа строят на сложных JKRS-триггерах, дополняя их выходы достаточно сложными схемами формирования и распространения переносов.

Чаще на практике используются сумматоры комбинационного типа. У такого сумматора на входе и выходе имеются регистры для хранения и преобразования кодов операндов и результата

8.

Вариант 4

1. Количество всевозможных функций N от n аргументов выражается зависимостью: N=2^(2n).

2.

Функция импликации, также известная как логическое следование или условное выражение, определяет отношение между двумя высказываниями. В алгебре логики она обозначается символом "→" или "⇒". Аналитически, она определяется следующим образом:

$$A \Rightarrow B = \neg A \vee B$$

где:

- A и B - две логические переменные или высказывания.
- $\neg$ - операция отрицания (NOT).
- $\vee$ - операция логического ИЛИ (OR).

Функция импликации возвращает истину (1) в случае, когда первое высказывание ложно или когда оба высказывания истинны. Она возвращает ложь (0) только в том случае, когда первое высказывание истинно, а второе ложно. функция импликации играет важную роль в математике, логике, и программировании, особенно в условных выражениях и логических операциях.

3.

Функция "эквивалентность" (XNOR) и функция "сложение по модулю 2" (XOR) обе являются важными функциями в алгебре логики. Рассмотрим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

1. Функция "Эквивалентность" (XNOR):
   - Удовлетворяет законам коммутативности: $A \oplus B = B \oplus A$
   - Удовлетворяет законам ассоциативности: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
   - Не удовлетворяет законам дистрибутивности, поглощения, склеивания, свертки и де Моргана.
2. Функция "Сложение по модулю 2" (XOR):
   - Удовлетворяет законам коммутативности: $A \oplus B = B \oplus A$
   - Удовлетворяет законам ассоциативности: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
   - Не удовлетворяет законам дистрибутивности, поглощения, склеивания, свертки и де Моргана.

Обратите внимание, что функции "Эквивалентность" и "Сложение по модулю 2" удовлетворяют законам коммутативности и ассоциативности, но не удовлетворяют другим перечисленным законам.

4.

- ***Закон свёртки***

$$x \vee \bar{x}F = x \vee F \qquad x(\bar{x} \vee F) = xF$$

5. Для создания схемы в базисе «И-НЕ», которая будет фиксировать отсутствие сигналов на любых двух входах в 3-входовой схеме, мы можем использовать операции И (AND) и НЕ (NOT) для всех возможных комбинаций входов.

```
A ------\
         AND ---- NOT ---- Output
B ------/
C ------/
```

6.

7.

Построим сумматор, в котором сложение производится как поразрядная операция и на распространение переноса не требуется дополнительного времени.

Для этого удобно ввести две функции:

γ — функция генерации переноса **$\gamma = x\&y$**

Функция равна 1, когда перенос в этом разряде возникает независимо от переноса на его входе.

π — функция прозрачности **$\pi = x \vee y$**

Функция равна 1, когда при возникновении переноса на входе данного разряда, на его выходе также возникает перенос, т.е. тракт разряда прозрачен для входного переноса.

8.

Вариант 5

1. Количество всевозможных функций N от n аргументов выражается зависимостью: N=2^(2n).

2.

Функция Шеффера, также известная как стрелка Пирса или логическое умножение отрицания, определяет логическую операцию, которая возвращает истину только тогда, когда оба входа ложны. Аналитически она определяется следующим образом:

$$A|B = \neg(A \wedge B)$$

где:

- A и B - две логические переменные.
- $\neg$ - операция логического отрицания (NOT).
- $\wedge$ - операция логического умножения (AND).

Функция Шеффера возвращает 1 только когда оба входа ложны, во всех остальных случаях она возвращает 0. Эта функция получила название в честь американского логика Генри Шеффера и широко используется в алгебре логики и цифровых схемах, часто в качестве базисной логической операции.

3.

