

НАДЁЖНОСТЬ ПОДВИЖНОГО СОСТАВА

Автор:

кандидат технических наук, доцент
кафедры «Вагоны и вагонное хозяйство»
Александр Анатольевич Иванов

ТЕМА 5

НАДЁЖНОСТЬ СИСТЕМ

5.1. ПОНЯТИЕ СИСТЕМЫ

5.2. ТЕХНОЛОГИЯ ПОСТРОЕНИЯ РАСЧЁТНОЙ СХЕМЫ СИСТЕМЫ (ВАГОНА)

5.3. СТРУКТУРНЫЕ ФУНКЦИИ СИСТЕМ

5.4. СИСТЕМЫ С ПРИВОДИМОЙ И НЕПРИВОДИМОЙ СТРУКТУРОЙ

5.5. МЕТОД СТРУКТУРНЫХ СХЕМ

5.6. МЕТОД ПЕРЕБОРА СОСТОЯНИЙ

5.7. МЕТОД ЛОГИЧЕСКИХ СХЕМ

5.7.1. МЕТОД МИНИМАЛЬНЫХ ПУТЕЙ

5.7.2. МЕТОД МИНИМАЛЬНЫХ СЕЧЕНИЙ

5.7.3. МЕТОД РАЗЛОЖЕНИЯ ПО БАЗОВОМУ ЭЛЕМЕНТУ

5.8. МЕТОД ДЕРЕВА ОТКАЗОВ (ДЕРЕВА СОБЫТИЙ)

5.9. НАДЁЖНОСТЬ РЕМОНТИРУЕМЫХ СИСТЕМ

5.10. НАДЁЖНОСТЬ СИСТЕМЫ СО МНОГИМИ СОСТОЯНИЯМИ

5.7. МЕТОД ЛОГИЧЕСКИХ СХЕМ

*Метод основан на использовании алгебры логики
(булевой алгебры)*

ОТСТУПЛЕНИЕ:

ОСНОВНЫЕ ПОЛОЖЕНИЯ АЛГЕБРЫ ЛОГИКИ

1. Любой элемент системы и вся система могут находиться только в одном из двух возможных состояний:

работоспособном неработоспособном

2. Для описания состояний элементов будем использовать булевы переменные, которые обозначим:

работоспособное состояние $\rightarrow a, e, u$

неработоспособное состояние $\rightarrow \bar{a}, \bar{e}, \bar{u}$

Т.о. переменная «состояние элемента» может принять одно из двух значений:

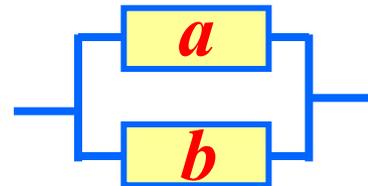
$$a = \begin{cases} 1, & \text{если элемент в работоспособном состоянии} \\ 0, & \text{если элемент в неработоспособном состоянии} \end{cases}$$

3. Логические операции с булевыми переменными:

Логическое умножение (конъюнкция):

$$y = a \cdot b = a \text{ и } b = a \cap b$$


Логическое сложение (дизъюнкция):

$$y = a + b = a \text{ или } b = a \cup b$$


Логическое отрицание:

$$y = \bar{a} = \text{не } a$$


4. *Законы булевой алгебры:*

$$(a \cdot b) = (b \cdot a)$$

переместительный

$$(a \cdot b) \cdot c = (b \cdot c) \cdot a$$

сочетательный

$$(a+b) \cdot c = a \cdot c + b \cdot c$$

распределительный

$$a + (b \cdot c) = (a + b) \cdot (a + c)$$

5. *Булевы тождества:*

$$a + \bar{a} = 1$$

$$a \cdot \bar{a} = 0$$

$$a \cdot 1 = a$$

$$a + 1 = 1$$

$$a + a = a$$

$$a \cdot a = a$$

$$a \cdot 0 = 0$$

$$a + 0 = a$$

6. *Законы поглощения:*

$$a+(a \cdot c) = a$$

$$(a \cdot c) \cdot a = a \cdot c$$

КОНЕЦ ОТСТУПЛЕНИЯ

Технология метода логических схем:

1. Составляют логическую схему работы системы
2. Применяя алгебру логики, получают вероятности искомых событий

Для составления логических схем системы можно воспользоваться:

- методом минимальных путей (системы)
 - методом минимальных сечений (системы)
-

5.7.1. Метод минимальных путей

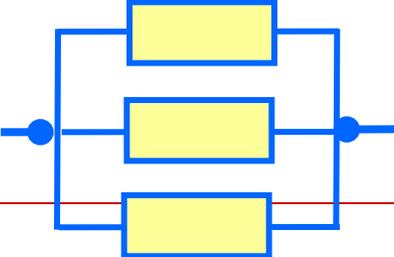
ОПРЕДЕЛЕНИЕ

ПУТЬ – совокупность элементов, **функционирование** которых обеспечивает работоспособное состояние системы

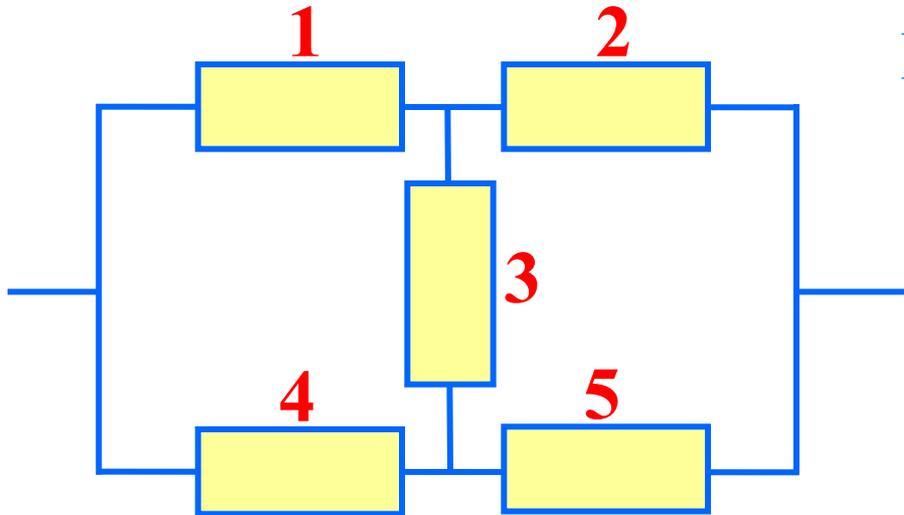
МИНИМАЛЬНЫЙ ПУТЬ – последовательный набор элементов, которые обеспечивают работоспособное состояние системы, а отказ любого из них приводит к отказу системы

НАПРИМЕР:

Для системы:  минимальный путь один

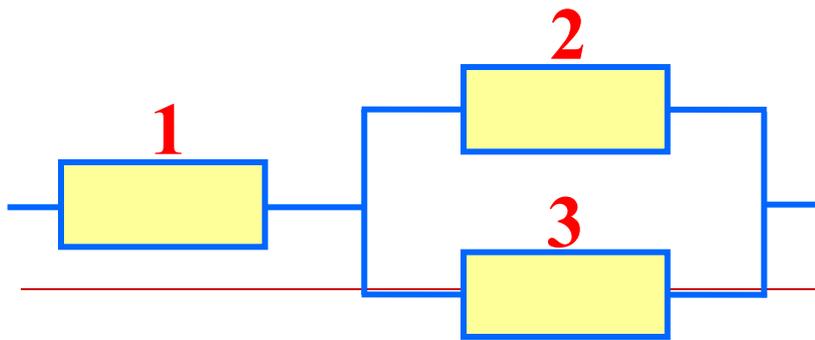
Для системы:  минимальных путей три

Рассмотрим мостиковую структуру



Имеем 4 минимальных пути:

- 1: → 1 и 2
- 2: → 4 и 5
- 3: → 1 и 3 и 5
- 4: → 4 и 3 и 2

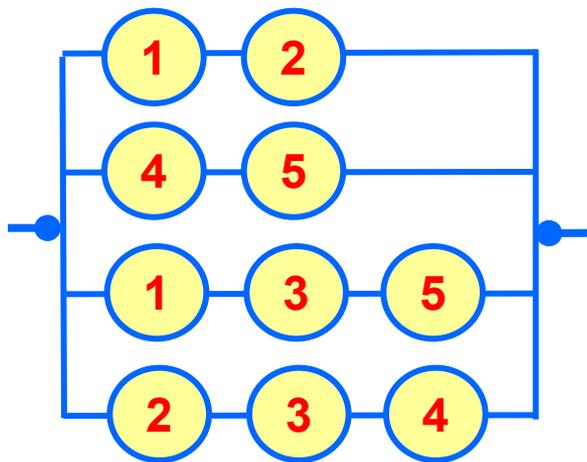
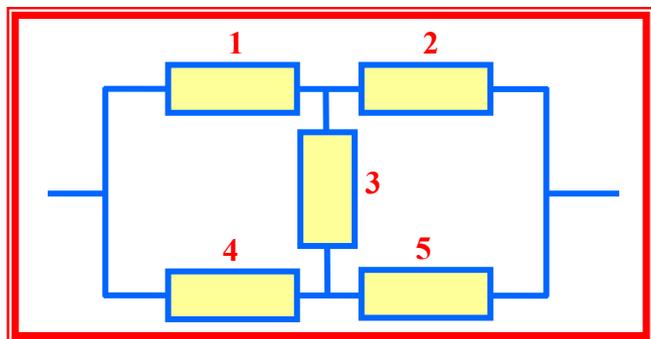


Сколько минимальных путей?

Правила построения ЛОГИЧЕСКОЙ СХЕМЫ методом минимальных путей

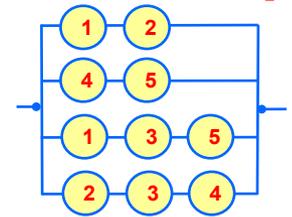
1. Все возможные минимальные пути ставят параллельно
2. Все элементы минимального пути ставят последовательно

ЛОГИЧЕСКАЯ СХЕМА ДЛЯ МОСТИКОВОЙ СТРУКТУРЫ:



ЗДЕСЬ **5**
работоспособное
состояние
элемента 5,
обозначим его
переменной a_5

Для логической схемы составляем функцию алгебры логики.



Отказ логической схемы произойдёт при одновременном отказе всех 4 минимальных путей, а безотказная работа хотя бы одной ветви гарантирует работоспособное состояние системы.

