Методы устранения РС в КС

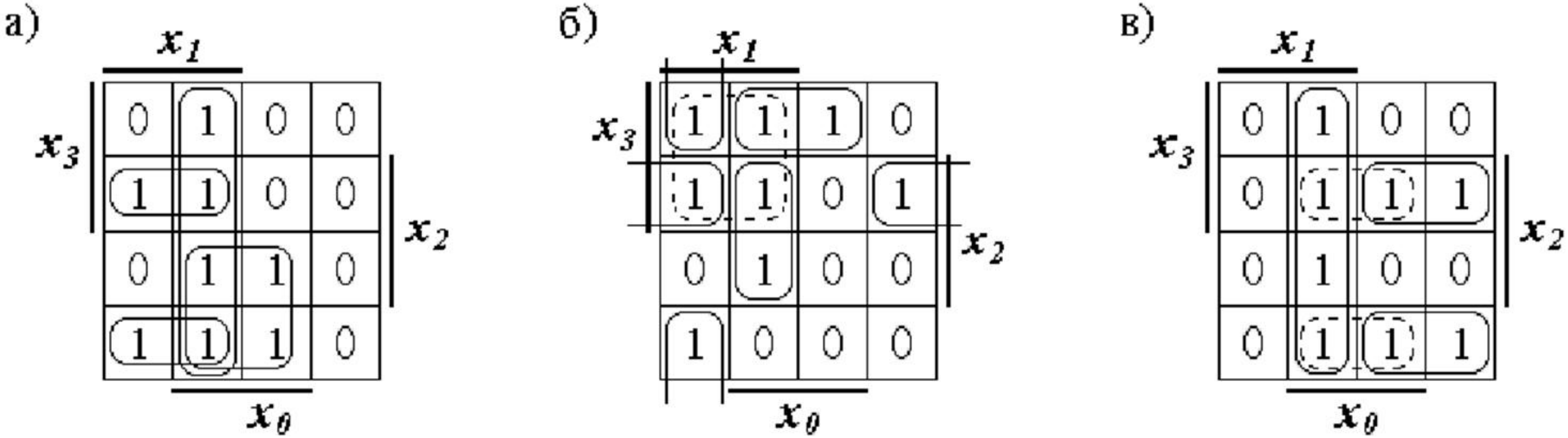
Структурные. Функциональные. Конструктивно-технологические. Рекомендации по устранению РС. Примеры.

Введение

- *Все методы, разработанные для устранения рисков сбоя в комбинационных схемах, можно объединить в три группы:* ***Структурные методы*** направлены на получение необходимых свойств реализации устройства при неизменном алгоритме его работы.
- ***Функциональные методы*** связаны с изменением алгоритма работы, в частности с изменением кодирования состояний входов.
- ***Конструктивно-технологические методы*** ориентированы на получение требуемых ограничений на уровне используемых математических моделей.
- ***Наиболее простыми для соответствующей математической модели являются структурные и функциональные методы**, **а наиболее сложными конструктивно-технологические**, **так как они часто связаны с разработкой принципиально новых видов производства интегральных цифровых схем.***

Структурные методы

- На рис. показаны карты Карно для функций, свободных от статических рисков сбоя S_1, при переходах между ***любыми соседними единичными клетками***.

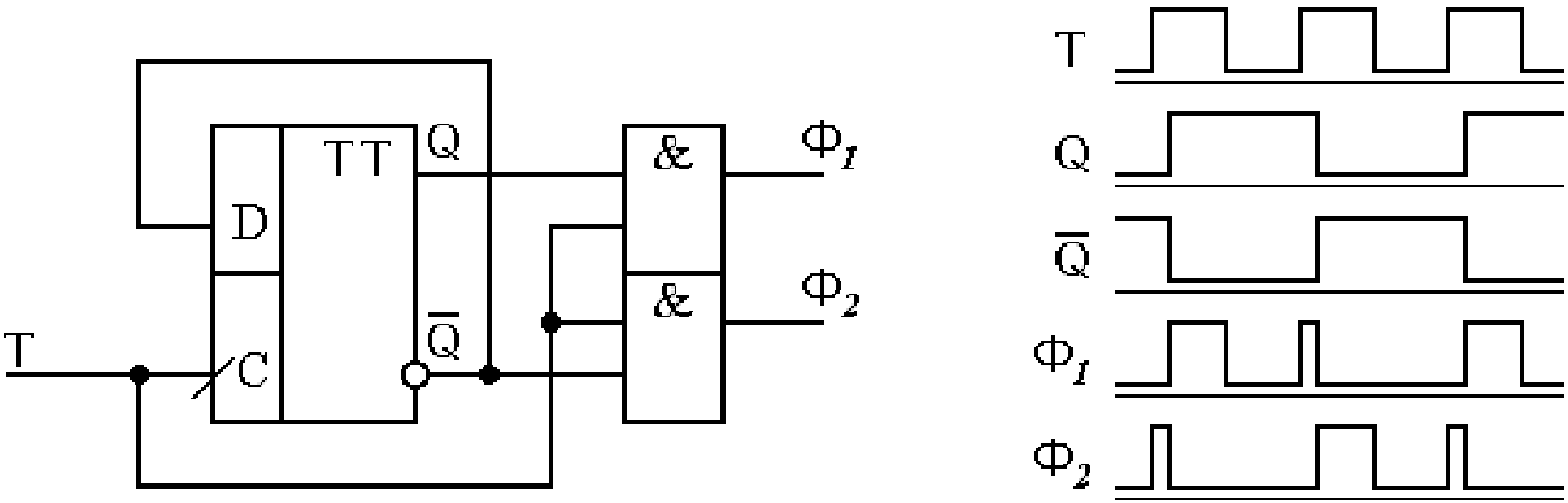


- Функция в случае **а**) не требует дополнительных импликант, так как все смежные контуры покрыты другими контурами. А в случаях **б**) и **в**) введены контуры, показанные штриховой линией.
- Если же допускаются не соседние изменения входных наборов, то в общем случае невозможно синтезировать **комбинационные схемы**, свободные от **рисков сбоя** (см. раздел “Функциональный риск сбоя”).

Структурные методы

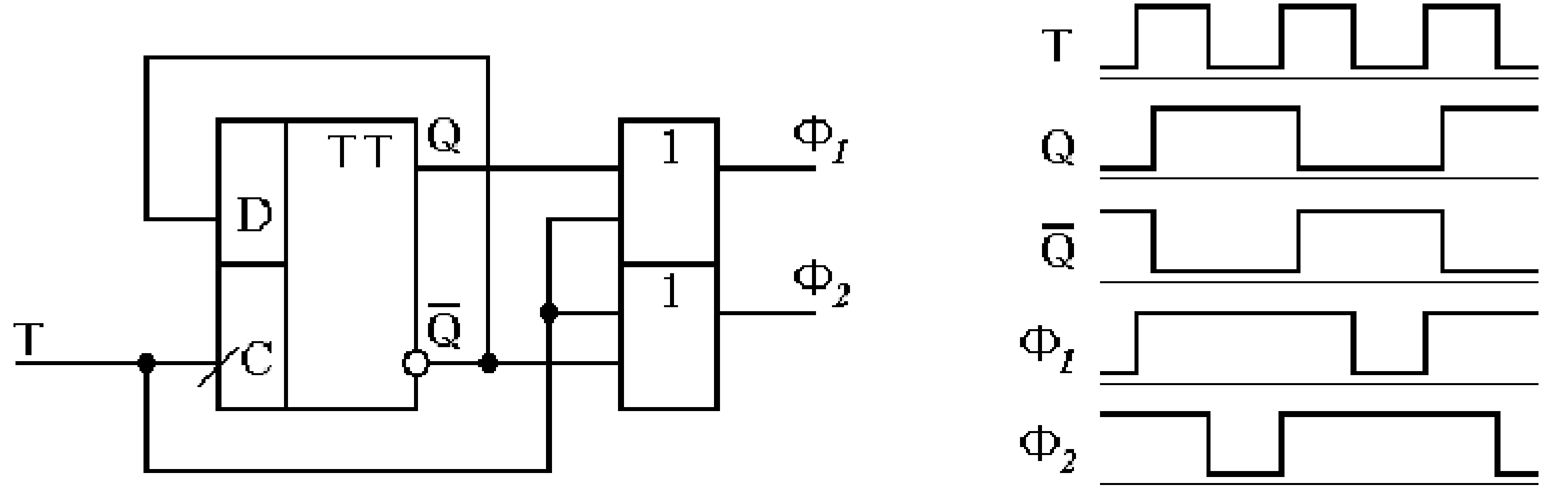
Причиной риска сбоя может быть ***неправильно спроектированная схема***. Примером такой схемы является генератор двухфазной системы синхронизации (рис. *а*). Задержка переключения триггера в данном случае приводит к появлению статических рисков сбоя S_0 на выходе при переходе с набора на набор и на выходе при переходе с набора на набор. Устранить эти сбои можно ***схемотехническими (структурными) методами***.

а)

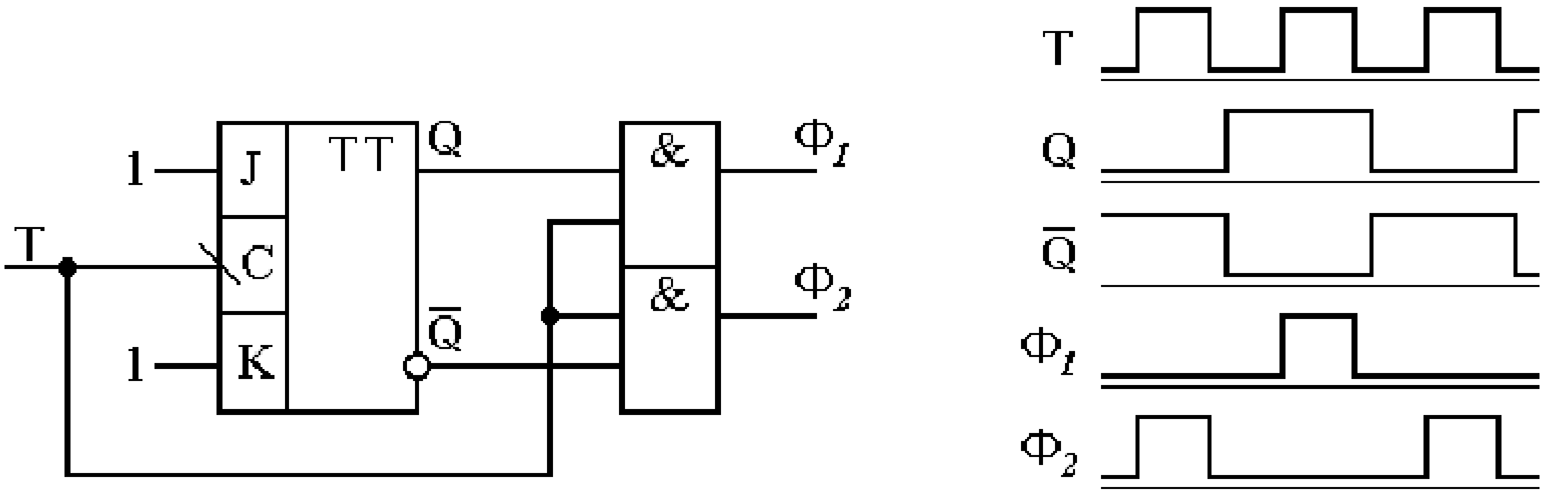


Структурные методы

б)



в)