Рассмотрим функции "сложение по модулю 2" (XOR) и "стрелка Пирса" (NAND) и проверим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

1. Функция "Сложение по модулю 2" (XOR):
   - Удовлетворяет законам коммутативности: $A \oplus B = B \oplus A$
   - Удовлетворяет законам ассоциативности: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
   - Не удовлетворяет законам дистрибутивности, поглощения, склеивания, свертки и де Моргана.
2. Функция "Стрелка Пирса" (NAND):
   - Удовлетворяет законам де Моргана: $\overline{A \cdot B} = \overline{A} + \overline{B}$
   - Удовлетворяет законам дистрибутивности: $A \cdot (B + C) = A \cdot B + A \cdot C$
   - Не удовлетворяет законам коммутативности и ассоциативности.

Обратите внимание, что функция "Сложение по модулю 2" удовлетворяет законам коммутативности и ассоциативности, но не удовлетворяет другим перечисленным законам. С другой стороны, функция "Стрелка Пирса" удовлетворяет законам де Моргана и дистрибутивности, но не удовлетворяет законам коммутативности и ассоциативности.

4. В процедуре минимизации булевых функций, таблица покрытия строится на основе закона поглощения.

- ***Закон поглощения***

В дизъюнктивной форме ЛФ конъюнкция меньшего ранга, т.е. с меньшим числом переменных, поглощает все конъюнкции большего ранга, если ее изображение содержится в них. Это же справедливо и для конъюнктивных форм

$X_1 \vee X_1 \cdot X_2 = X_1$

$X_1 \cdot (X_1 \vee X_2) = X_1$

5. Во всех вычислительных машинах используются и параллельно-последовательные коды

представления информации. При этом информация отображается частями. Части поступают

на обработку последовательно, а каждая часть данных представляется параллельным

кодом.

6.

Для фиксации наличия "1-ых" сигналов одновременно на любых двух входах в 3-входовой схеме с использованием базиса "И-ИЛИ-НЕ", мы можем построить логическую схему с использованием элементов И (AND), ИЛИ (OR) и НЕ (NOT).

```
A -------\
          AND ----\
B -------/         \
                    OR ---- Output
C -----------------/
```

7. Таким образом, главное отличие между ними заключается в том, что схемы с памятью имеют встроенные элементы памяти для хранения информации, в то время как комбинационные схемы зависят только от текущих входных значений и не сохраняют свое состояние между операциями.

8.

Вариант 6

1.

При n=0 можно определить две основные функции *(N=2)*, не зависящие от каких-либо переменных:

y_0 , тождественно равную нулю (**y_0=0**), и

y_1 , тождественно равную единице (**y_1=1**).

Технической интерпретацией функции $y_1=1$ может быть **генератор импульсов**. При отсутствии входных сигналов на выходе этого устройства всегда имеются импульсы (единицы). Функция $y_0=0$ может быть интерпретирована **отключенной схемой**, сигналы от которой не поступают ни к каким устройствам.

2.

Функция "стрелка Пирса", также известная как штрих Шеффера, определяет логическую операцию, которая возвращает ложь только в том случае, когда оба входа истинны. Аналитически она определяется следующим образом:

$$A \downarrow B = \neg(A \vee B)$$

где:

- A и B - две логические переменные.
- $\neg$ - операция логического отрицания (NOT).
- $\vee$ - операция логического сложения (OR).

Функция "стрелка Пирса" возвращает 1 только когда оба входа ложны, во всех остальных случаях она возвращает 0. Эта функция получила название в честь американского логика Чарлза Сандерса Пирса и также широко используется в алгебре логики и цифровых схемах, часто в качестве базисной логической операции.

3.