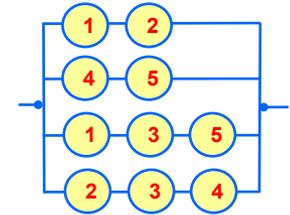
$$\left[\begin{array}{l} \text{ОТКАЗ} \\ \text{СХЕМЫ} \end{array} \right] \equiv \bar{A} \equiv \left[\begin{array}{l} \bar{1} \text{ или } \bar{2} \quad \text{и} \quad \bar{4} \text{ или } \bar{5} \quad \text{и} \\ \bar{1} \text{ или } \bar{3} \text{ или } \bar{5} \quad \text{и} \quad \bar{2} \text{ или } \bar{3} \text{ или } \bar{4} \end{array} \right]$$

$$= 1 - a_1 \cdot a_2 \quad \text{и} \quad 1 - a_4 \cdot a_5 \quad \text{и} \quad 1 - a_1 \cdot a_3 \cdot a_5 \quad \text{и} \quad 1 - a_2 \cdot a_3 \cdot a_4 =$$

$$= (1 - a_1 \cdot a_2) \cdot (1 - a_4 \cdot a_5) \cdot (1 - a_1 \cdot a_3 \cdot a_5) \cdot (1 - a_2 \cdot a_3 \cdot a_4)$$

$$\left[\underline{\text{НЕОТКАЗ}} \right] \equiv A = 1 - (1 - a_1 \cdot a_2) \cdot (1 - a_4 \cdot a_5) \cdot (1 - a_1 \cdot a_3 \cdot a_5) \cdot (1 - a_2 \cdot a_3 \cdot a_4)$$

Раскрывая скобки и используя законы алгебры логики, получим:



$$A = a_1 \cdot a_4 + a_2 \cdot a_5 + a_1 \cdot a_3 \cdot a_5 + a_2 \cdot a_3 \cdot a_4 - a_1 \cdot a_2 \cdot a_3 \cdot a_4 - a_1 \cdot a_2 \cdot a_3 \cdot a_5 - 2a_1 \cdot a_2 \cdot a_4 \cdot a_5 - a_2 \cdot a_3 \cdot a_4 \cdot a_5 + 2a_1 \cdot a_2 \cdot a_3 \cdot a_4 \cdot a_5$$

Заменив a_i на ВБР i -го элемента, получим:

$$P\{A\} = p_1 \cdot p_4 + p_2 \cdot p_5 + p_1 \cdot p_3 \cdot p_5 + p_2 \cdot p_3 \cdot p_4 - p_1 \cdot p_2 \cdot p_3 \cdot p_4 - p_1 \cdot p_2 \cdot p_3 \cdot p_5 - 2 \cdot p_1 \cdot p_2 \cdot p_4 \cdot p_5 - p_2 \cdot p_3 \cdot p_4 \cdot p_5 + 2 \cdot p_1 \cdot p_2 \cdot p_3 \cdot p_4 \cdot p_5$$

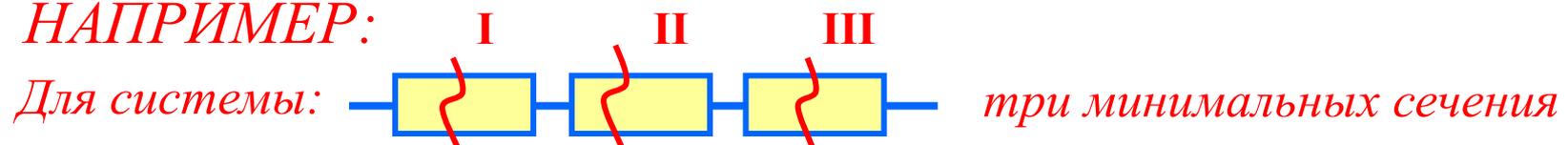
5.7.2. Метод минимальных сечений

ОПРЕДЕЛЕНИЕ

СЕЧЕНИЕ – совокупность элементов, **отказ** которых гарантирует отказ системы (неработоспособное состояние)

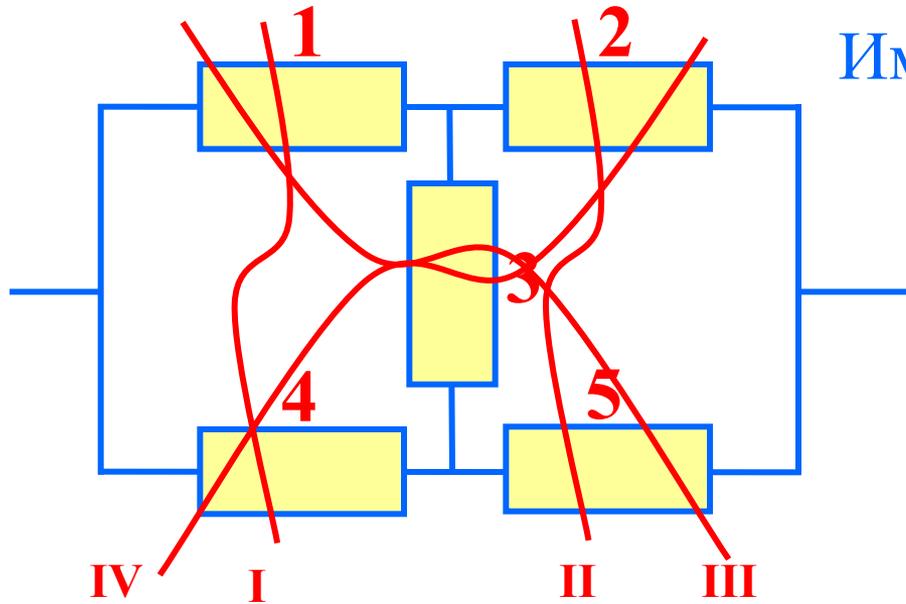
МИНИМАЛЬНОЕ СЕЧЕНИЕ – минимальный набор неработоспособных элементов, отказ которых приводит к отказу всей системы, а восстановление любого из них обеспечит работоспособность системы

НАПРИМЕР:



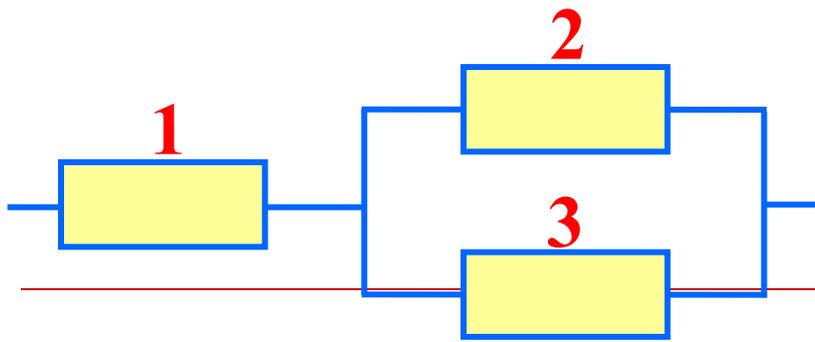
Для системы:

Рассмотрим мостиковую структуру



Имеем 4 минимальных сечения:

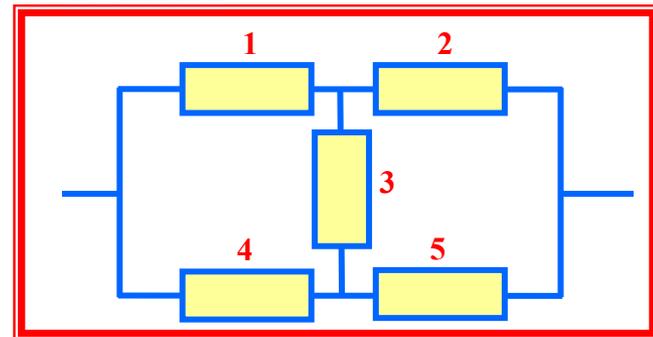
- 1: → 1 и 4
- 2: → 2 и 5
- 3: → 1 и 3 и 5
- 4: → 4 и 3 и 2



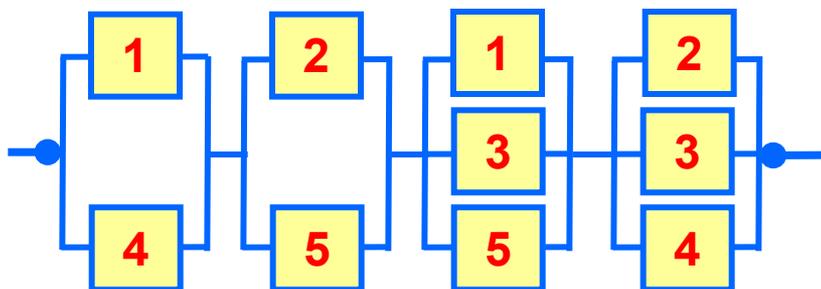
Сколько минимальных путей?

Правила построения ЛОГИЧЕСКОЙ СХЕМЫ методом минимальных сечений

1. Элементы каждого минимального сечения ставят параллельно
2. Все минимальные сечения ставят последовательно



ЛОГИЧЕСКАЯ СХЕМА ДЛЯ МОСТИКОВОЙ СТРУКТУРЫ:

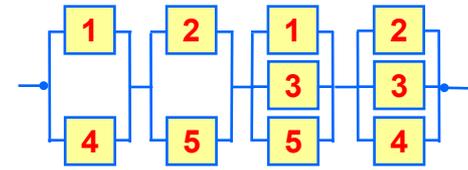


ЗДЕСЬ 2

неработоспособное состояние
элемента 2,
обозначим его переменной

$$\overline{a_2} = (1 - a_2)$$

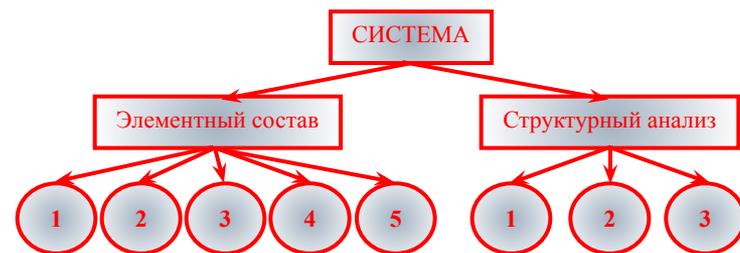
Для логической схемы составляем функцию алгебры логики.