Универсальные методы устранения РС

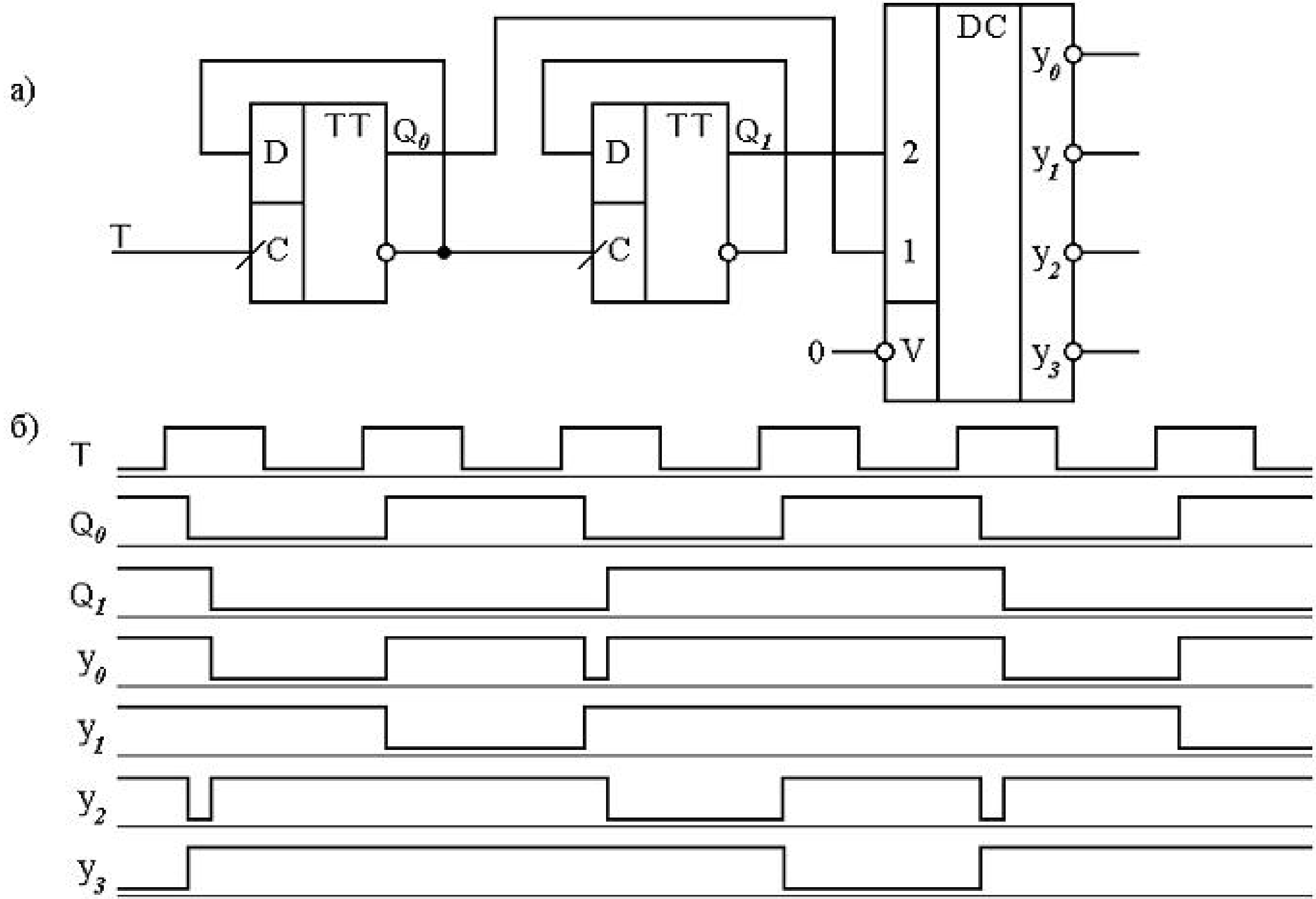
Наиболее универсальными и поэтому широко используемыми методами борьбы с рисками сбоя являются ***тактирование*** и ***стробирование***.

I Суть ***тактирования*** заключается в следующем. По всему цифровому устройству разводится единая система тактирующих (синхронизирующих) сигналов, обеспечивающих ***запись информационных данных в регистры*** через время, которое превышает самый длинный процесс неопределенности, то есть самую большую задержку во всех трактах схемы.

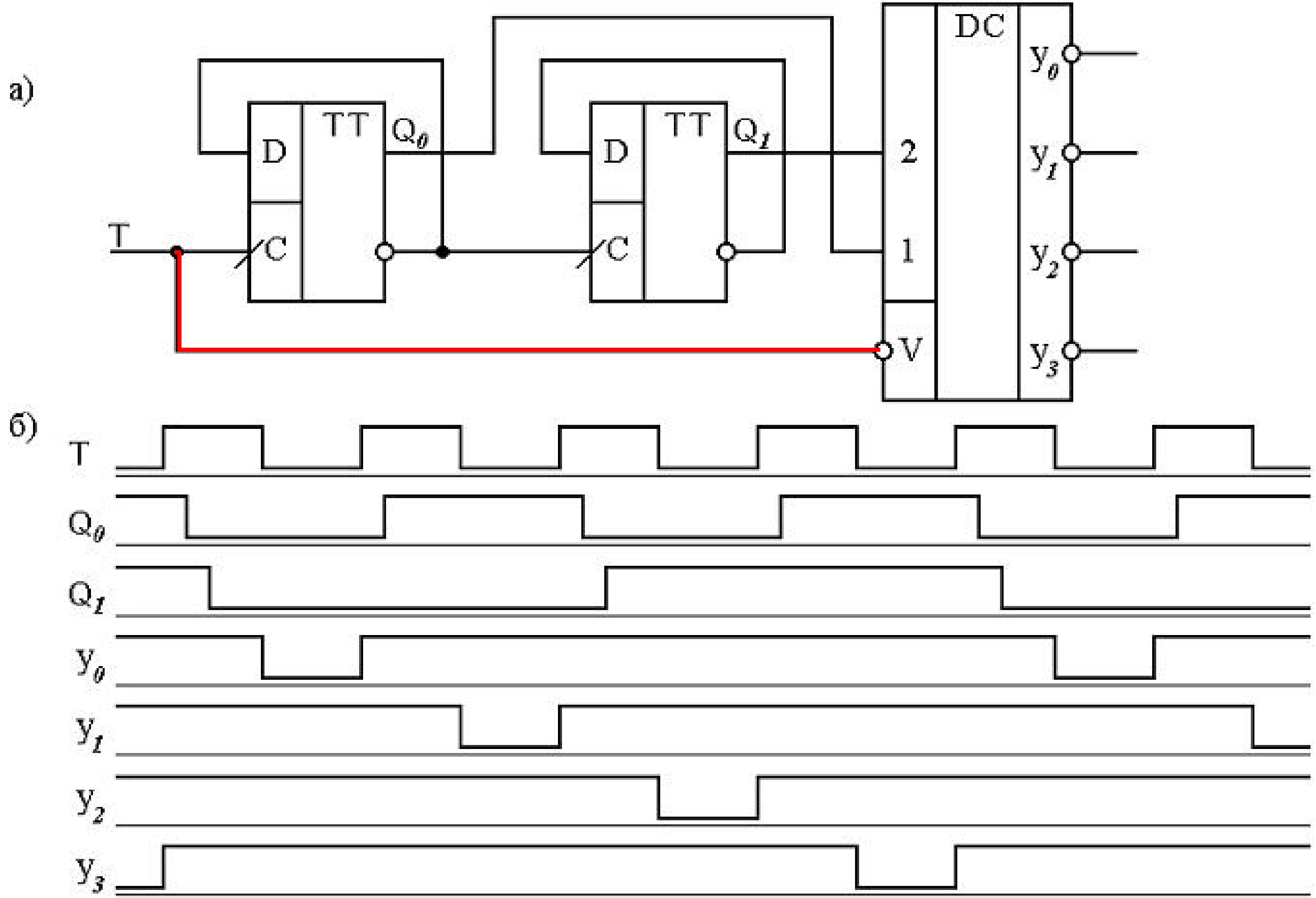
II Если же необходимо лишь ***очистить*** сигнал от рисков сбоя, а не запомнить его, то используется ***метод стробирования***, реализуемый соответствующим построением комбинационной схемы.

Достоинством тактирования и стробирования является то, что разработчику не требуется вникать в специфику протекания переходных процессов, в характер возникающих гоночных ситуаций, не нужно знать минимального значения задержки и т. д. Все, что должен знать разработчик, - это максимально возможную задержку самого длинного тракта логической схемы, а это легко вычисляется по паспортным данным используемых элементов.

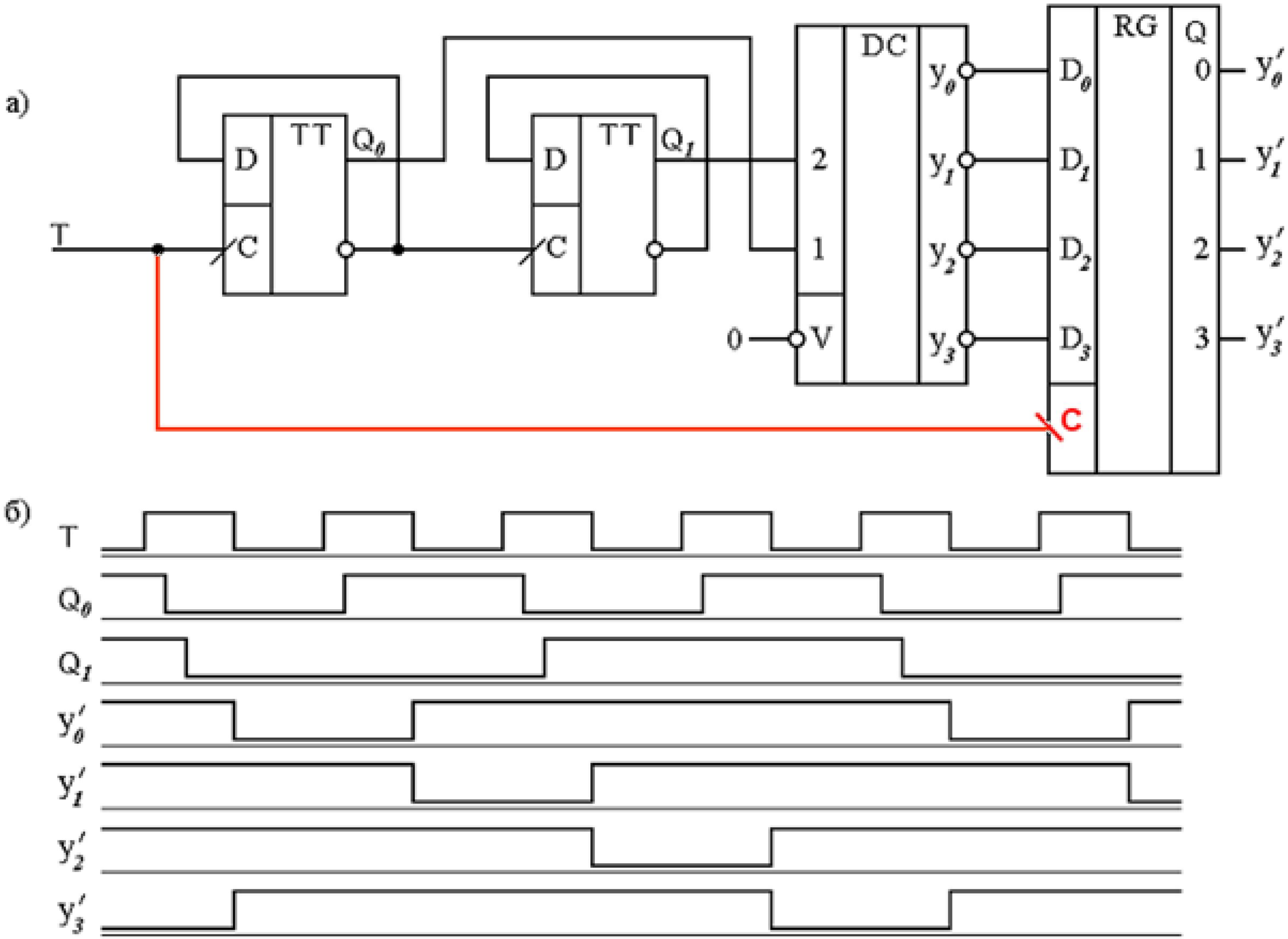
Универсальные методы устранения РС



Метод стробирования.



Метод тактирования.