Рассмотрим функции "импликация" и "сложение по модулю 2" (XOR) и проверим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

1. Функция "Импликация":
   - Удовлетворяет закону поглощения: $A + (A \rightarrow B) = A + B$
   - Удовлетворяет закону склеивания: $A \rightarrow A = 1$
   - Не удовлетворяет законам коммутативности и дистрибутивности.
2. Функция "Сложение по модулю 2" (XOR):
   - Удовлетворяет законам коммутативности: $A \oplus B = B \oplus A$
   - Удовлетворяет законам ассоциативности: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
   - Не удовлетворяет законам дистрибутивности, поглощения, склеивания, свертки и де Моргана.

Обратите внимание, что функция "Импликация" удовлетворяет законам поглощения и склеивания, но не удовлетворяет законам коммутативности и дистрибутивности. С другой стороны, функция "Сложение по модулю 2" удовлетворяет законам коммутативности и ассоциативности, но не удовлетворяет другим перечисленным законам.

4.

- ***Закон свёртки***

$$x \vee \bar{x}F = x \vee F \qquad x(\bar{x} \vee F) = xF$$

5.

Для фиксации наличия "1-ых" сигналов, появляющихся одновременно на любых двух входах или на всех входах в 3-входовой схеме с использованием базиса "И-НЕ", можно использовать элементы И (AND), ИЛИ (OR) и НЕ (NOT).

```
A -------\
          AND ----\
B -------/         \
                    OR ---- Output
C -------\         /
          AND ----/
```

6. Машинный такт (или просто такт) — основная единица времени в работе центрального процессора (ЦП) компьютера. Он представляет собой один полный цикл работы процессора, включающий выполнение одной базовой операции.

7. Состояние "Сброс" (R=0, S=1): триггер сбрасывается, выход устанавливается в 0.
Состояние "Установка" (R=1, S=0): триггер устанавливается, выход устанавливается в 1. 3
Состояние "Хранения" (R=0, S=0 или R=1, S=1): триггер находится в устойчивом состоянии, выход

Состояние "Запрещено" (R=1, S=1): состояние запрещено, оно может привести к неопределенному поведению триггера.

8.

Вариант 7

1. Количество всевозможных функций N от n аргументов выражается зависимостью: N=2^(2n).

2.

Функция импликации, также известная как логическое следование или условное выражение, определяет отношение между двумя высказываниями. В алгебре логики она обозначается символом "→" или "⇒". Аналитически, она определяется следующим образом:

$$A \Rightarrow B = \neg A \vee B$$

где:

- A и B - две логические переменные или высказывания.
- $\neg$ - операция отрицания (NOT).
- $\vee$ - операция логического ИЛИ (OR).

Функция импликации возвращает истину (1) в случае, когда первое высказывание ложно или когда оба высказывания истинны. Она возвращает ложь (0) только в том случае, когда первое высказывание истинно, а второе ложно. функция импликации играет важную роль в математике, логике, и программировании, особенно в условных выражениях и логических операциях.

3.

Рассмотрим функции "штрих Шеффера" (NOR) и "эквивалентность" (XNOR) и проверим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

1. Функция "Штрих Шеффера" (NOR):
   - Удовлетворяет законам де Моргана: $\overline{A+B} = \overline{A} \cdot \overline{B}$
   - Удовлетворяет законам поглощения: $A + A \cdot B = A$
   - Не удовлетворяет законам коммутативности и ассоциативности.
2. Функция "Эквивалентность" (XNOR):
   - Удовлетворяет законам коммутативности: $A \oplus B = B \oplus A$
   - Удовлетворяет законам ассоциативности: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
   - Не удовлетворяет законам дистрибутивности, поглощения, склеивания, свертки и де Моргана.

Обратите внимание, что функция "Штрих Шеффера" удовлетворяет законам де Моргана и поглощения, но не удовлетворяет законам коммутативности и ассоциативности. С другой стороны, функция "Эквивалентность" удовлетворяет законам коммутативности и ассоциативности, но не удовлетворяет другим перечисленным законам.