Безотказная работа логической схемы заключается в безотказной работе всех последовательных участков. А отказ схемы – когда откажут элементы хотя бы одного сечения

$$\left[\begin{array}{l} \text{ОТКАЗ} \\ \text{СХЕМЫ} \end{array} \right] = \overline{A} = \left[\begin{array}{l} \overline{1} \text{ и } \overline{4} \quad \text{или} \quad \overline{2} \text{ и } \overline{5} \quad \text{или} \\ \overline{1} \text{ и } \overline{3} \text{ и } \overline{5} \quad \text{или} \quad \overline{2} \text{ и } \overline{3} \text{ и } \overline{4} \end{array} \right]$$

$$\begin{aligned} A &= 1 - \overline{a_1} \cdot \overline{a_4} \quad \text{и} \quad 1 - \overline{a_2} \cdot \overline{a_5} \quad \text{и} \quad 1 - \overline{a_1} \cdot \overline{a_3} \cdot \overline{a_5} \quad \text{и} \quad 1 - \overline{a_2} \cdot \overline{a_3} \cdot \overline{a_4} = \\ &= [1 - (1 - a_1) \cdot (1 - a_4)] \cdot [1 - (1 - a_2) \cdot (1 - a_5)] \times \\ &\quad \times [1 - (1 - a_1) \cdot (1 - a_3) \cdot (1 - a_5)] \cdot [1 - (1 - a_2) \cdot (1 - a_3) \cdot (1 - a_4)] \end{aligned}$$

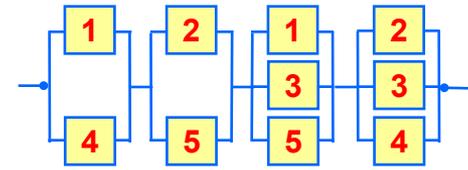


4. Элементы, отказы которых за рассматриваемый период времени могут привести к отказу вагона

5. Элементы, которые: имеют ограниченную контролепригодность при непосредственном использовании по назначению, могут привести к рушению поезда и для ремонта которых нужны стационарные условия вагонных депо

В расчётную схему «ВАГОН» нужно включать элементы 5 и 4 групп

Раскрывая скобки и используя законы алгебры логики, получим:



$$A = a_1 \cdot a_4 + a_2 \cdot a_5 + a_1 \cdot a_3 \cdot a_5 + a_2 \cdot a_3 \cdot a_4 - a_1 \cdot a_2 \cdot a_3 \cdot a_4 -$$

$$- a_1 \cdot a_2 \cdot a_3 \cdot a_5 - 2a_1 \cdot a_2 \cdot a_4 \cdot a_5 - a_2 \cdot a_3 \cdot a_4 \cdot a_5 + 2a_1 \cdot a_2 \cdot a_3 \cdot a_4 \cdot a_5$$

Заменив a_i на ВБР i -го элемента, получим:

$$P\{A\} = p_1 \cdot p_4 + p_2 \cdot p_5 + p_1 \cdot p_3 \cdot p_5 + p_2 \cdot p_3 \cdot p_4 - p_1 \cdot p_2 \cdot p_3 \cdot p_4 -$$

$$- p_1 \cdot p_2 \cdot p_3 \cdot p_5 - 2 \cdot p_1 \cdot p_2 \cdot p_4 \cdot p_5 - p_2 \cdot p_3 \cdot p_4 \cdot p_5 + 2 \cdot p_1 \cdot p_2 \cdot p_3 \cdot p_4 \cdot p_5$$

5.7.3. Метод разложения относительно базового элемента

Метод использует теорему алгебры логики о разложении функции логики по любому аргументу.

Согласно теореме:

$$P = p_i \cdot P(p_i=1) + q_i \cdot P(p_i=0)$$

ЗДЕСЬ

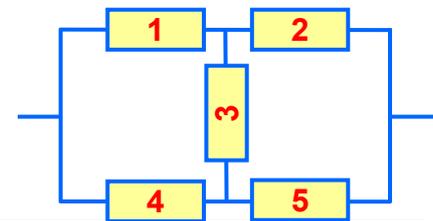
p_i – ВБР i -го элемента (работоспособного состояния)

q_i – вероятность отказа i -го элемента (неработоспособности)

$P(p_i=1)$ – ВБР схемы, при условии, что i -й элемент абсолютно надёжен (всегда в работоспособном состоянии)

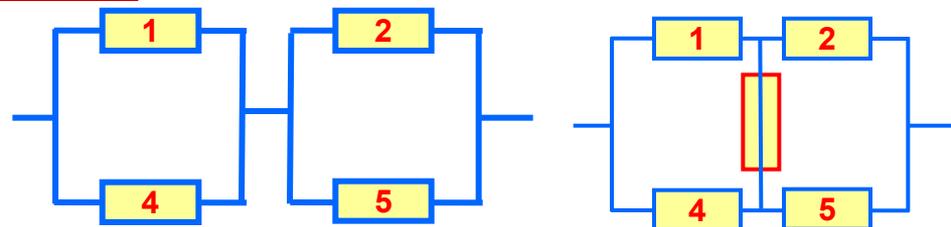
$P(p_i=0)$ – ВБР схемы, при условии, что i -й элемент абсолютно НЕнадёжен (всегда в НЕработоспособном состоянии)

Для мостиковой схемы в качестве базового целесообразно выбрать 3 элемент



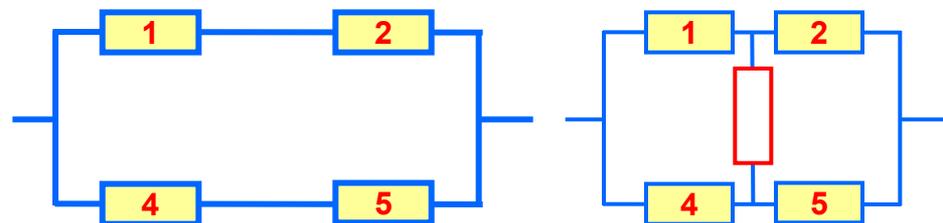
ТОГДА,
если $p_3 = 1$, получим схему:

$$P(p_3 = 1) = (p_1 * p_4) * (p_2 * p_5) = [1 - (1 - p_1)(1 - p_4)][1 - (1 - p_2)(1 - p_5)]$$



если $p_3 = 0$, получим схему:

$$P(p_3 = 0) = (p_1 p_2) * (p_4 p_5) = 1 - (1 - p_1 p_2)(1 - p_4 p_5)$$



ТОГДА по теореме:

$P =$

$$p_3 [1 - (1 - p_1)(1 - p_4)][1 - (1 - p_2)(1 - p_5)] + (1 - p_3) [1 - (1 - p_1 p_2)(1 - p_4 p_5)]$$

Для нескольких базовых элементов теорема имеет вид:

$$P = p_i p_j P[p_i = 1; p_j = 1] + p_i q_j P[p_i = 1; p_j = 0] + q_i p_j P[p_i = 0; p_j = 1] + q_i q_j P[p_i = 0; p_j = 0]$$

5.8. МЕТОД ДЕРЕВА ОТКАЗОВ (СОБЫТИЙ)

При разработке расчётной схемы технической системы для оценки их надёжности основная трудность в идентификации адекватной функции, т.е. выявлении связей двух типов: а) элемент – элемент; б) элемент – система. Особенно сложно решать задачу а. Метод дерева событий в сочетании с методом минимальных сечений позволяет формализовать и упростить процесс установления этих связей.

Ранее были рассмотрены т.н. двухполюсные сети (имеющие один «вход» и один «выход»), через которые «проходит продукт» – электрический ток, сжатый воздух, силовой поток, изделие и т.п. Будет рассмотрено другое представление монотонной структуры технической системы, имеющее несколько «входов» и один «выход» – дерево событий. Этот метод используется при анализе надёжности сложных систем.

Под **деревом событий** понимают знаковую форму логического сведения вершинного события (например, отказа системы)

к отказам элементов системам (элементарным, не разлагаемым событиям).

Логические связи между событиями, которые приводят к вершинному событию, отображаются в направленном графе, имеющем древовидную структуру.

После построения древовидной модели системы можно определить множество минимальных сечений в системе и оценить её надёжность

Правила построения :

1. Сложные события (состояния элементов) обозначают прямоугольником



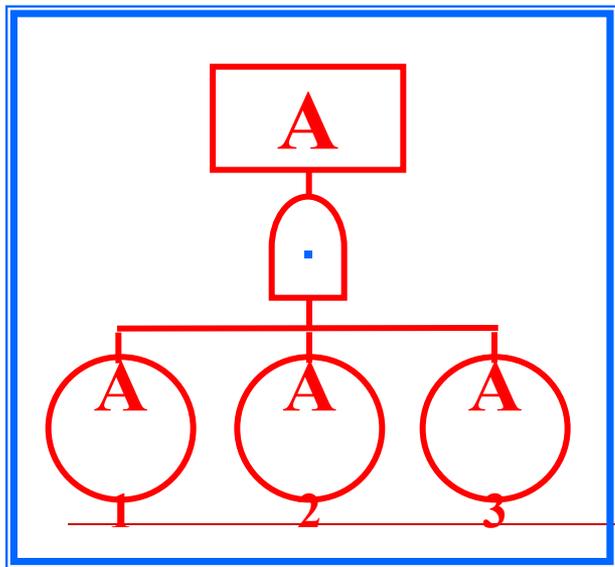
2. Элементарные, не разлагаемые события обозначают кругом



3. Связи между событиями отображают с помощью двух логических операторов:

Оператор **И** (совпадает с операцией \cap)

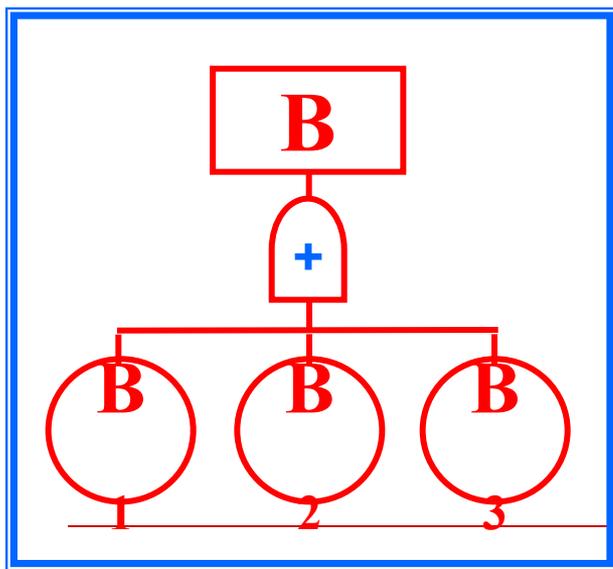
сигнал на выходе появляется только тогда, когда поступают все входные сигналы:



*Событие **A** может произойти лишь в случае одновременного появления событий A_1, A_2, A_3*

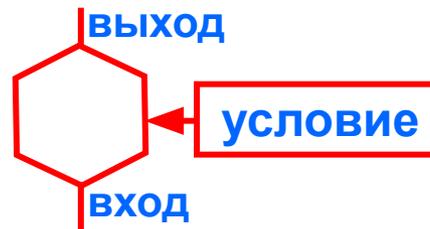
Оператор **ИЛИ** (совпадает с операцией **U**)

сигнал на выходе появляется при поступлении на входе хотя бы одного сигнала:



*Событие **V** может произойти когда на входе появится хотя бы одно из событий **V₁** или **V₂** или **V₃***

Оператор



означает, что сигнал на выходе
появляется только при выполнении
определённого условия