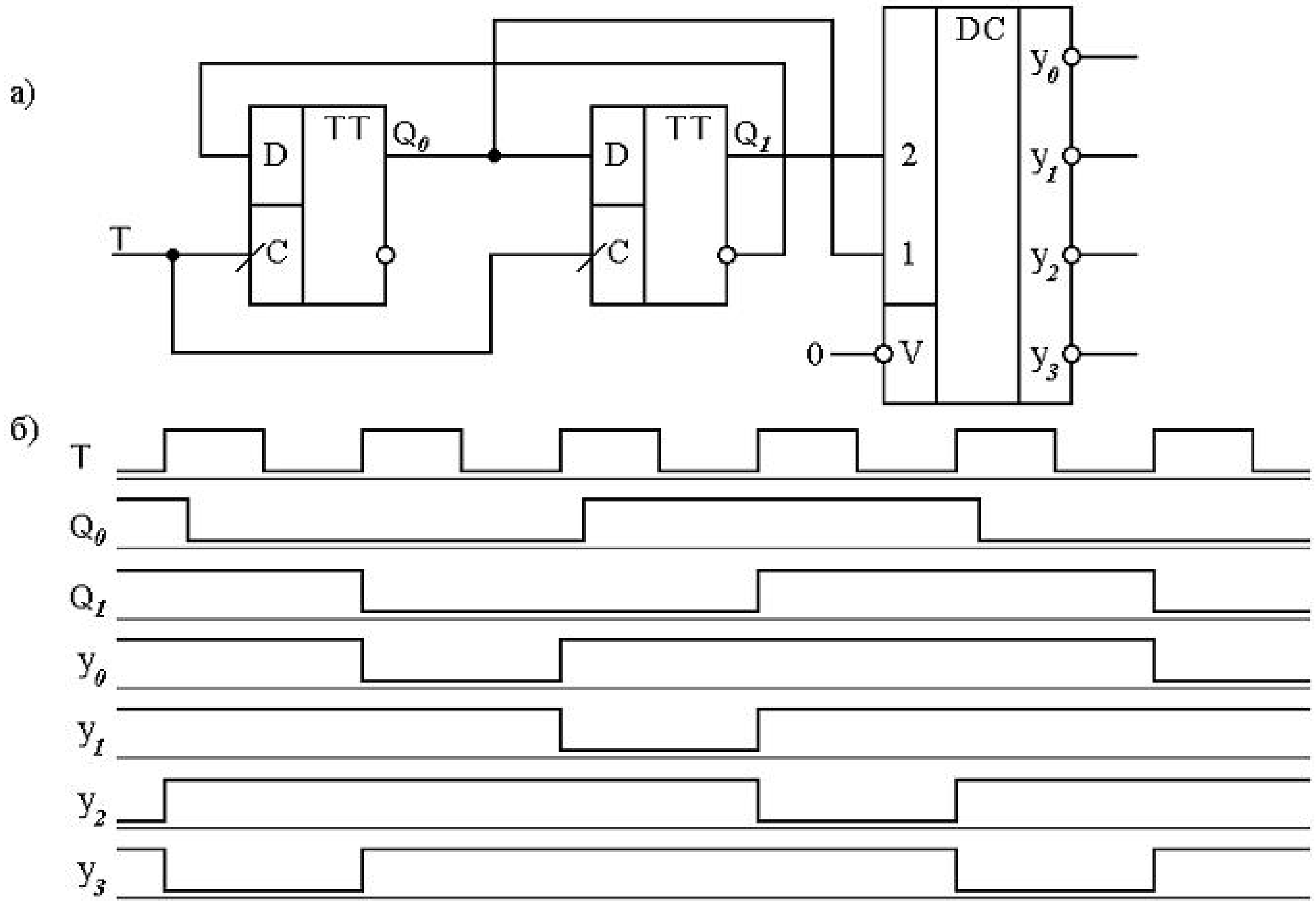


Методы устранения рисков сбоя.

Полезно помнить, что ***стробирование осуществляется комбинационной схемой и деформирует длительность информационного сигнала**, **а тактирование осуществляется последовательностной схемой и сдвигает во времени информационный сигнал***.

<u>**Пример**</u> ***функционального метода*** устранения рисков сбоя приведен на рис. Здесь используется принцип изменения кодирования последовательных состояний входов комбинационной схемы. В схеме рис. *а* счетчик изменяет свои состояния не в естественной двоичной последовательности 0 - 1- 2 - 3, а в последовательности 0 - 1 - 3 - 2, когда в каждом такте свое состояние изменяет только один разряд счетчика (здесь используется двухразрядный счетчик Джонсона - сдвиговый регистр с одной перекрестной связью).

Функциональный метод.



Методы устранения рисков сбоя.

Разрабатываются методы борьбы с гонками - построению ***самосинхронизирующихся схем***.

Рабочие узлы в этом случае строятся непротивогоночными, но они дополняются специальными схемами, которые обнаруживают факт окончания переходных процессов и вырабатывают разрешающий сигнал для следующих схем, играющий в каком-то смысле роль “асинхронного синхросигнала”. Это направление рассматривается как весьма перспективное для построения БИС и особенно сверхБИС, где применение обычной синхронизации встречает ряд трудностей.

Однако есть ряд сложностей построения такого рода схем, так и из-за удвоения аппаратурных затрат.

Конструктивно-технологический метод.

К ***конструктивно-технологическому методу*** можно отнести метод борьбы с влиянием входных гонок на уровне одного логического элемента, основанный исключительно на совершенствовании технологии изготовления логических элементов и их конструктивного исполнения, которые в совокупности должны обеспечить минимальные разбросы задержки распространения сигналов в элементе по логически одинаковым цепям. В частности, использование интегральной технологии производства ИС, позволяющей получать весьма малые относительные вариации различных параметров, дает возможность считать, что гонки по входу практически не влияют на устойчивость цифровых устройств, проектируемых на любой серии логических ИС.

Полезно придерживаться также следующих рекомендаций:

- соблюдайте требования технических условий (ТУ) на ИС;

- тщательно проектируйте аппаратуру, уделяя особое внимание рациональной разводке печатных плат;

- обеспечивайте режимы эксплуатации, не изменяющие задержки распространения сигналов (стабилизация источников питания, термостатирование, буферизация нагрузок и др.).

Реальные логические элементы

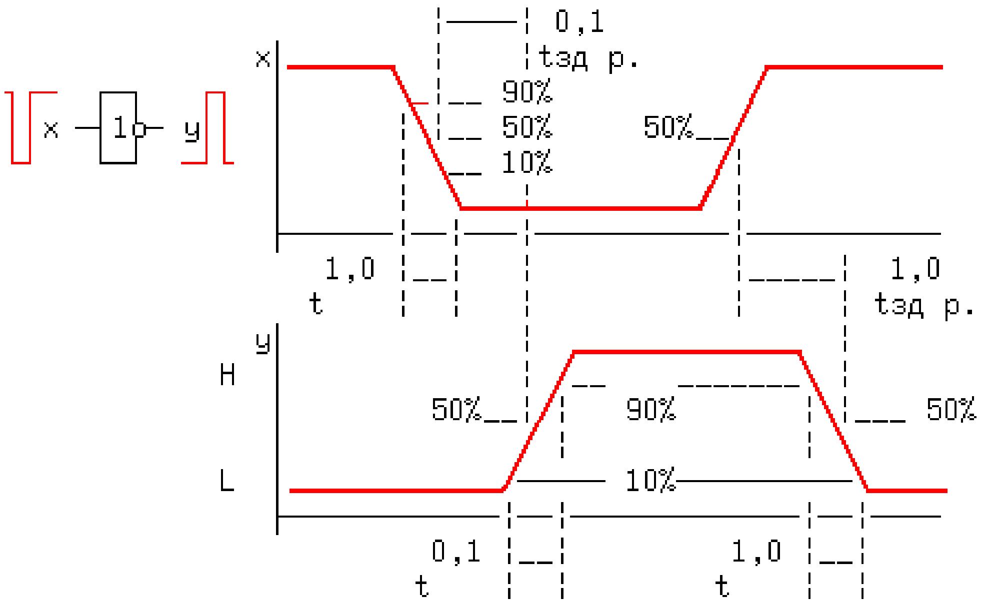
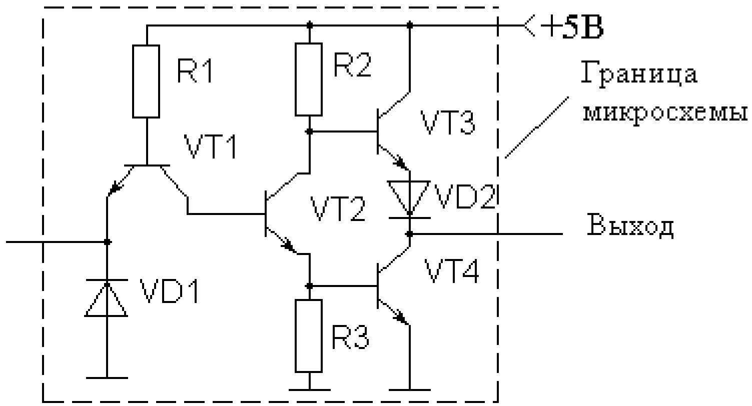


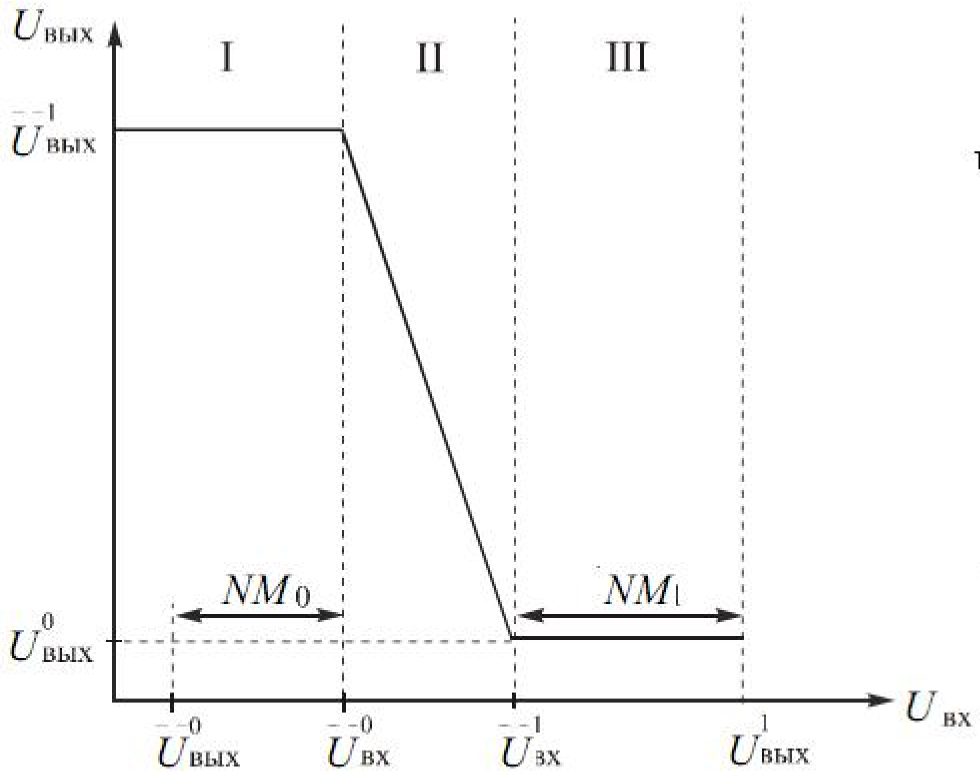
Рис.9.



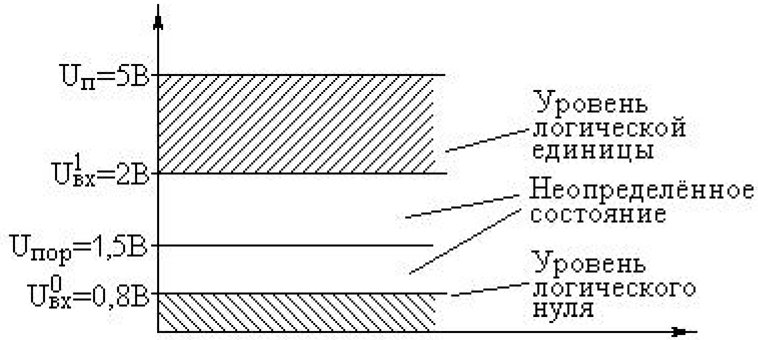
$$t_{\text{зд.р.ср.}} = \frac{t^{01}_{\text{зд.р.}} + t^{10}_{\text{зд.р.}}}{2}$$