4

- ***Закон поглощения***

В дизъюнктивной форме ЛФ конъюнкция меньшего ранга, т.е. с меньшим числом переменных, поглощает все конъюнкции большего ранга, если ее изображение содержится в них. Это же справедливо и для конъюнктивных форм

$X_1 \vee X_1 \cdot X_2 = X_1$

$X_1 \cdot (X_1 \vee X_2) = X_1$

.

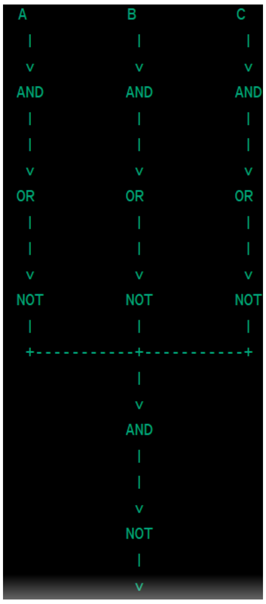
Доказательство:

Для доказательства закона поглощения используем таблицу истинности булевой алгебры:

A	B	AB	$A + AB$
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Из таблицы истинности видно, что при любых значениях переменных A и B выполняется равенство $A + AB = A$, что и подтверждает закон поглощения.

5. Давайте обозначим решение задач №1, №2 и №3 как A, B и C соответственно. Тогда результатом, который считается, будет решение 2-х любых из 3-х задач или решение всех задач.



6.

7. Для RS-триггера существует так называемый запрещенный или нежелательный состояний, когда оба входа (S и R) находятся в высоком состоянии (1). Это приводит к состоянию метастабильности, когда выходное

состояние трудно определить и может привести к непредсказуемому поведению триггера. Это нежелательное состояние известно как запрещенное состояние RS-триггера.

8.

Вариант 8

1. Схема, реализующая

зависимость y2=x, называется повторителем, а схема y3=¬x - инвертором.

2.

Функция "стрелка Пирса", также известная как штрих Шеффера, определяет логическую операцию, которая возвращает ложь только в том случае, когда оба входа истинны. Аналитически она определяется следующим образом:

$$A \downarrow B = \neg(A \vee B)$$

где:

- A и B - две логические переменные.
- $\neg$ - операция логического отрицания (NOT).
- $\vee$ - операция логического сложения (OR).

Функция "стрелка Пирса" возвращает 1 только когда оба входа ложны, во всех остальных случаях она возвращает 0. Эта функция получила название в честь американского логика Чарлза Сандерса Пирса и также широко используется в алгебре логики и цифровых схемах, часто в качестве базисной логической операции.

3.

Рассмотрим функции "сложение по модулю 2" (XOR) и "эквивалентность" (XNOR) и проверим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

. Функция "Сложение по модулю 2" (XOR):
  - Удовлетворяет законам коммутативности: $A \oplus B = B \oplus A$
  - Удовлетворяет законам ассоциативности: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
  - Не удовлетворяет законам дистрибутивности, поглощения, склеивания, свертки и де Моргана.

. Функция "Эквивалентность" (XNOR):
  - Удовлетворяет законам коммутативности: $A \oplus B = B \oplus A$
  - Удовлетворяет законам ассоциативности: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
  - Не удовлетворяет законам дистрибутивности, поглощения, склеивания, свертки и де Моргана.

Обратите внимание, что функции "Сложение по модулю 2" и "Эквивалентность" обе удовлетворяют законам коммутативности и ассоциативности, но не удовлетворяют другим перечисленным законам.

4. Закон алгебры логики, на основании которого выполняется склеивание импликант при минимизации, называется законом склеивания (или законом поглощения).