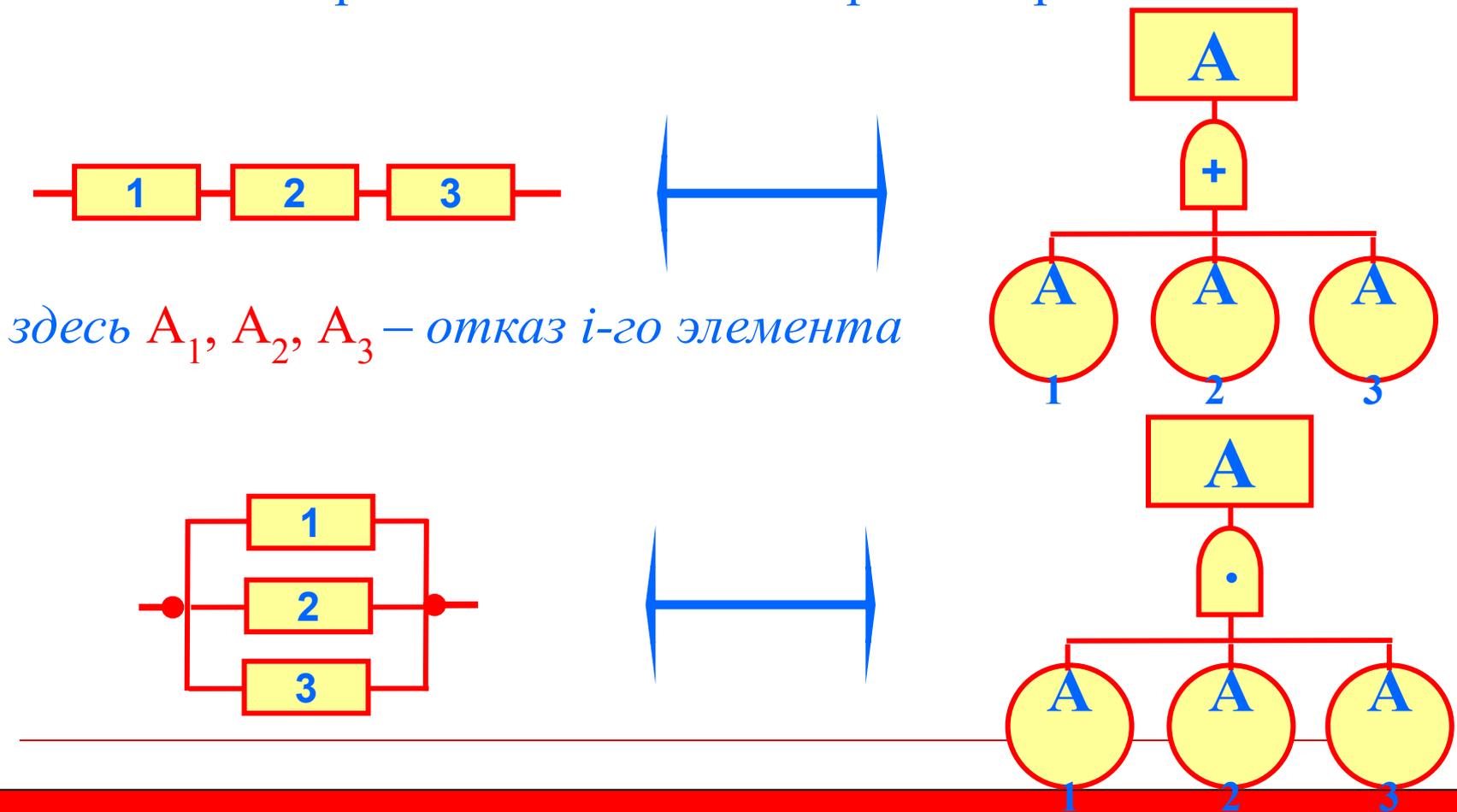
Между двухполюсным представлением структуры системы и древовидным существует однозначное соответствие

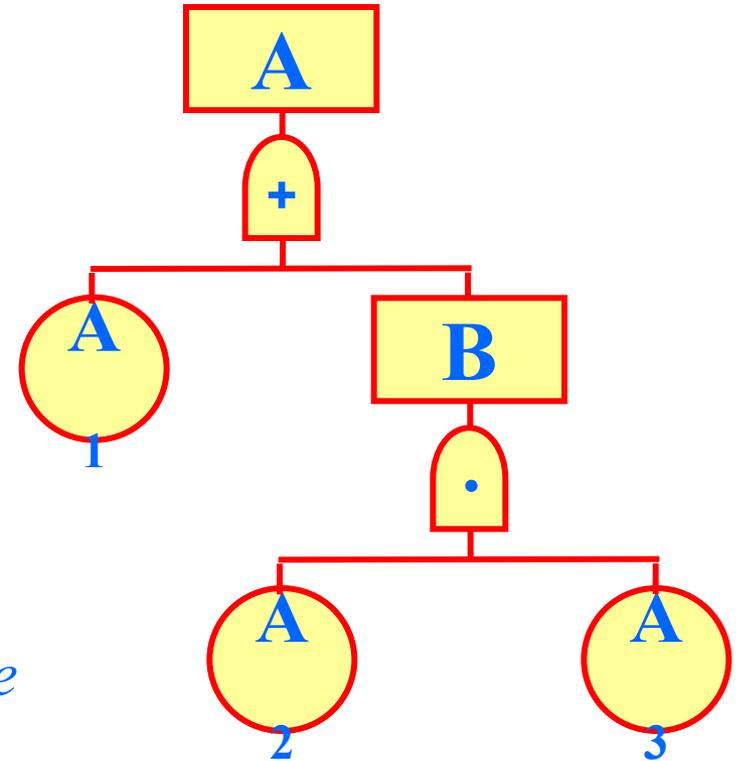
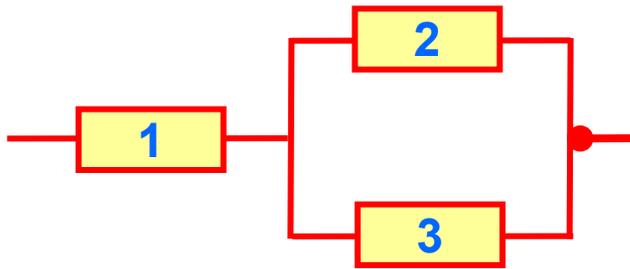
Это позволяет разбить процесс построения структурной функции сложной системы на два этапа:

1. Построение дерева событий (отказов)
 2. Переход к адекватной двухполюсной структуре
-

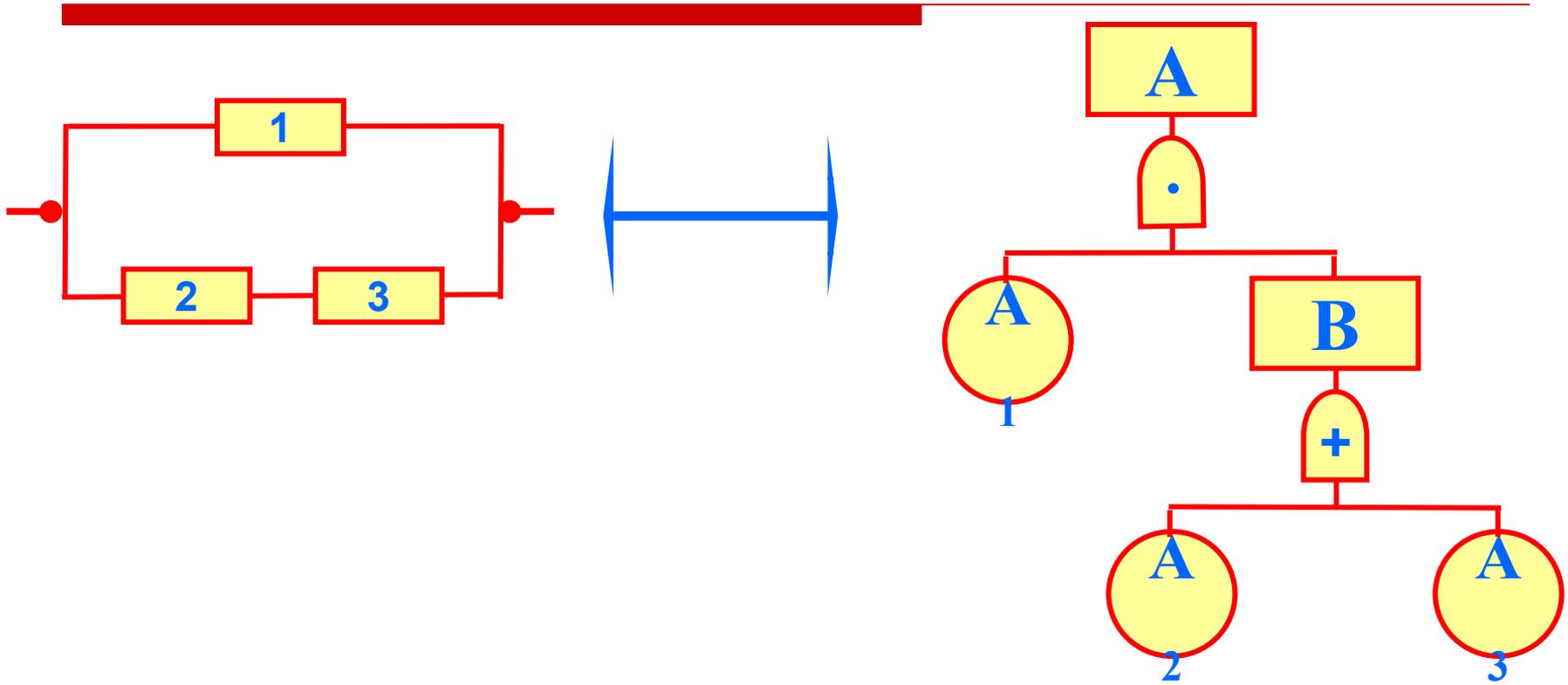
Рассмотрим взаимное соответствие между двумя представлениями на примере систем 3-го порядка

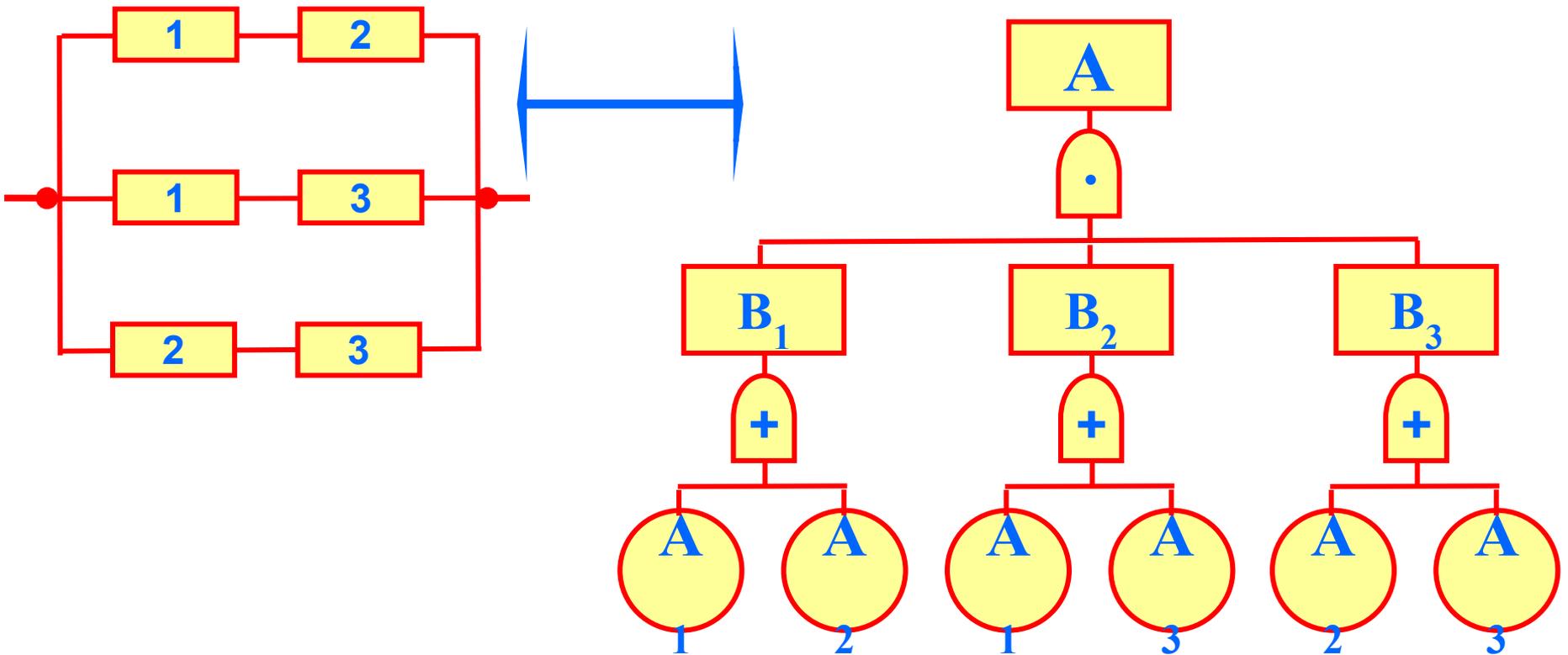
В качестве вершинного события A рассмотрим отказ системы