Реальные логические элементы

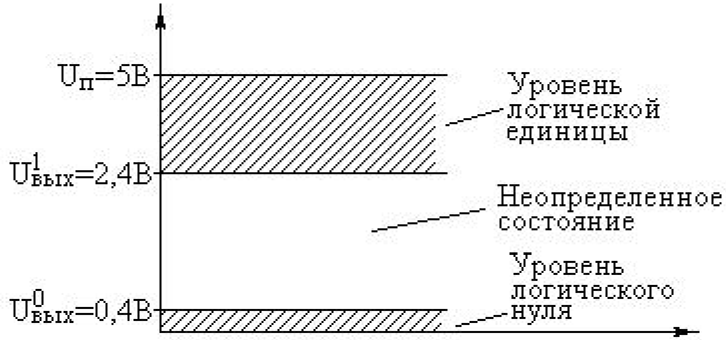
Передаточная функция ЛЭ



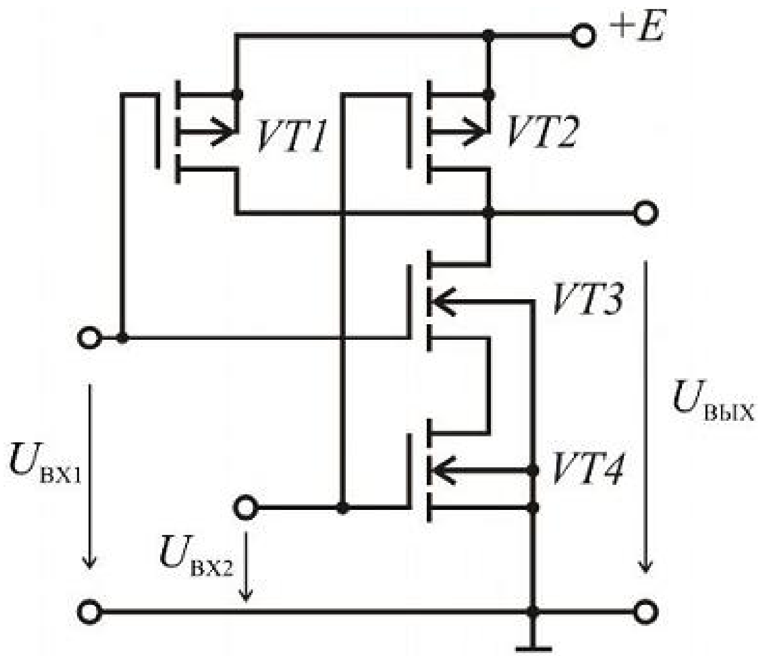
Уровни на входе ЛЭ

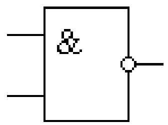


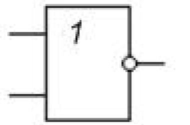
Уровни на выходе ЛЭ

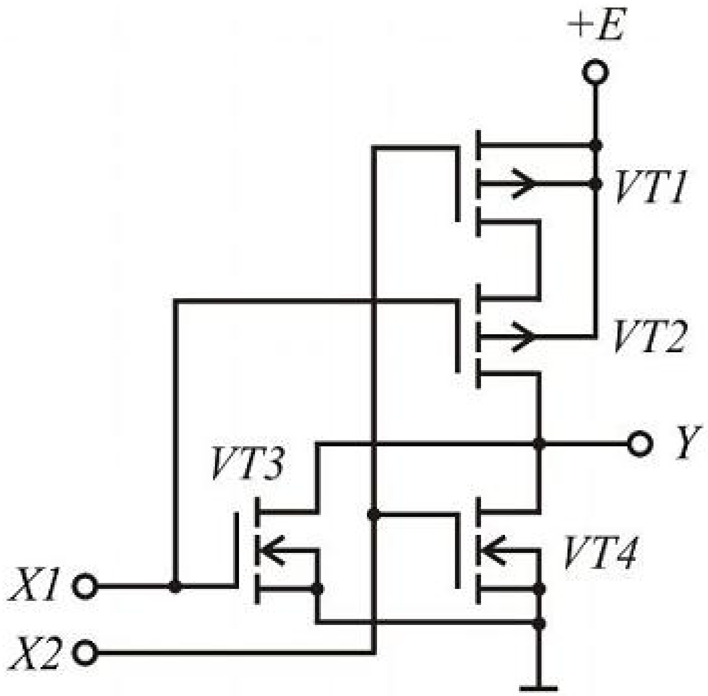


Реальные логические элементы

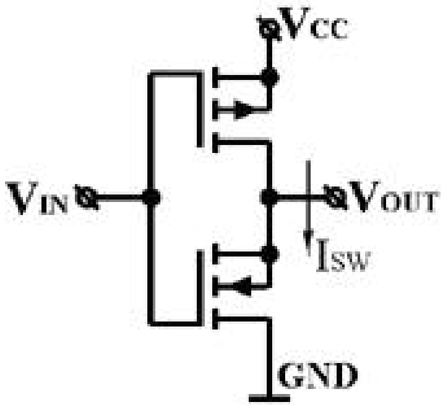


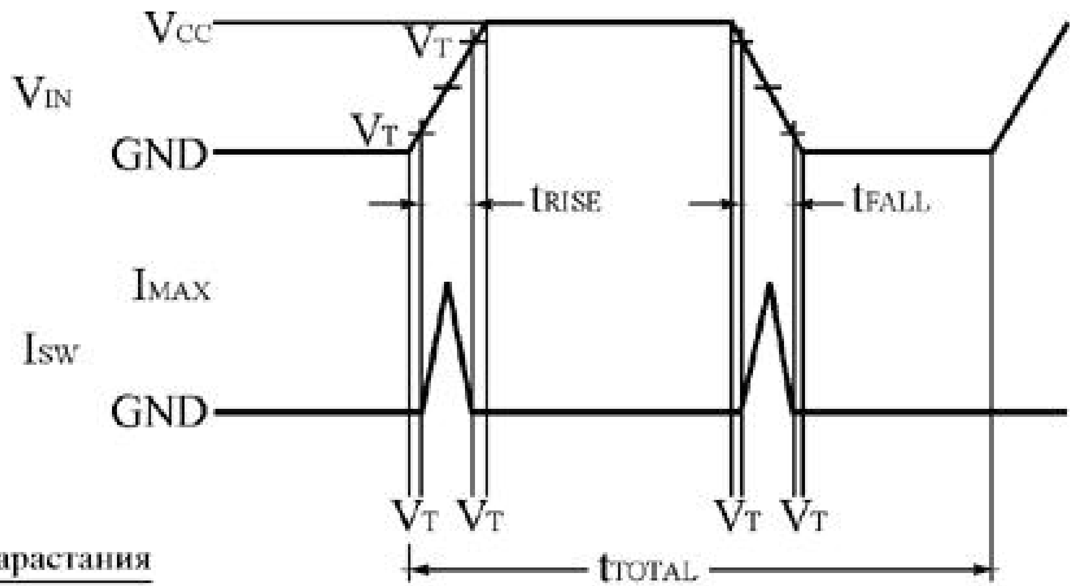






Реальные логические элементы





Мощность VI определяется:

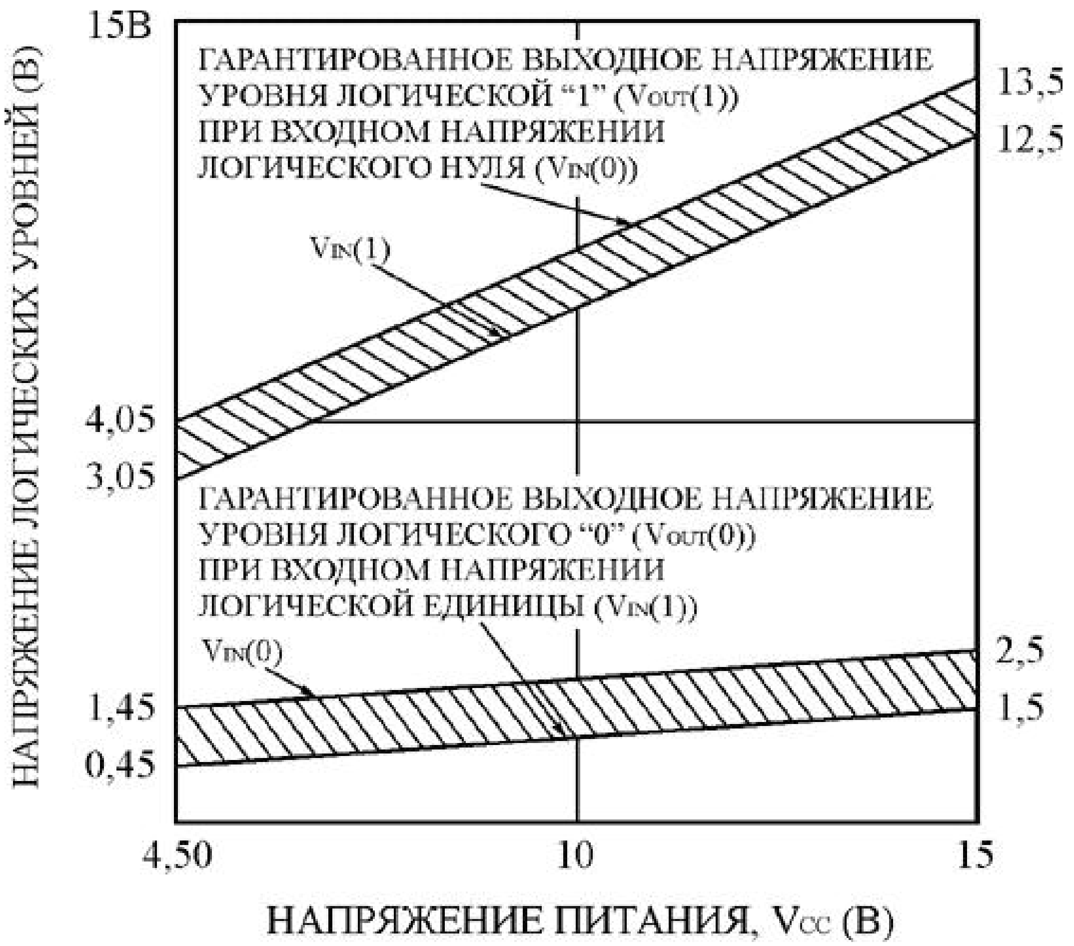
$$P_{VI} = V_{CC} \times \frac{1}{2} I_{MAX} \times \frac{\text{Время нарастания}}{\text{Период}}$$

$$\frac{\text{Время нарастания}}{\text{Период}} = \frac{V_{CC} - 2V_T}{V_{CC}} \times \frac{t_{RISE} + t_{FALL}}{t_{TOTAL}}, \text{ где } \frac{1}{t_{TOTAL}} = \text{ЧАСТОТА}$$

$$P_{VI} = \frac{1}{2}(V_{CC} - 2V_T)\, I_{CC\ MAX}\, (t_{RISE} + t_{FALL}) \times \text{ЧАСТОТУ}$$

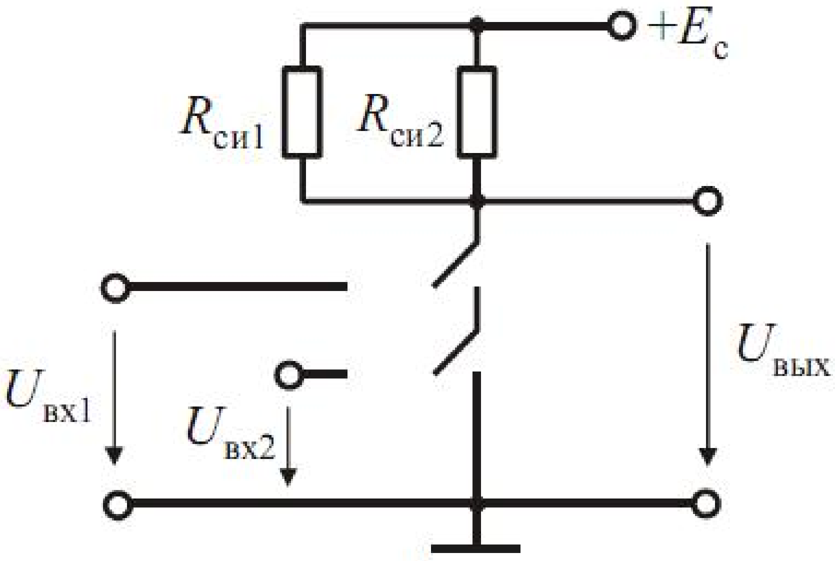
Мощность, рассеиваемая в моменты переключения

Реальные логические элементы



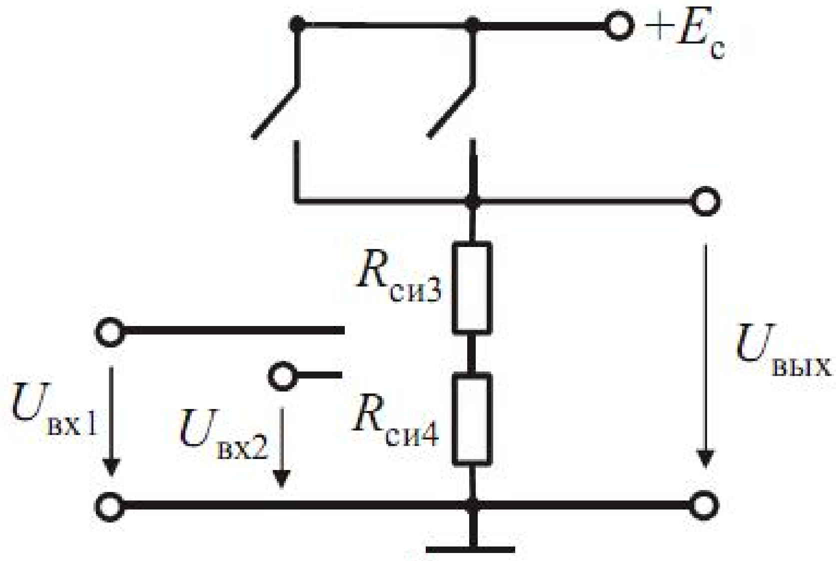
Логические элементы на ЭМ переключателях