- ***Закон поглощения***

В дизъюнктивной форме ЛФ конъюнкция меньшего ранга, т.е. с меньшим числом переменных, поглощает все конъюнкции большего ранга, если ее изображение содержится в них. Это же справедливо и для конъюнктивных форм

$X_1 \vee X_1 \cdot X_2 = X_1$

$X_1 \cdot (X_1 \vee X_2) = X_1$

5. Для фиксации отсутствия сигналов на любых двух входах в 3-входовой схеме с использованием базиса "И-НЕ", можно построить схему, используя элементы И (AND) и НЕ (NOT).

```
A -------\
          AND ----\
B -------/         \
                    NOT ---- Output
C ----------------/
```

6. Длительность машинного такта измеряется во времени и обычно выражается в наносекундах (нс) или пикосекундах (пс), в зависимости от того, насколько короткими являются периоды тактового сигнала.

7. При построении суммирующего счетчика обычно используются JK-триггеры. Число триггеров определяет количество возможных состояний счетчика. Если у вас есть n триггеров, максимальное значение, которое можно представить в двоичной системе, равно 2^n - 1. Например, при использовании 4 триггеров (четырехразрядный счетчик) максимальное значение будет 2^4 - 1 = 15.

8.

Вариант 9

1. Количество всевозможных функций N от n аргументов выражается зависимостью: N=2^(2n).

2.

Функция сложения по модулю 2, также известная как исключающее ИЛИ (XOR), определяет логическую операцию, которая возвращает истину только в том случае, когда количество истинных входов нечетно. Аналитически она определяется следующим образом:

$$A \oplus B = (A \wedge \neg B) \vee (\neg A \wedge B)$$

где:

- A и B - две логические переменные.
- $\wedge$ - операция логического умножения (AND).
- $\vee$ - операция логического сложения (OR).
- $\neg$ - операция логического отрицания (NOT).

Эта функция возвращает 1 только когда количество входов, равных 1, нечетно. Если количество входов, равных 1, четно или равно 0, функция возвращает 0. Функция сложения по модулю 2 широко используется в цифровых схемах и криптографии.

3.

Рассмотрим функции "штрих Шеффера" (NOR) и "стрелка Пирса" (NAND) и проверим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

1. Функция "Штрих Шеффера" (NOR):
   - Удовлетворяет законам де Моргана: $\overline{A+B} = \overline{A} \cdot \overline{B}$
   - Удовлетворяет законам поглощения: $A + A \cdot B = A$
   - Не удовлетворяет законам коммутативности и ассоциативности.
2. Функция "Стрелка Пирса" (NAND):
   - Удовлетворяет законам де Моргана: $\overline{A \cdot B} = \overline{A} + \overline{B}$
   - Удовлетворяет законам дистрибутивности: $A \cdot (B + C) = A \cdot B + A \cdot C$
   - Не удовлетворяет законам коммутативности и ассоциативности.

Обратите внимание, что функция "Штрих Шеффера" удовлетворяет законам де Моргана и поглощения, но не удовлетворяет законам коммутативности и ассоциативности. С другой стороны, функция "Стрелка Пирса" удовлетворяет законам де Моргана и дистрибутивности, но не удовлетворяет законам коммутативности и ассоциативности.

4.

- ***Правило де Моргана***

$$\overline{x_1 \vee x_2} = \overline{x}_1 \cdot \overline{x}_2 \qquad \overline{x_1 \cdot x_2} = \overline{x}_1 \vee \overline{x}_2.$$

5.

```
A -----------\
              AND -----\
B -----------/          \
                         AND ------\
C ------------\                     \
              AND ----\              OR ---- Output
                         AND ------/
              NOT ----/
```

6. Элементы ЭВМ можно классифицировать по различным признакам.

Наиболее часто такими признаками являются:

• тип сигналов,

• назначение элементов,

• технология их изготовления и т.д.

7. При значении Т =0 триггер сохраняет свое ранее

установленное состояние - режим хранения состояния,

при Т=1 триггер переходит в противоположное состояние.

8.

Вариант 10

1. Схема, реализующая

зависимость y2=x, называется повторителем, а схема y3=⌐x - инвертором.

2.