*здесь **B** – промежуточное событие*





здесь B_1, B_2, B_3 – i -е промежуточное событие

Основной целью построения дерева отказов является символическое представление существующих в системе условий, способных вызвать её отказ

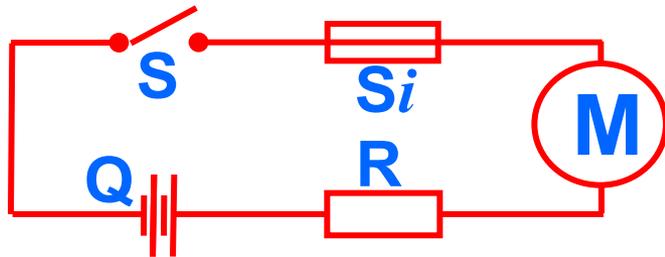
Древовидное представление структуры системы позволяет в явном виде показать слабые места системы

Процедура построения дерева событий:

1. Формулировка завершающего события
(например, что считается отказом системы)
 2. Детальное описание нормального процесса функционирования рассматриваемой системы
 3. Выяснение причин возникновения отказов элементов системы
 4. Построение дерева событий
для логически связанных событий
-

ПРИМЕР

Построим дерево СОБЫТИЙ для электродвигателя шуруповёрта



R – омическое сопротивление

S – выключатель

Si – предохранитель

Q – источник тока

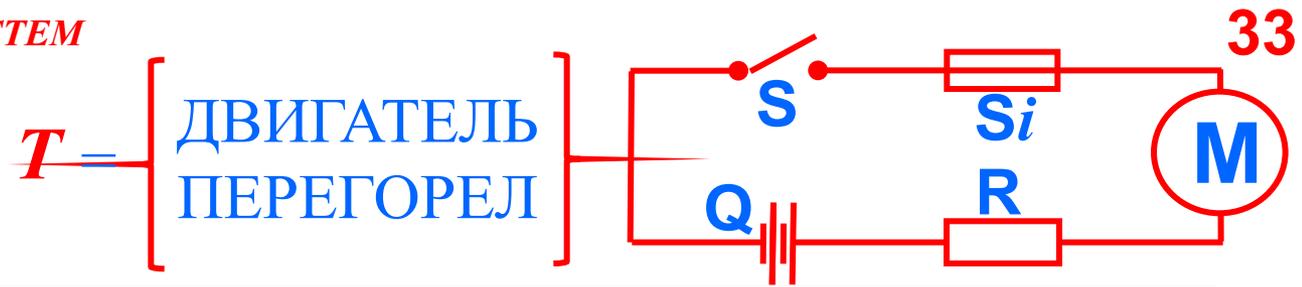
В качестве вершинного события рассмотрим событие T:

$$T = \left[\begin{array}{l} \text{ДВИГАТЕЛЬ} \\ \text{ПЕРЕГОРЕЛ} \end{array} \right]$$

Технология построения дерева связана с последовательным процессом ответов на вопросы.

ВОПРОС №1 Что должно произойти, чтобы наступило вершинное событие *T*?

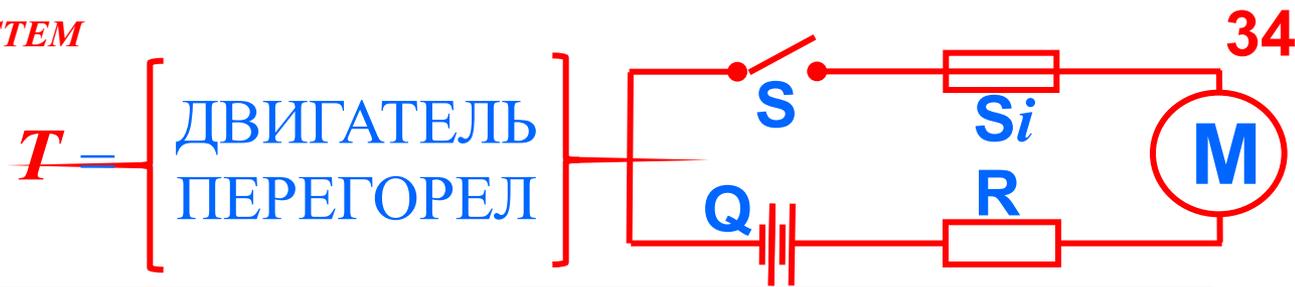
ОТВЕТ



Событие T может быть вызвано отказами, происходящими в расчётных условиях функционирования системы (т.н. первичным отказом двигателя, например, замыкание в обмотке, ошибка в подключении клемм), а может быть вызван неблагоприятным воздействием и завышенными нагрузками, приводящими к его перегреву (т.н. вторичным отказом)

ВОПРОС №2 Достаточно ли произойти одному из этих событий, чтобы наступило T ?

ОТВЕТ



Для наступления события T достаточно, чтобы произошло хотя бы одно из этих событий.

Тогда в древовидной модели указанные события соединены символом ИЛИ 

ВОПРОС №3 Требуется ли дальнейшее разложение события «первичный отказ двигателя»?

ОТВЕТ Первичный отказ двигателя детализировать не будем.

Тогда первичный отказ двигателя будем считать элементарным событием (не разлагаемым в рамках рассматриваемой задачи и обозначим его A_1)

ВОПРОС №4 Требуется ли дальнейшее разложение события «вторичный отказ двигателя»?

ОТВЕТ Да

С учётом ответов, имеем следующую структуру



Продолжаем детализировать событие «перегрев двигателя»

ВОПРОС №5 Что должно произойти, чтобы двигатель перегрелся?

ОТВЕТ К перегреву могут привести

– повышенное тепловыделение

– недостаточное охлаждение (повышенная температура окружающей среды)

ВОПРОС №6 Достаточно ли одного из событий или они должны наступить совместно?

ОТВЕТ Для перегрева двигателя достаточно одного из этих событий

Тогда эти события будут в схеме соединены оператором ИЛИ 

ВОПРОС №7 Нужно ли разложение события «повышенная температура окружающей среды»?

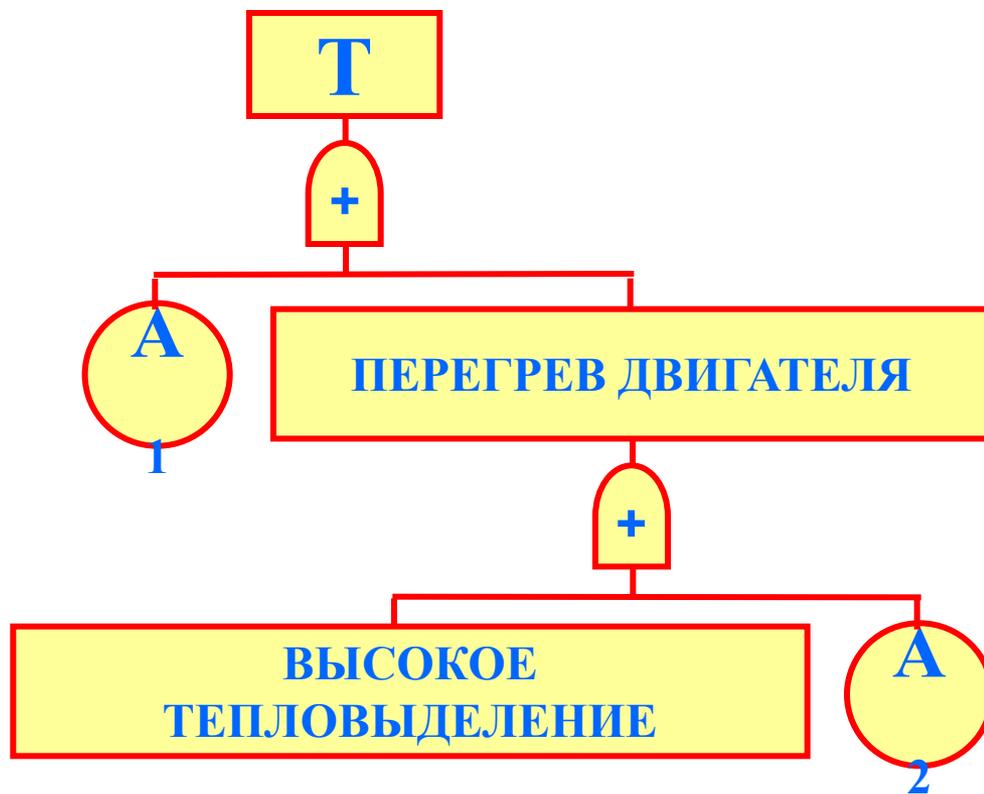
ОТВЕТ Нет

Тогда «повышенная температура окружающей среды» считаем неразлагаемым и обозначим A_2

ВОПРОС №8 Требуется ли дальнейшее разложение события «повышенное тепловыделение»?

ОТВЕТ Да

С учётом ответов, имеем следующую структуру



ВОПРОС №9 Какие причины повышенного тепловыделения?