_2И-НЕ_



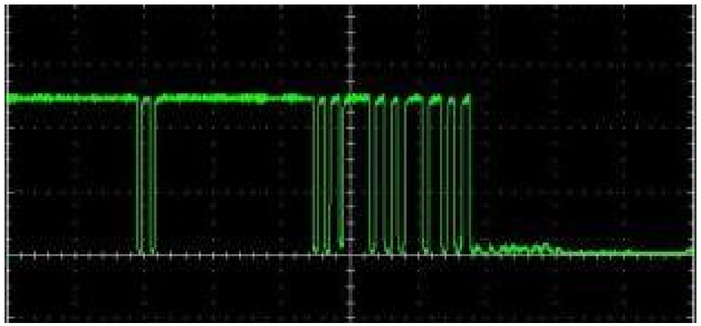
а

2ИЛИ-НЕ

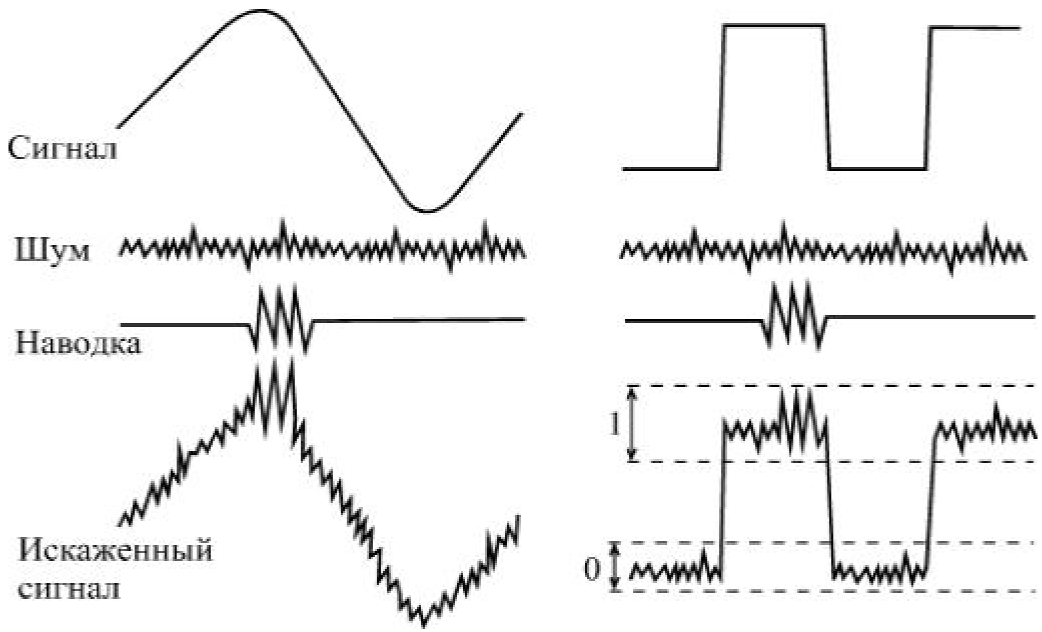


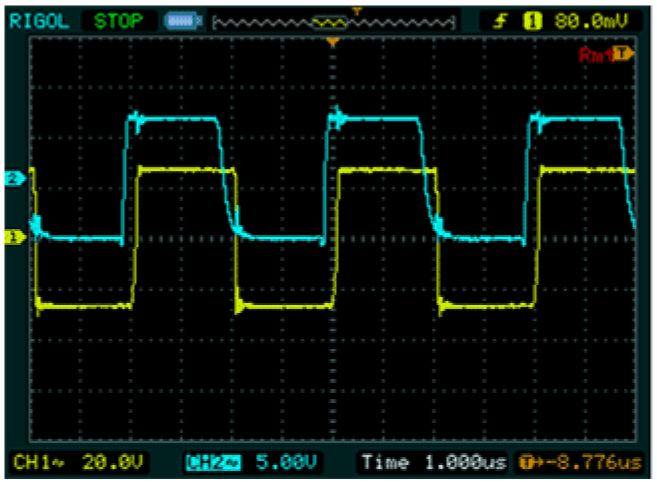
б

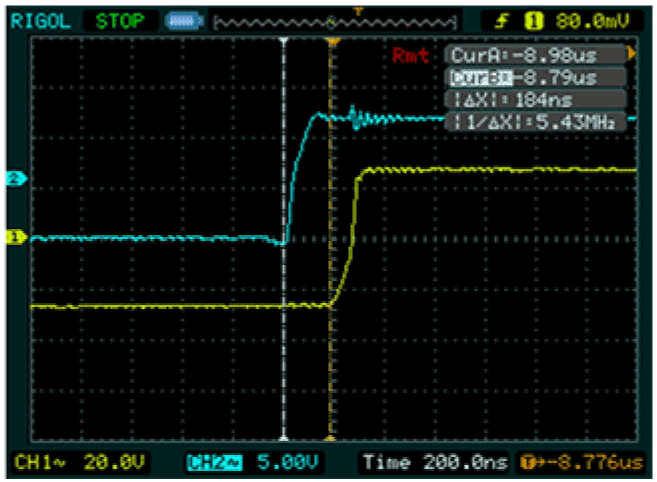
«Дребезг» контактов:
(D_)



Реальные логические элементы







Риски сбоя в комбинационных схемах

Определения:

Риск сбоя - *возможность* появления на выходе цифрового устройства сигнала, не предусмотренного алгоритмом его работы и могущего привести к *ложному срабатыванию.*

РИСК СБОЯ → ЭТО НАИХУДШИЙ СЛУЧАЙ!!

Функциональная устойчивость определяется *стабильностью реализации цифровым устройством заданного алгоритма работы* при наличии разброса задержек выполнения операций в логических элементах, задержек сигналов в линиях связи и эл ектромагнитных наводок паразитных сигналов.
Синоним "функциональной устойчивости" - ***алгоритмическая устойчивость.***

Риски сбоя в комбинационных схемах

Определения:

Состязания (гонки) сигналов - процесс распространения сигналов в различных цепях цифрового устройства при существовании разбросов временных задержек этих цепей.

Цепь - совокупность логических и других элементов и линий связи между ними.

Алгоритмический переход - изменение сигнала на выходе какой-либо схемы, предусмотренное алгоритмом ее работы.

Неалгоритмический переход - изменение выходного сигнала, не предусмотренное алгоритмом ее работы.

Опасные состязания - которые могут привести в цифровой схеме к неалгоритмическому переходу при заданных условиях ее работы.

Риски сбоя в комбинационных схемах

Определения:

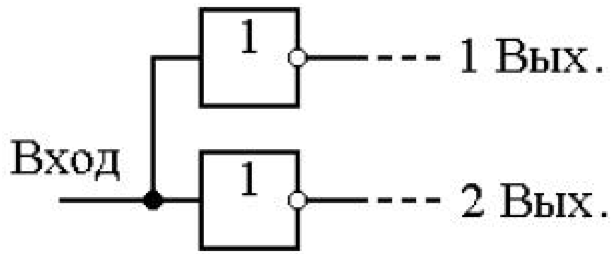
Неопасные состязания – которые не могут привести в схеме к неалгоритмическому переходу при заданных условиях ее работы.

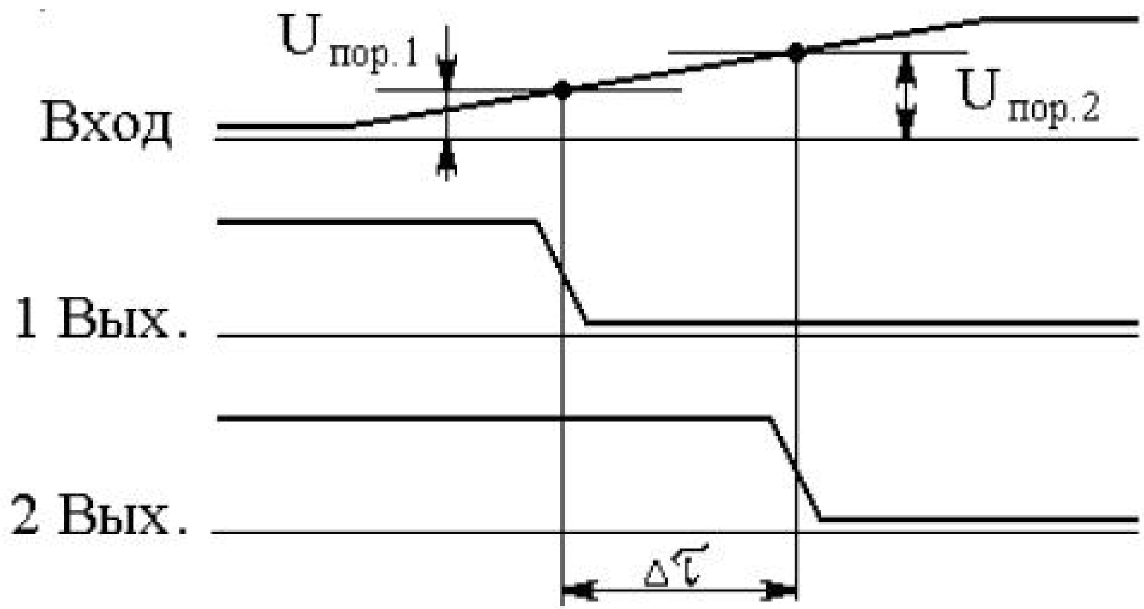
Схема, свободная от влияния опасных состязаний - цифровая структура, в которой неалгоритмический переход, возникший в части схемы из-за опасных состязаний, не изменяет алгоритма работы схемы в целом при заданных условиях ее работы.

Опасными состязаниями (гонками) по входу – называют неодновременное переключение логических элементов, при одновременном изменении сигнала на их входах, связанное с тем, что логические элементы изготовленные в различных технологических циклах имеют различные параметры. При различных пороговых входных напряжениях получаем неодновременное срабатывание логических элементов.

Риски сбоя в комбинационных схемах

Гонки по входу.





Риски сбоя в комбинационных схемах

Определения:

*Изменение сигнала на каждом выходе схемы реально происходит не мгновенно, а образует некоторый сложный **динамический процесс**.*

*Нахождение этих процессов называется **динамическим анализом** комбинационной схемы.*

***Динамический анализ** учитывает обстоятельства:*

1)Изменение входного набора схемы состоит из неодновременных изменений различных входных переменных, образующих этот набор. (последовательность входных наборов можно рассматривать как набор входных нулей и единиц, действующих независимо друг от друга на разных входах.

*2)Помимо логических элементов в схеме могут иметься специальные вспомогательные элементы – **задержки***

3)Каждый инерционный логический элемент в большинстве случаев можно представить в виде модели, содержащей последовательное соединение безынерционного ЛЭ с элементом задержки (иногда фильтром) на ‘τ’

Риски сбоя в комбинационных схемах

Определения:

*Изменение сигнала на каждом выходе схемы реально происходит не мгновенно, а образует некоторый сложный **динамический процесс**.*

*Нахождение этих процессов называется **динамическим анализом** комбинационной схемы.*

***Динамический анализ** учитывает обстоятельства:*

1)Изменение входного набора схемы состоит из неодновременных изменений различных входных переменных, образующих этот набор. (последовательность входных наборов можно рассматривать как набор входных нулей и единиц, действующих независимо друг от друга на разных входах.

*2)Помимо логических элементов в схеме могут иметься специальные вспомогательные элементы – **задержки***

3)Каждый инерционный логический элемент в большинстве случаев можно представить в виде модели, содержащей последовательное соединение безынерционного ЛЭ с элементом задержки (иногда фильтром) на ‘τ’

Риски сбоя в комбинационных схемах

Определения:

Переключательный процесс - последовательность уровней “1” и “0” (импульсов и пауз), которая на любом конечном наблюдаемом интервале времени содержит конечное число переходов 01 и 10.

Длиной переключательного процесса называется общее число изменений сигнала в нем. Например, для процесса x_4 на рис. 2 длина равна 3.

Переключательный процесс сложный - если его длина $\geq$2, в случае если длина $\leq$2 - ***простое переключение***.

Векторный переключательный процесс считается ***простым переключением***, если все его компоненты - *простые переключения, совершаемые* ***одновременно***. В противном случае векторный процесс считается ***сложным***.