Функция эквивалентности в логике обозначается как $A \equiv B$ и определяется аналитически следующим образом:

$$A \equiv B = (A \wedge B) \vee (\neg A \wedge \neg B)$$

где:

- A и B - две логические переменные.
- $\wedge$ - операция логического умножения (AND).
- $\vee$ - операция логического сложения (OR).
- $\neg$ - операция логического отрицания (NOT).

Функция эквивалентности возвращает 1 (истину), если оба входа равны между собой (либо оба истинны, либо оба ложны). Возвращает 0 (ложь), если входы различны. Эта функция используется для определения равенства между двумя логическими выражениями.

3.

Рассмотрим функции "штрих Шеффера" (NOR) и "сложение по модулю 2" (XOR) и проверим, каким законам алгебры логики они удовлетворяют, а каким не удовлетворяют:

1. Функция "Штрих Шеффера" (NOR):
   - Удовлетворяет законам де Моргана: $\overline{A+B} = \overline{A} \cdot \overline{B}$
   - Удовлетворяет законам поглощения: $A + A \cdot B = A$
   - Не удовлетворяет законам коммутативности и ассоциативности.
2. Функция "Сложение по модулю 2" (XOR):
   - Удовлетворяет законам коммутативности: $A \oplus B = B \oplus A$
   - Удовлетворяет законам ассоциативности: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
   - Не удовлетворяет законам дистрибутивности, поглощения, склеивания и де Моргана.

Обратите внимание, что функция "Штрих Шеффера" удовлетворяет законам де Моргана и поглощения, но не удовлетворяет законам коммутативности и ассоциативности. С другой стороны, функция "Сложение по модулю 2" удовлетворяет законам коммутативности и ассоциативности, но не удовлетворяет другим перечисленным законам.

4. на основе закона покрытия

- ***Закон поглощения***

В дизъюнктивной форме ЛФ конъюнкция меньшего ранга, т.е. с меньшим числом переменных, поглощает все конъюнкции большего ранга, если ее изображение содержится в них. Это же справедливо и для конъюнктивных форм

$X_1 \vee X_1 \cdot X_2 = X_1$

$X_1 \cdot (X_1 \vee X_2) = X_1$

5. Для фиксации наличия "1-ых" сигналов, появляющихся одновременно на любых двух входах или на всех входах в 3-входовой схеме с использованием базиса "И-НЕ", можно построить схему, используя элементы И (AND) и НЕ (NOT).

```
A -------\
          AND ----\
B -------/         \
                    AND ---- Output
C -------/         /
          AND ----/
```

6. При импульсном способе представления сигналов единичному значению некоторой двоичной переменной ставится в соответствие наличие импульса (тока или напряжения), нулевому значению — отсутствие импульса. Длительность импульсного сигнала не превышает один такт синхроимпульсов.

7.

8.

Законы алгебры логики (см курс Мат логики)

В алгебре логики установлен целый ряд законов, с помощью которых возможно преобразование логических функций (ЛФ):

- коммутативный (переместительный)
  $x_1\&x_2=x_2\&x_1$
  $x_1 V\, x_2=x_2 V\, x_1$

- ассоциативный (сочетательный)
  $(x_1\&x_2)\&x_3=(x_1\&x_3)\&x_2=x_1\&(x_2\&x_3)$
  $(x_1Vx_2)Vx_3=(x_1Vx_3)Vx_2=x_1V(x_2Vx_3)$

Эти законы полностью идентичны законам обычной алгебры;

- дистрибутивный (распределительный)
  $x_1 \cdot (x_2 \vee x_3) = x_1 \cdot x_2 \vee x_1 \cdot x_3$
  $x_1 \vee x_2 \cdot x_3 = (x_1 \vee x_2)(x_1 \vee x_3)$

A	B	A ↓ B
0	0	1
0	1	0
1	0	0
1	1	0

A	B	A | B
0	0	1
0	1	1
1	0	1
1	1	0

A	**B**	**A⊕B**
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**0**