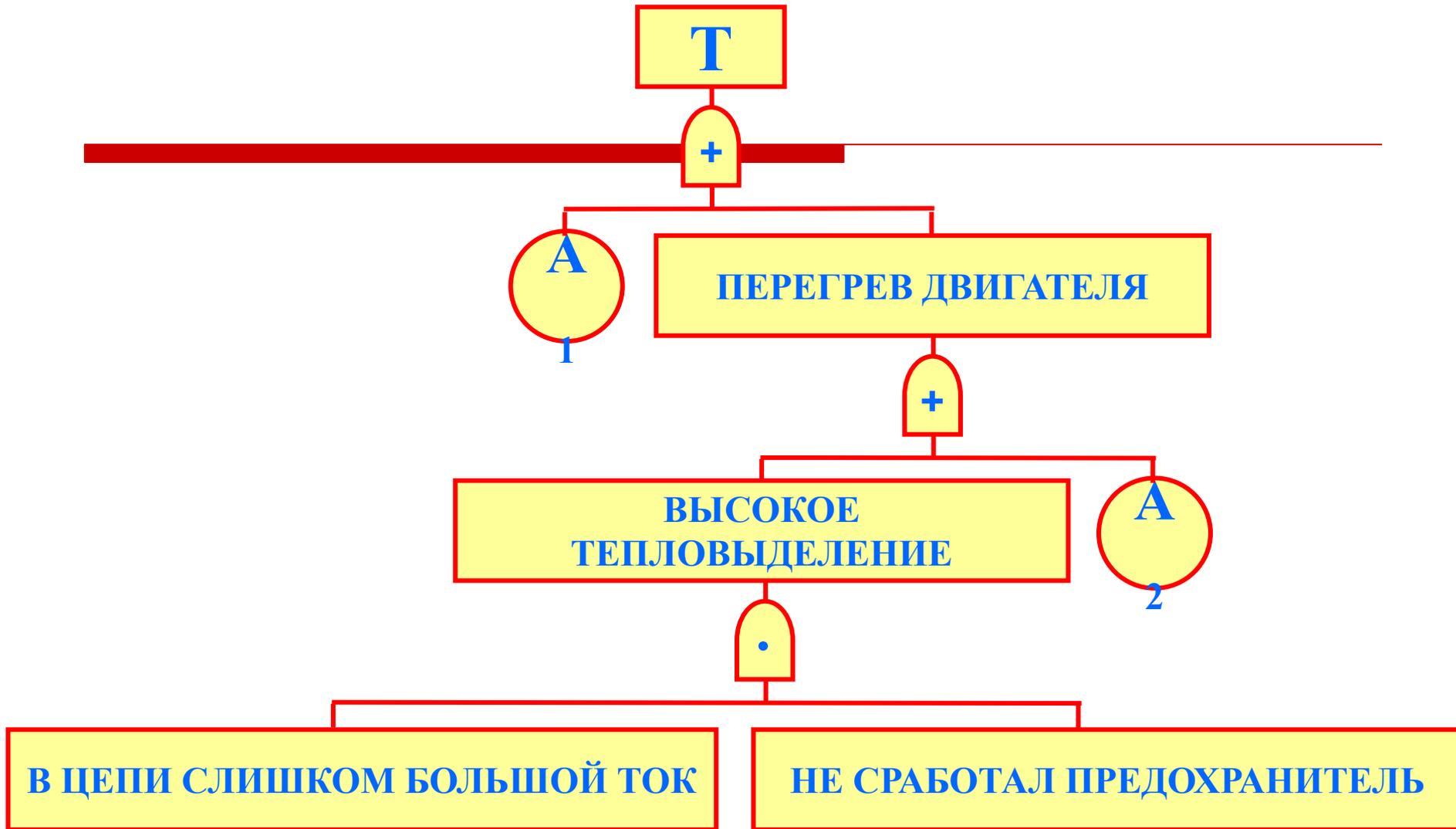
ОТВЕТ – по цепи течёт слишком большой ток
– не сработал предохранитель

ВОПРОС №10 Достаточно ли наступления хотя бы одного из них, чтобы тепловыделение стало критическим

ОТВЕТ Нет. Для этого нужно, одновременное наступление двух событий

Тогда эти события будут в схеме соединены оператором И





ВОПРОС №11 Нужно ли рассмотреть причины упомянутых событий

ОТВЕТ *Да*

ВОПРОС №12 Какие события приводят к повышению тока в цепи электродвигателя?

ОТВЕТ – *короткое замыкание (первичный отказ R)*
– *механический останов двигателя*
– *отказ источника тока (первичный отказ Q)*

ВОПРОС №13 Достаточно ли одного события для повышения тока в цепи электродвигателя?

ОТВЕТ *Да*

Тогда эти события будут в схеме соединены оператором

ИЛИ 

ВОПРОС №14 Нужно ли дальнейшее разложение этих событий

ОТВЕТ *Нет*

Тогда обозначим:

A_3 – первичный отказ R (короткое замыкание)

A_4 – механическая остановка двигателя

A_5 – первичный отказ источника тока (отказ Q)

ВОПРОС №15 Какие события приводят к тому, что предохранитель не сработал?

ОТВЕТ – предохранитель в неработоспособном состоянии (первичный отказ R)

– предохранитель не подходит по параметрам (или установлен «жучок»)

ВОПРОС №16 Достаточно ли одного из события, чтобы предохранительное устройство не среагировало на повышение тока в цепи?

ОТВЕТ *Да*

Тогда эти события будут в схеме соединены оператором **ИЛИ** 

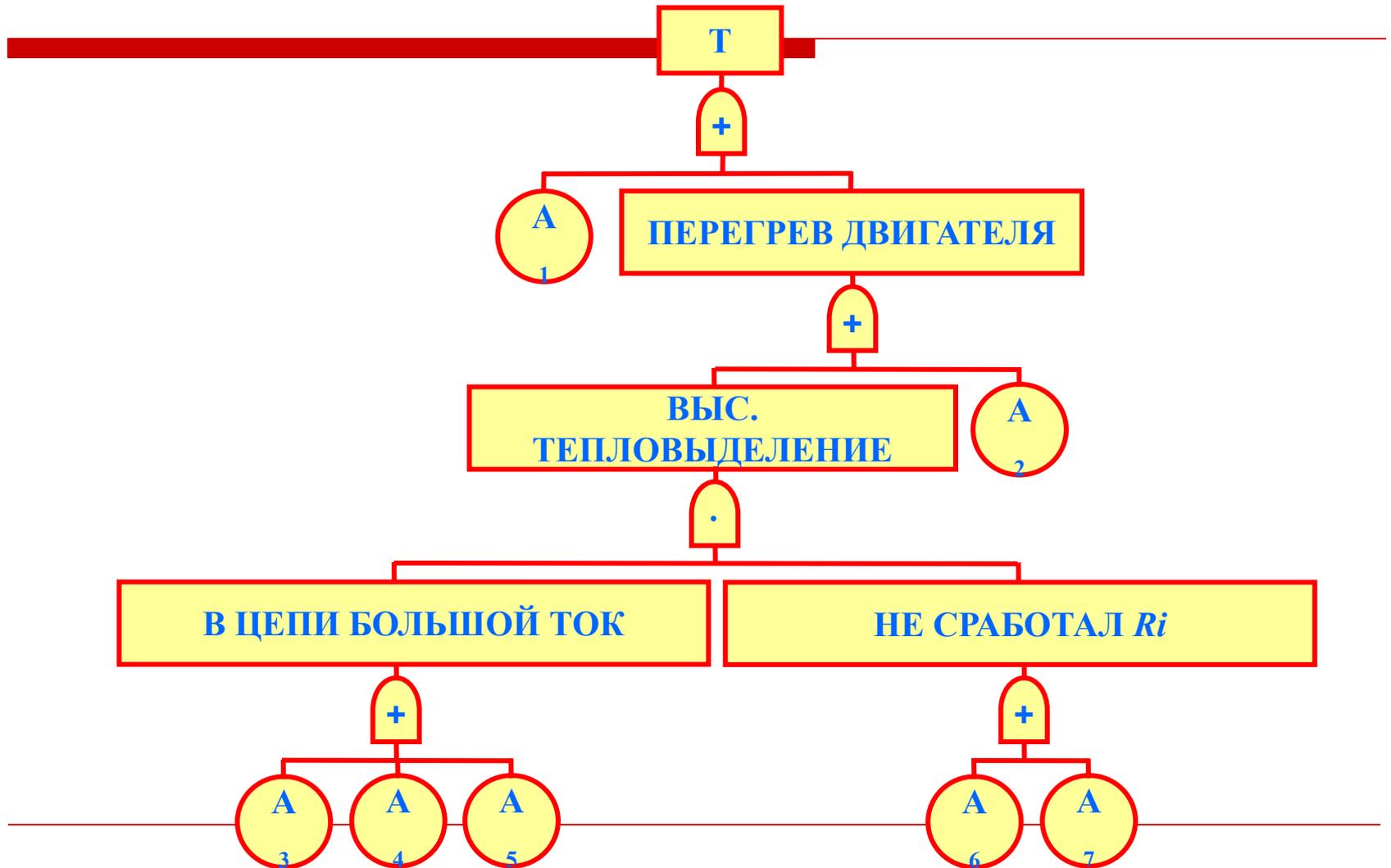
ВОПРОС №17 Требуется ли дальнейшее разложение этих событий?

ОТВЕТ *Нет*

Тогда обозначим:

A_6 – первичный отказ предохранителя R_i

A_7 – установлен не соответствующий предохранитель



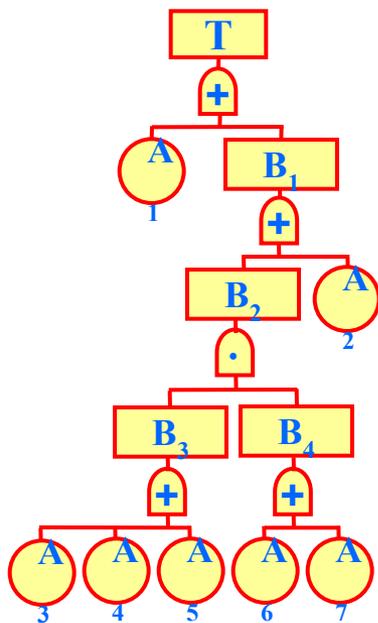
Замечание:

Дерево будет другим при ином вершинном событии

*Для получения количественных показателей надёжности
необходимо перейти от древовидного представления к
двухполюсному*

5.9. ПЕРЕХОД ОТ ДРЕВОВИДНОЙ МОДЕЛИ СИСТЕМЫ К ДВУХПОЛЮСНОЙ

Рассмотрим технологию перехода от древовидной модели системы к двухполюсной на примере дерева для электродвигателя



Переход осуществляется с помощью метода минимальных сечений

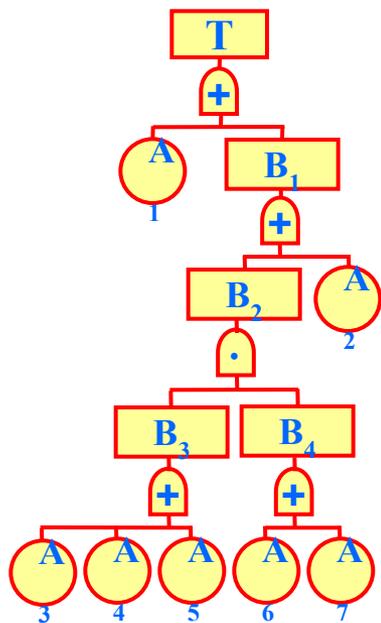
Список основных событий (элементов), принадлежащих ко множеству минимальных сечений, имеет матричную структуру, которая строится согласно следующему мнемоническому правилу:

Если промежуточным или вершинным событием управляет оператор «ИЛИ», то заменяем это событие на входные элементы (события) отдельным столбцом

Если промежуточным или вершинным событием управляет оператор «И», то заменяем это событие на входные события в виде отдельной строки

Поиск минимального сечения начинаем с вершинного события T

Вершинным событие управляет оператор ИЛИ
 Поэтому заменяем событие T на входящие события A_1 и B_1 в виде столбца