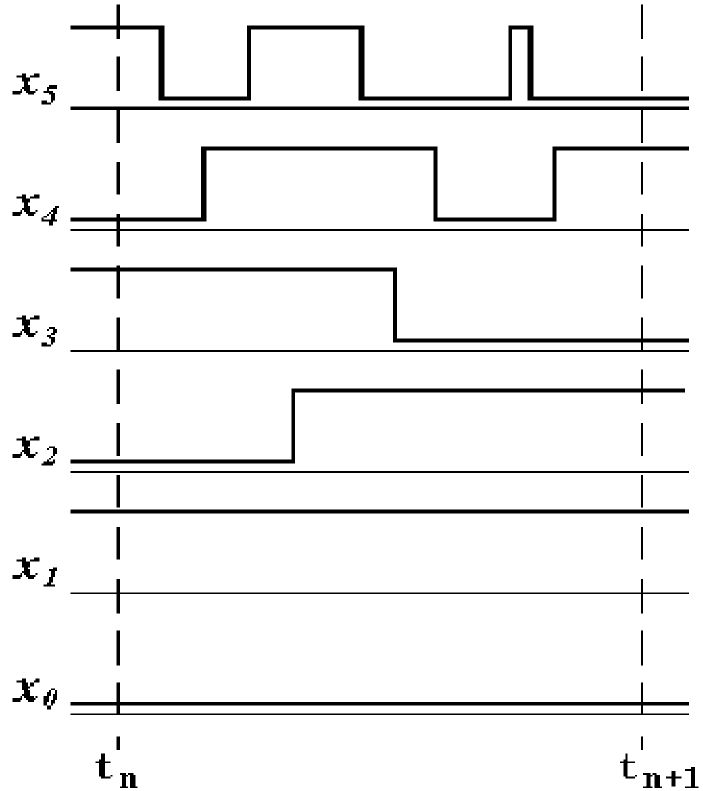
Риски сбоя в комбинационных схемах

Векторный переключательный процесс считается ***простым переключением***, если все его компоненты - *простые переключения, совершаемые* ***одновременно***. В противном случае векторный процесс считается ***сложным***.

$$\mathbf{X}_1 = x_5x_4x_3x_2x_1x_0 = 101010$$
$$\mathbf{X}_2 = 010110$$

Для векторного процесса существует понятие - вектор длин. Компоненты этого вектора - длины процессов, являющихся компонентами векторного процесса. Например, векторный процесс, на рис., имеет вектор длин:

(5, 3, 1, 1, 0, 0).



Риски сбоя в комбинационных схемах

Событие *- любое изменение логического сигнала, в том числе сложный переключательный процесс.*

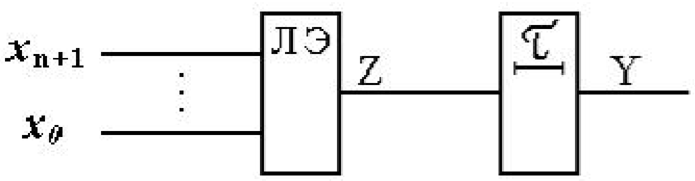
Различают два вида задержек:

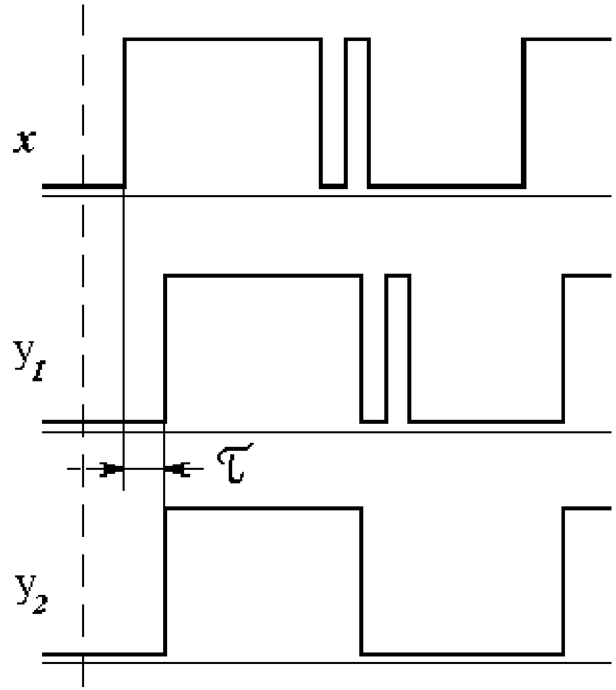
*1)**чистая задержка**, которая при подаче на вход сигнала $x(t)$ обусловливает на выходе сигнал $y(t–\tau)$, где τ – величина задержки*

*2)**инерционная задержка** или **фильтр** - осуществляет ту же операцию, что и чистая задержка, но сверх того не пропускает на выход изменений входного сигнала, отстоящих одно от другого по времени менее чем на ‘τ’, благодаря чему процесс на выходе может изменить форму.*

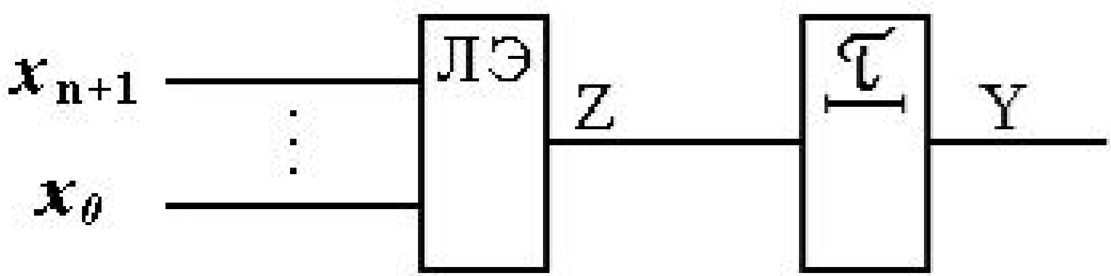
Риски сбоя в комбинационных схемах

Задержки, связанные с логическими элементами и линиями связи, обычно называют ***паразитными задержками****.* *Справедливо утверждение, что* ***паразитные задержки*** *имеют компоненты как чистой, так и инерционной задержки.* *Различия двух видов задержек:*





Риски сбоя в комбинационных схемах



*Под ‘**τ**’подразумевается паразитная задержка. Величину ‘**τ**’, а также моменты изменений входных переменных схемы, называют **временными параметрами**. Очевидно, что в общем случае значение ‘**τ**’моделирующей задержки зависит от того, какое изменение сигнала 01 или 10 имеет место на выходе элемента, то есть* $\boldsymbol{\tau=\tau_{01}}$ *и* $\boldsymbol{\tau=\tau_{10}}$. *В простейшем случае* $\boldsymbol{\tau_{01}=\tau_{10}=\tau}$.

Деформирование выходных сигналов

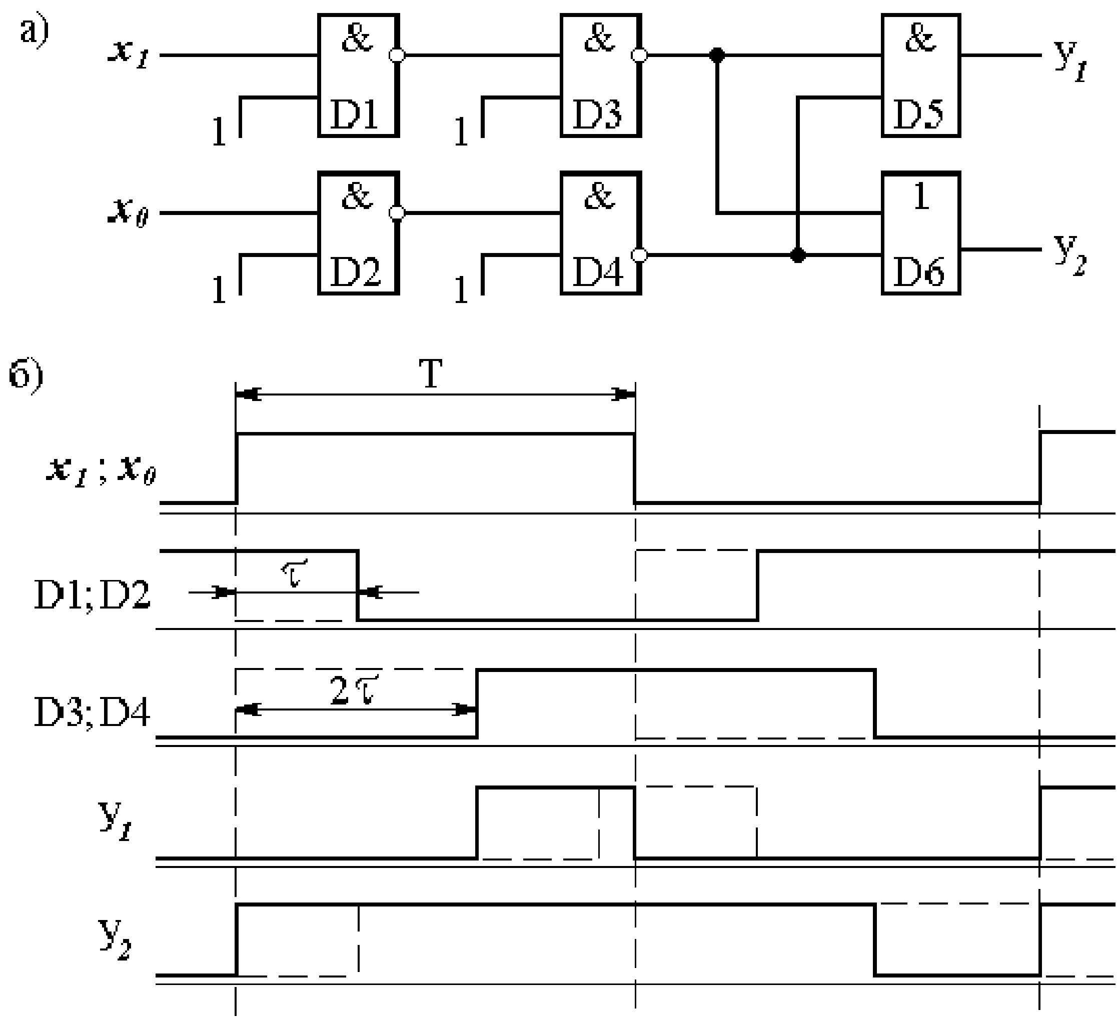
В различных частях комбинационной схемы в зависимости от числа последовательно включенных элементов переходный процесс после смены входного набора будет заканчиваться в разное время. Это приведет либо к ***деформированию длительности выходных сигналов*** *либо к появлению* ***рисков сбоя****.*

Если сигналы в схеме распространяются по цепочкам, задержки в которых различны, то это приводит к смещению сигналов относительно друг друга во времени. В свою очередь, это может вызвать уменьшение длительности сигнала "1" на выходе элемента ***И*** *и увеличение - на выходе* ***ИЛИ****.*

Деформирование выходного сигнала может привести к исчезновению алгоритмически верного сигнала*!!!*

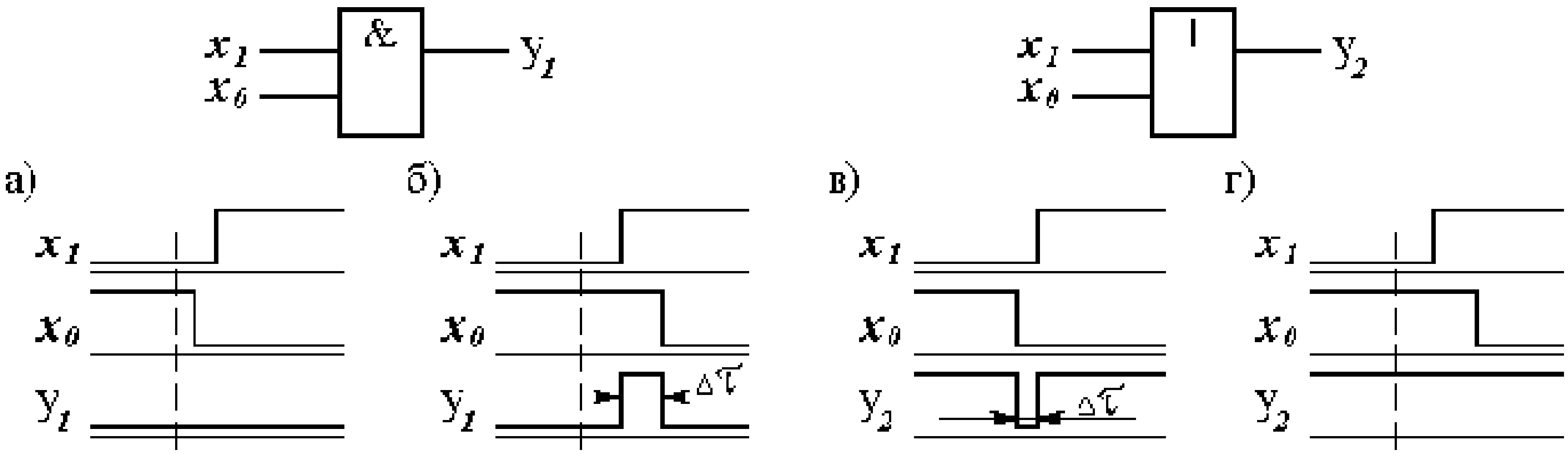
.