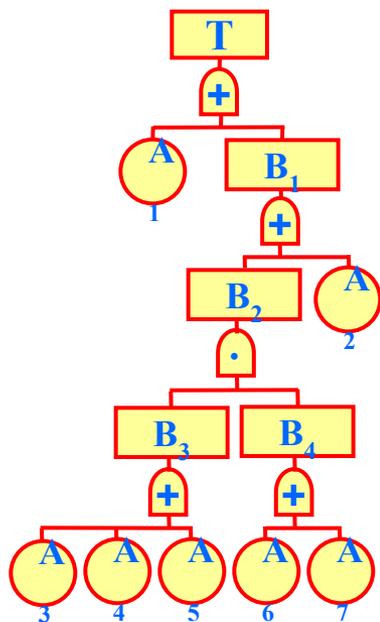


$$T = \left| \begin{array}{c} A_1 \\ B_1 \end{array} \right| =$$

событием B_1 управляет оператор ИЛИ,
 заменяем его на входные элементы B_2 и A_2 в виде столбца

$$= \left| \begin{array}{c} A_1 \\ B_2 \\ A_2 \end{array} \right| =$$

событием V_2 управляет оператор И,
заменяем его на входные элементы
 V_3 и V_4 в виде строки

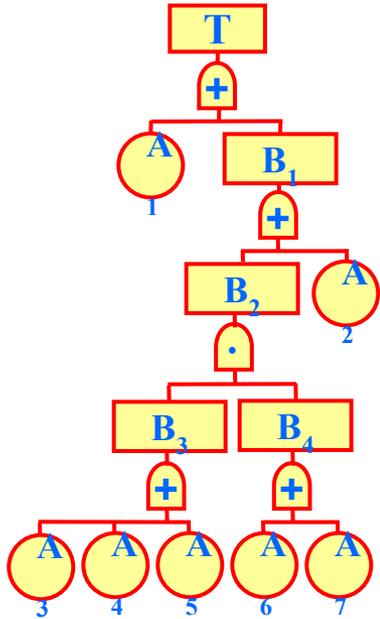


$$= \left| \begin{array}{c|c} A_1 & 0 \\ \hline V_3 & V_4 \\ A_2 & 0 \end{array} \right| =$$

событием V_3 управляет оператор ИЛИ,
заменяем его на входные элементы
 A_3, A_4 и A_5 в виде столбца

$$= \left| \begin{array}{c|c} A_1 & 0 \\ \hline A_3 & V_4 \\ A_4 & V_4 \\ A_5 & V_4 \\ A_2 & 0 \end{array} \right| =$$

событием B_4 управляет оператор ИЛИ,
 заменяем каждое на входные элементы
 A_6 и A_7 в виде столбца

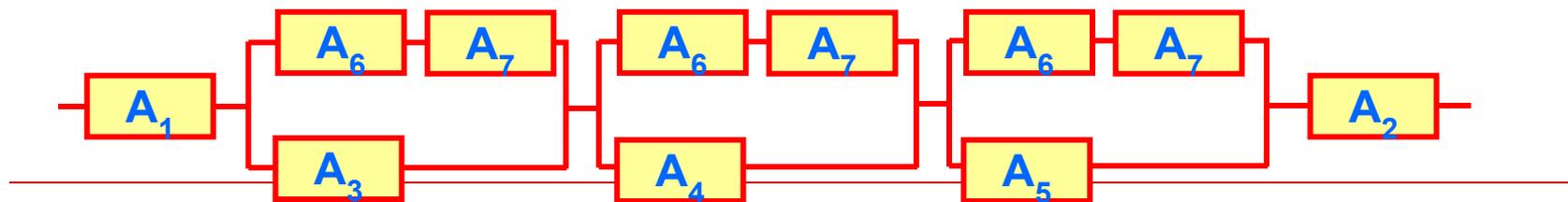


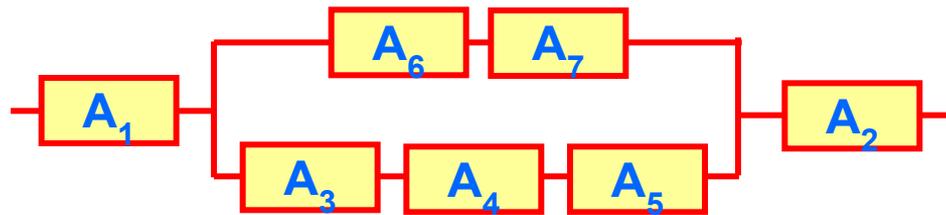
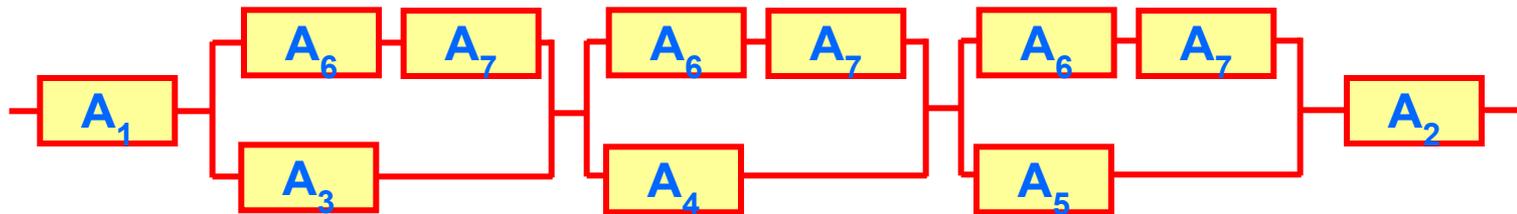
$$= \begin{vmatrix} A_1 & 0 \\ A_3 & B_4 \\ A_4 & B_4 \\ A_5 & B_4 \\ A_2 & 0 \end{vmatrix} = \begin{vmatrix} A_1 & 0 \\ A_3 & A_6 \\ A_3 & A_7 \\ A_4 & A_6 \\ A_4 & A_7 \\ A_5 & A_6 \\ A_5 & A_7 \\ A_2 & 0 \end{vmatrix}$$

$$T = \begin{array}{|l} A_1 \ 0 \\ A_3 \ A_6 \\ A_3 \ A_7 \\ A_4 \ A_6 \\ A_4 \ A_7 \\ A_5 \ A_6 \\ A_5 \ A_7 \\ A_2 \ 0 \end{array}$$

Каждая строка матрицы – есть минимальное сечение

Таким образом получили последовательно-параллельную структуру системы:

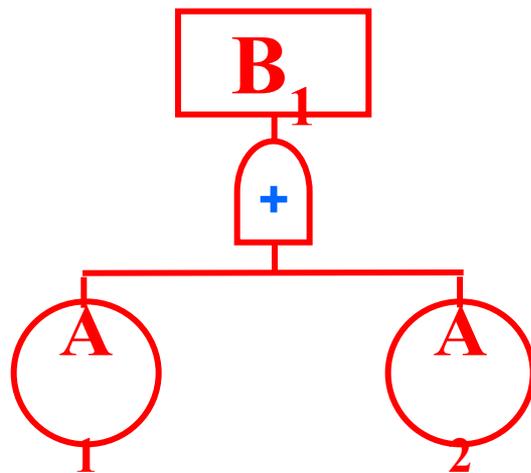




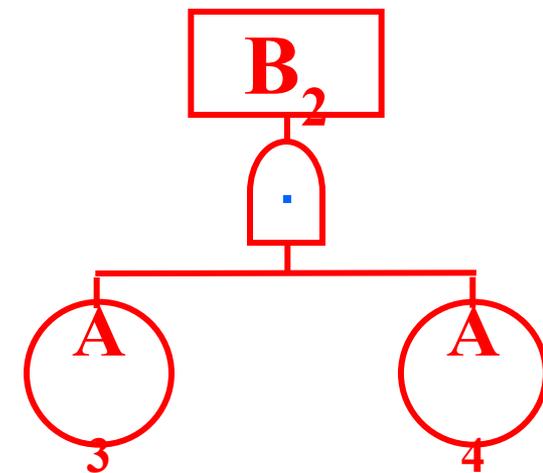
Получить выражение для ВБР системы (P_c)

5.10. ИСПОЛЬЗОВАНИЕ АЛГЕБРЫ ЛОГИКИ ДЛЯ ДРЕВОВИДНОЙ СТРУКТУРЫ СИСТЕМЫ

Существует ещё один способ получения количественных показателей надёжности на основе дерева отказов — это непосредственное представление схемы в математической форме с помощью основных законов булевой алгебры. Рассмотрим простейшие схемы:



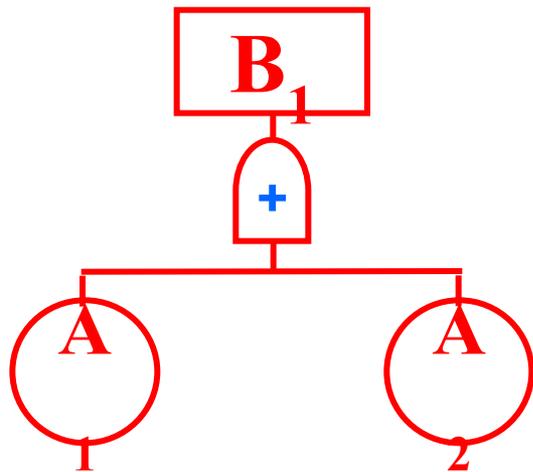
Логический
оператор **ИЛИ**



Логический
оператор **И**

Рассматриваем простейшие схемы:

Логический оператор **ИЛИ**



Событие B_1 – отказ с помощью булевой алгебры можно представить с помощью выражения:

$$B_1 = A_1 + A_2$$

Вероятность появления отказа (завершающего события) B_1 :

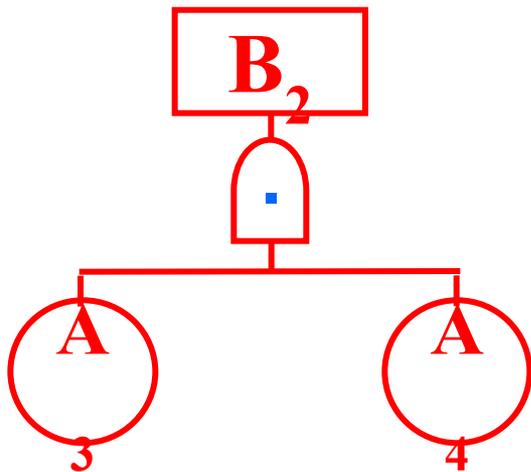
$$P\{B_1\} = P\{A_1\} + P\{A_2\} - P\{A_1 \cdot A_2\}$$

Если отказы A_1 и A_2 не зависимы и вероятность их произведения очень мала, то можно приближённо записать:

$$P\{B_1\} \approx P\{A_1\} + P\{A_2\}$$

Рассматриваем простейшие схемы:

Логический оператор И



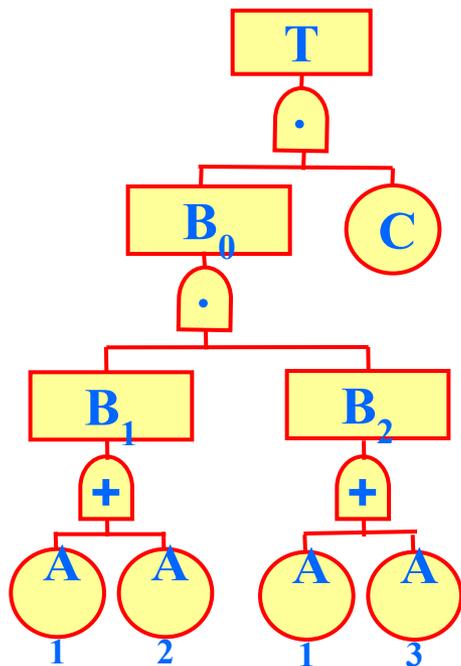
Событие B_2 – отказ с помощью булевой алгебры можно представить с помощью выражения:

$$B_2 = A_3 \cdot A_4$$

Если отказы A_3 и A_4 не зависимы, то:

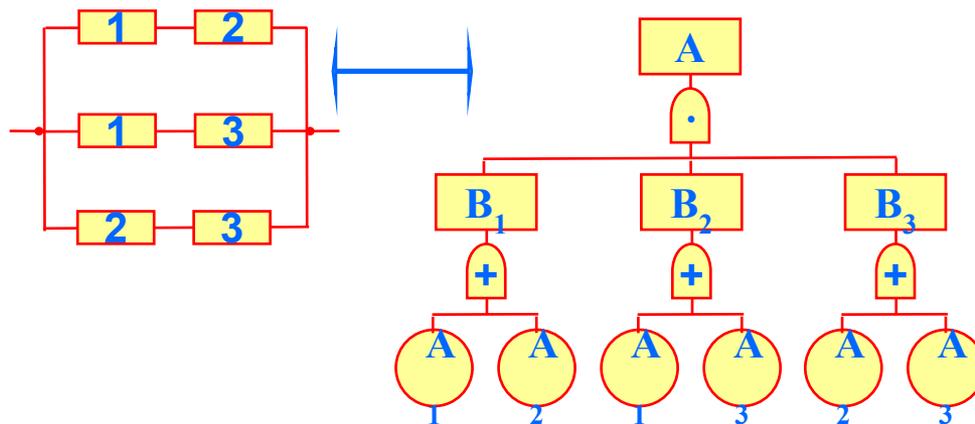
$$P\{B_2\} = P\{A_3\} \cdot P\{A_4\}$$

Рассмотрим сложную древовидную структуру:



Получим выражение для завершающего события T

При этом нужно, чтобы дерево не имело повторяющихся событий, как например в случае:



С помощью булевой алгебры можно перейти от дерева с повторяющимися событиями к эквивалентному дереву без повторяющихся событий

Воспользуемся

законами булевой алгебры:

$$(Y+Z) \cdot X = X \cdot Y + X \cdot Z$$

$$X + (Y \cdot Z) = (X + Y) \cdot (X + Z)$$

булевыми тождествами:

$$X + X = X$$

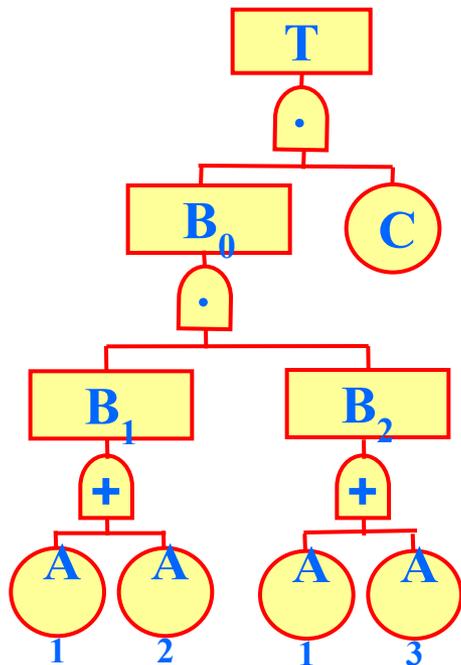
$$X \cdot X = X$$

законами поглощения:

$$X + (X \cdot Y) = X$$

$$(X \cdot Y) \cdot X = X \cdot Y$$

Представим событие T с помощью булевых выражений:



$$T = C \cdot B_0$$

$$B_0 = B_1 \cdot B_2$$

$$B_1 = A_1 + A_2$$

$$B_2 = A_1 + A_3$$

Тогда для события T можно записать:

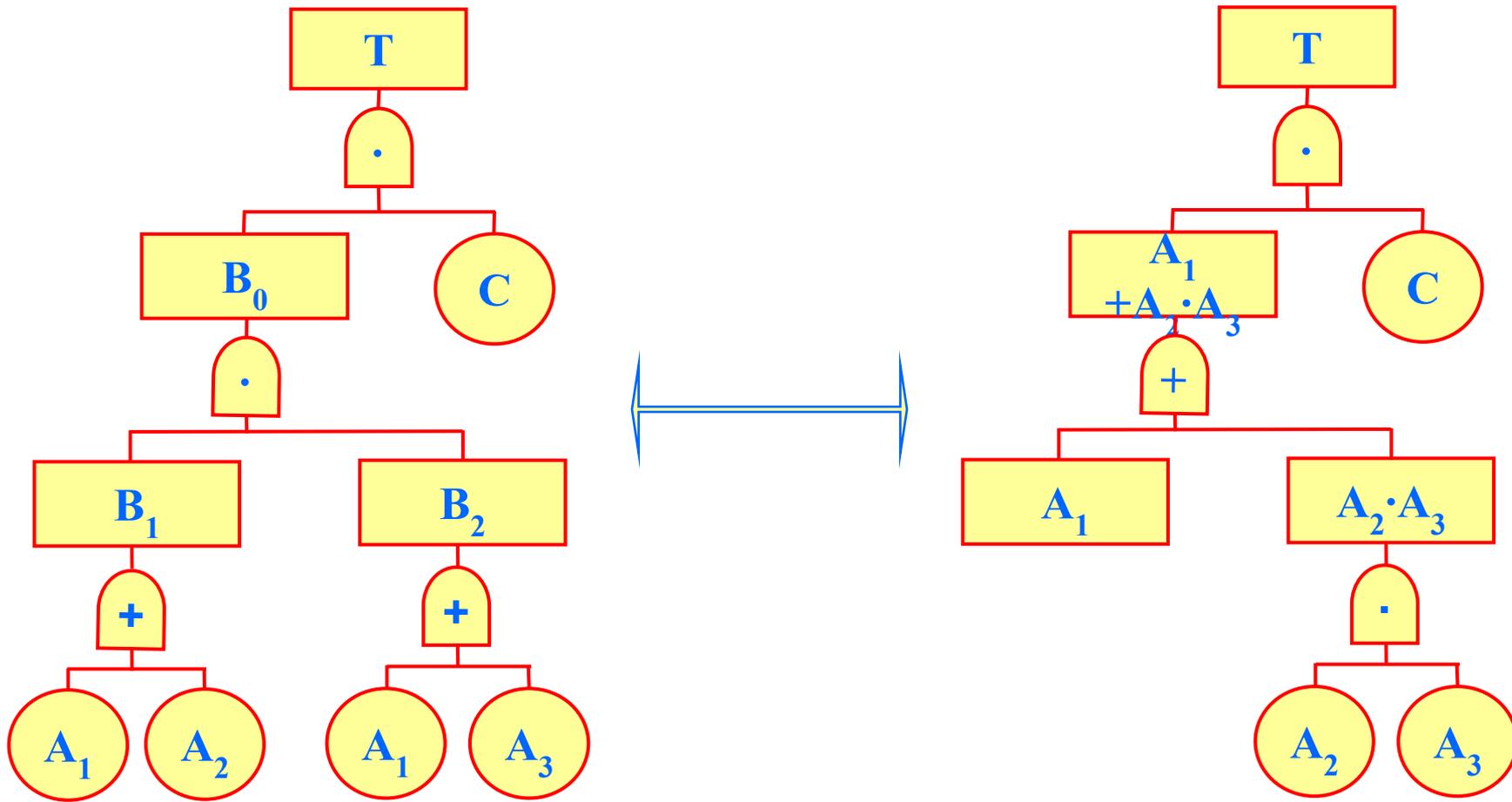
$$T = C \cdot (A_1 + A_2) \cdot (A_1 + A_3)$$

Упростим выражение с помощью распределительного закона:

$$T = C \cdot (A_1 + A_2 \cdot A_3)$$

Δ

ТАКИМ ОБРАЗОМ, ПЕРВОНАЧАЛЬНОЕ ДЕРЕВО С ПОВТОРЯЮЩИМИСЯ СОБЫТИЯМИ ПРИВЕЛИ К СЛЕДУЮЩЕМУ ВИДУ:



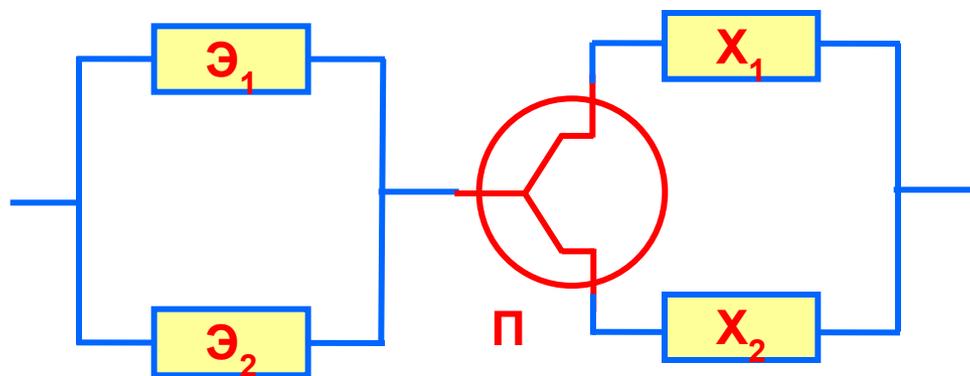
С помощью булевой алгебры получают модели состояния системы по состояниям её элементов в некоторый момент времени, т.е. для мгновенных состояний элементов и системы

При этом считают, что временная последовательность отказов не имеет значения. Однако, в некоторых случаях эта последовательность имеет значение. Например:

автоматизированная система автономного
рефрижераторного вагона (АРВ)

Охлаждение обеспечивает одна из двух холодильных установок. При её отказе, который должен быть обнаружен при ТО-2 на станции в пути следования, осуществляется переключение на резервный холодильный агрегат.

Структурная схема автоматизированной системы АРВ имеет вид



П – переключатель, который приводится в действие обслуживающим персоналом,

Э₁, Э₂ – источники энергоснабжения,

X₁, X₂ – основной и резервный холодильный агрегат.

Если в начале откажет X₁, то последующий отказ переключателя П не приведёт к отказу АРВ.

Если в начале откажет переключатель П, то последующий отказ X₁ приведёт к отказу АРВ

Невозможность учитывать очерёдность наступления отказов элементов системы является недостатком булевых моделей надёжности систем.

Поэтому получил развитие инженерный метод построения моделей надёжности систем с учётом очерёдности отказов элементов на основе дерева отказов с использованием **ГРАФОВ**
– ГРАФОВЫЙ МЕТОД