Деформирование выходных сигналов



Статические риски сбоя

*На рис. показана работа элементов **И** и **ИЛИ** при подаче на их входы двух последовательных во времени наборов* $\mathbf{X}_1 = x_1x_0 = 01$ *и* $\mathbf{X}_2 = x_1x_0 = 10$



*Ложные сигналы на выходе и являются **рисками сбоя**, причем видно, что они могут быть, а могут и отсутствовать.*

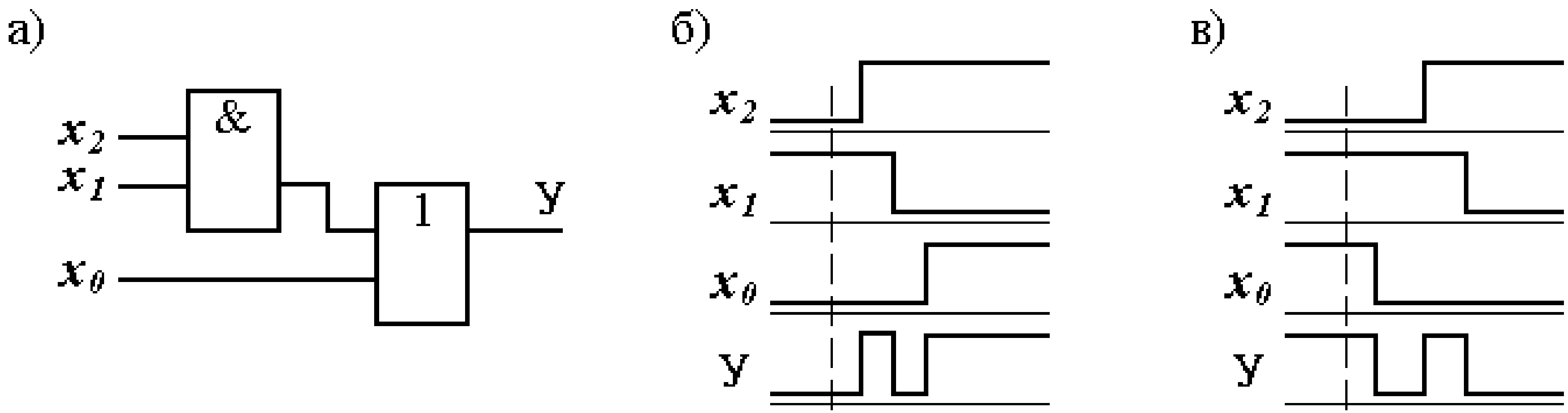
Статические риски сбоя

Риск сбоя называется ***статическим****, если* $\mathbf{y}(X_1) = y(\mathbf{X}_2)$*, где* y *- булева функция. Риск сбоя называется* ***статическим в нуле*** S_0*, если* $\mathbf{y}(X_1) = y(\mathbf{X}_2) = 0$*. Риск сбоя называется* ***статическим в единице*** S_1*, если* $\mathbf{y}(X_1) = y(\mathbf{X}_2) = 1$*.*

На рис. б имеет место статический риск сбоя в нуле S_0*, а на рис. в - статический риск сбоя в единице* S_1*.*

Динамические риски сбоя

На рис. а приведена схема, реализующая функцию $\boldsymbol{y} = x_2x_1 + x_0$*. Пусть входной набор* $\boldsymbol{X}_1 = x_2x_1x_0 = 010$ *изменяется на входной набор* $\boldsymbol{X}_2 = x_2x_1x_0 = 101$



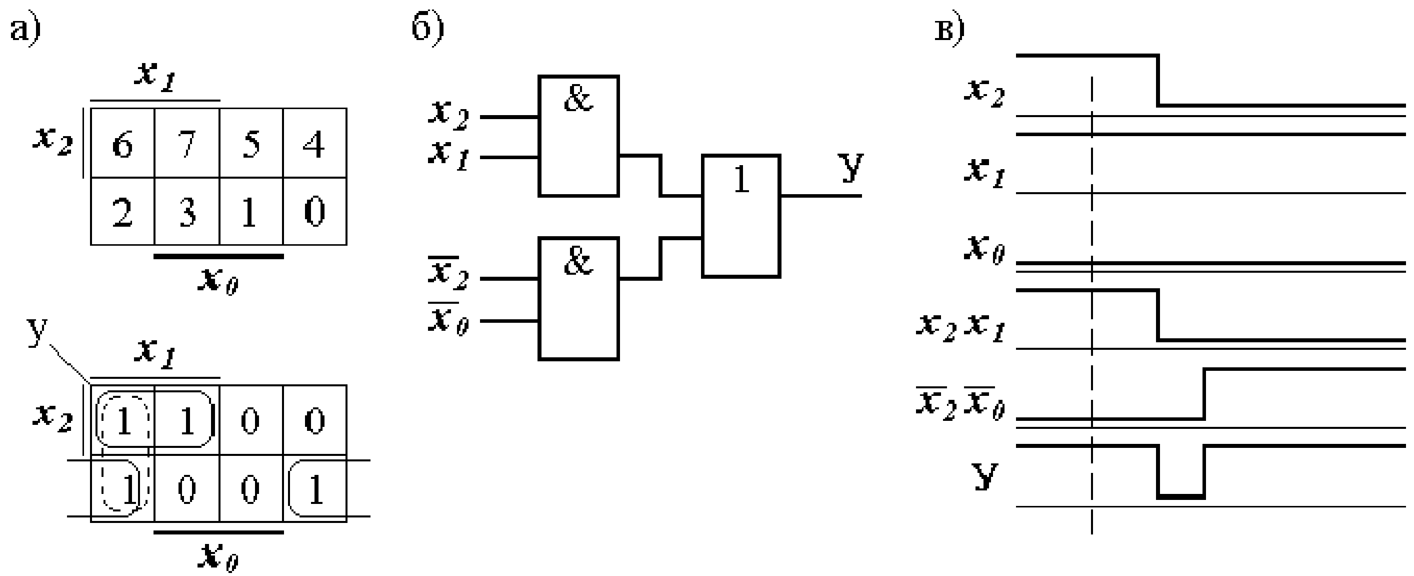
На рис. б) имеет место динамический риск сбоя D_+*, а на рис. в) -* D_-*. Наличие динамических рисков сбоя в цифровой схеме также может привести к нарушению закона ее функционирования.*

Динамические риски сбоя

Риск сбоя называется ***динамическим****, если* $\mathbf{y}(X_1) \neq y(\mathbf{X}_2)$*, где* y *- булева функция. Риск сбоя называется* ***динамическим*** D_+ *при переходе на выходе 01, если* $\mathbf{y}(X_1) = 0$*, а* $y(\mathbf{X}_2) = 1$*. Риск сбоя называется* ***динамическим*** D_-*, если* $\mathbf{y}(X_1) = 1$*, а* $y(\mathbf{X}_2) = 0$*.*

Логический риск сбоя

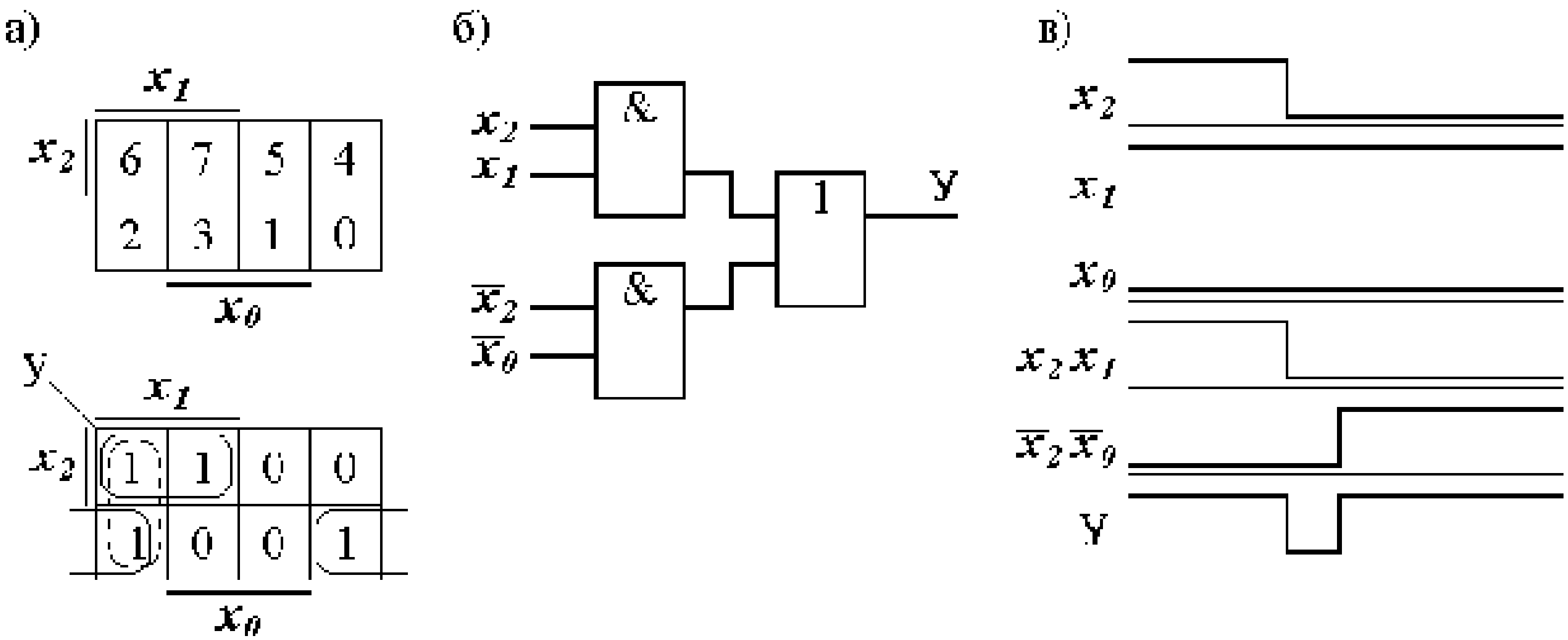
Рассмотрим переход от $\mathbf{X}_1 = x_2x_1x_0 = 110$ *к* $\mathbf{X}_2 = x_2x_1x_0 = 010$ *для функции* ***y****, представленной картой Карно (рис. 8, а). Для нее можно записать* $y = x_2 x_1 + \overline{x}_2 \overline{x}_0$.



Логический риск сбоя

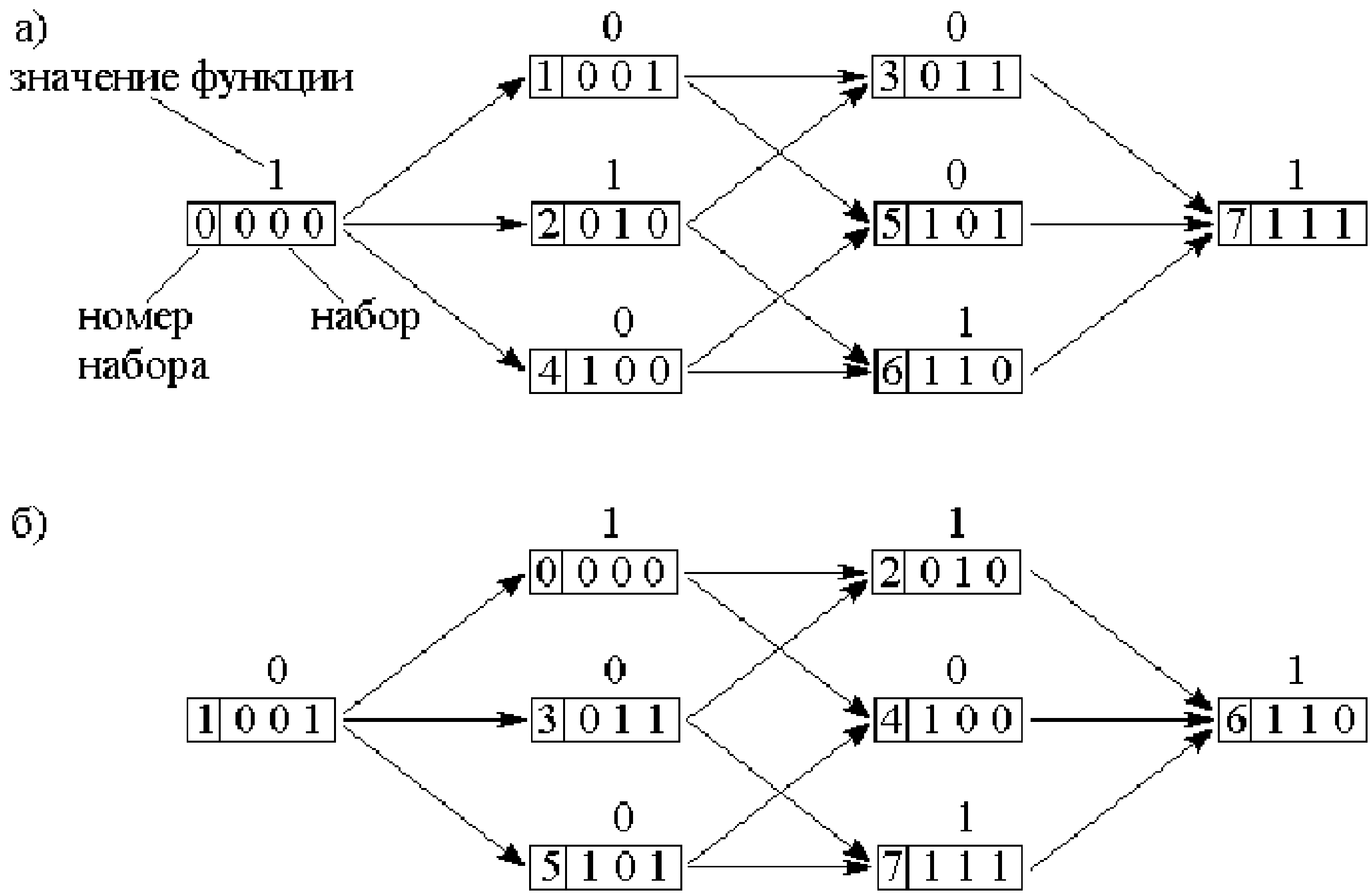
Устраним риск сбоя, для этого введем дополнительный контур

$$y = x_2 x_1 + \bar{x}_2 \bar{x}_0 \rightarrow y = x_2 x_1 + \bar{x}_2 \bar{x}_0 + x_1 \bar{x}_0$$



Статический риск сбоя, проявляющийся при ***соседней смене наборов****, называется* ***логическим****, так как может быть устранен* ***изменением логической структуры****, реализующей булеву функцию*

Функциональный риск сбоя

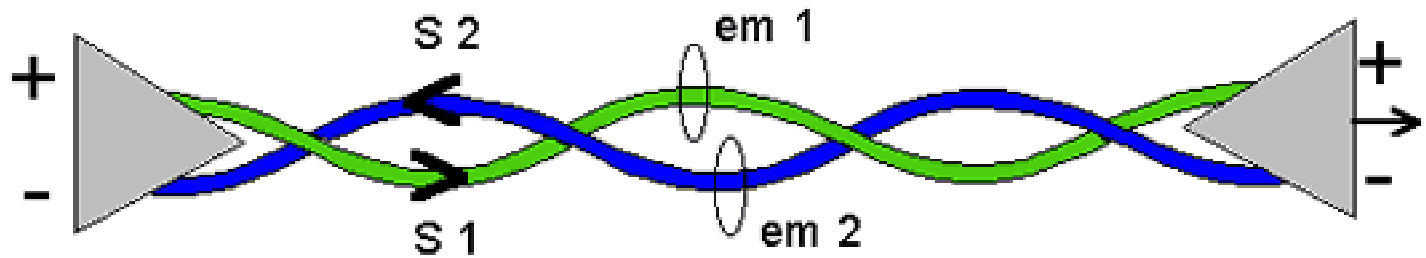


Функциональный риск сбоя

Есть единственный путь смены наборов: 0 2 6 7, при котором не будет статического риска сбоя, так как $\mathbf{y}(X_1 = 0) = y(\mathbf{X}_2 = 7) = 1$. *Во всех остальных случаях будет статический риск сбоя в единице* S_1, *причем никакими аппаратными средствами устранить его нельзя, так как значения выхода на промежуточных наборах определяются* ***характером самой функции***. *Аналогично для рис б) - имеет место динамический риск сбоя* D_+, *который также определяется характером самой функции.*

Риски сбоя, проявляющиеся при многоместной смене наборов и определяемые характером самой функции, называются ***функциональными***. *Такие риски сбоя не могут быть устранены изменением логической структуры, реализующей булеву функцию.*

Спасибо за внимание!



S1 = S2 \ em1 = em2

Минимизация функций алгебры логики

На практике решается более простая задача представления ФАЛ в дизъюнктивной или конъюнктивной форме, содержащей наименьшее возможное число букв (например, для СДНФ -- как можно меньше слагаемых, являющихся элементарными произведениями, которые содержат как можно меньше сомножителей). ***канонической задачей минимизации ФАЛ***

Задачу минимизации ФАЛ можно разложить на этапы:

1) *Переход от **СДНФ** к **сокращенной ДНФ***

2) *Переход от **сокращенной ДНФ** к **тупиковой ДНФ (ТДНФ)**, сокращением лишних импликант.*

3) *Переход от **ТДНФ** к **минимальной форме**.*

Методы минимизации ФАЛ

1) *Расчетный метод* – метод непосредственных преобразований;
2) *Метод Квайна*;
3) ***Расчетно-табличный метод (метод Квайна – Мак'Класски)***;
4) *Метод Петрика;*
5) ***Табличный метод (метод карт Карно)***;
6) ***Метод неопределенных коэффициентов***;
7) *Метод гиперкубов;*
8) *Метод факторизации;*
9) *Метод функциональной декомпозиции;*

...

Табличный метод

В данном методе применяются или ***диаграммы Вейча*** или ***карты Карно***, которые отличаются друг от друга расположением столбцов и строк.

В картах **Карно** порядок следования : 00 01 11 10.

В диаграммах **Вейча** порядок следования : 00 01 10 11.

Позиции наборов располагаются в соответствии с циклическим кодом Грея. Наборы **соседние**, если различаются только в одном разряде.

Эталонная и рабочая карты Карно для функции n=3:

	x_1	x_1		
x_2	6	7	5	4
	2	3	1	0
		x_0	x_0	

y	x_1	x_1		
x_2	1	0	1	1
	0	1	1	0
		x_0	x_0	

Правила минимизации для карт Карно

1. **В карте Карно группы единиц (ДНФ) необходимо покрыть контурами. Внутри контура должны находится только единицы. (соответствует операции *склеивания* - нахождения импликант данной функции).**

2. **Количество единиц контура должно быть степенью двойки** (1, 2, 4, 8, ...).

3. **Каждый контур должен включать максимально возможное количество клеток. В этом случае он будет соответствовать простой импликанте.**

4. **Все единицы в карте (даже одиночные) должны быть охвачены контурами. Любая единица может входить в контуры произвольное количество раз.**

5. **Множество контуров, покрывающих все единицы функции, образуют тупиковую ДНФ.**

6. **В элементарной конъюнкции, которая соответствует одному контуру, остаются только те переменные, значение которых не изменяется внутри контура.**

7. **На основании закона тавтологии любая единичная клетка может быть включена в любое количество контуров, для получения минимальной ДНФ нужно стремиться к отсутствию лишних покрытий.**

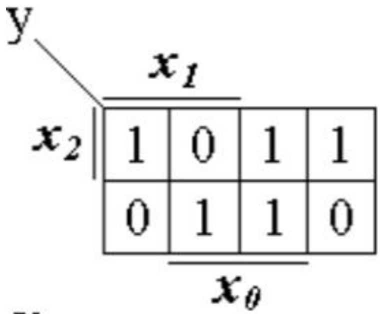
Табличный метод

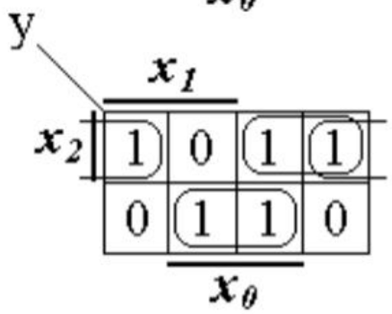
Пример. ФАЛ, заданную таблицей истинности (табл. 1), можно представить следующими выражениями

№ наб.	x_2	x_1	x_0	y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

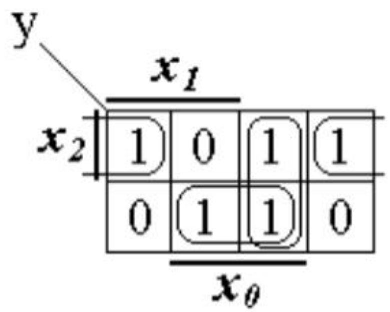
$$y = x_2 x_1 \overline{x}_0 + x_2 \overline{x}_1 x_0 + x_2 \overline{x}_1 \overline{x}_0 + \overline{x}_2 x_1 x_0 + \overline{x}_2 \overline{x}_1 x_0 \qquad (1)$$

$$y = x_2 \overline{x}_1 + x_2 \overline{x}_0 + \overline{x}_2 x_0 \qquad (2)$$



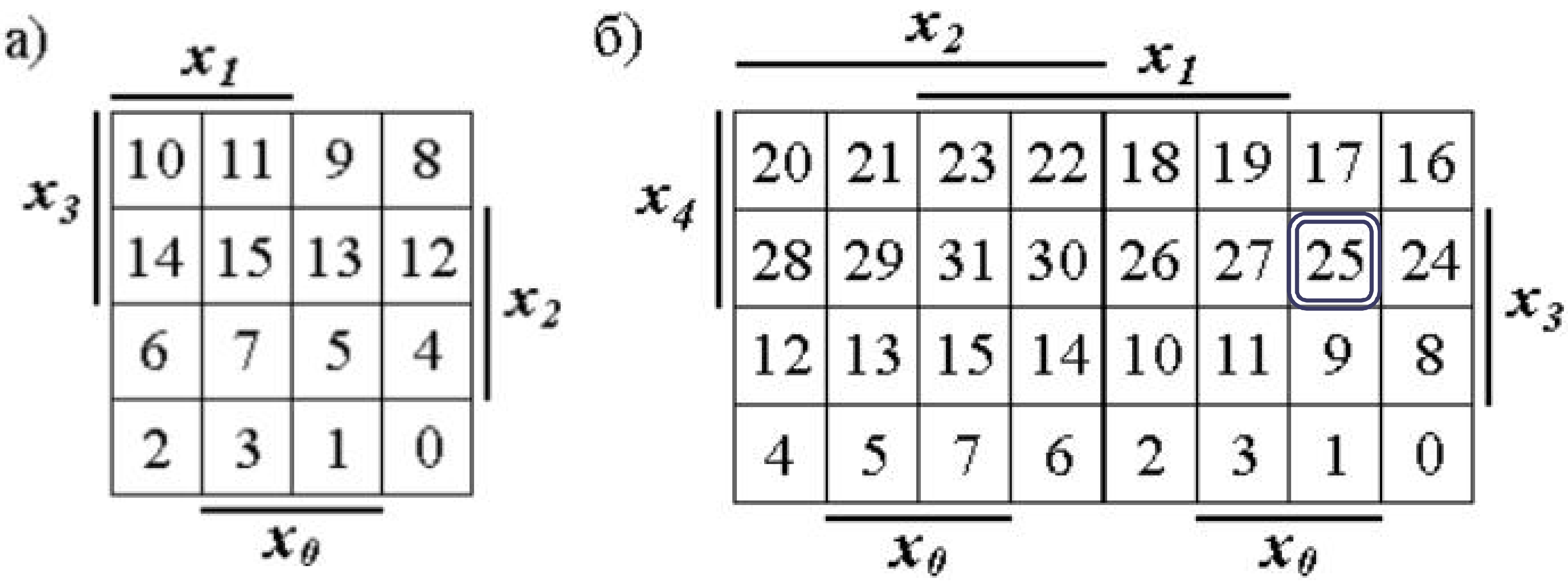


$$y = x_2 \overline{x}_1 + x_2 \overline{x}_0 + \overline{x}_2 x_0$$



$$y = x_2 \overline{x}_0 + \overline{x}_2 x_0 + \overline{x}_1 x_0$$

Эталонные карты Карно для n= 4, 5



Пример. n=5 Для клетки с набором 25 на рис. 4,б соседними являются клетки с номерами наборов 9, 27, 17, 24 и 29. Для клетки с набором 2 на рис. 4,б соседними являются клетки 3, 10, 0, 18 и 6. Для клетки с набором 43 на рис. 4,в соседними являются клетки с наборами 59, 42, 35, 41 и 47, 11. Для клетки с набором 22 на рис. 4, в соседними являются клетки с наборами 23, 30, 20, 6 и 54, 18.

Эталонная карта Карно для n= 6

в)

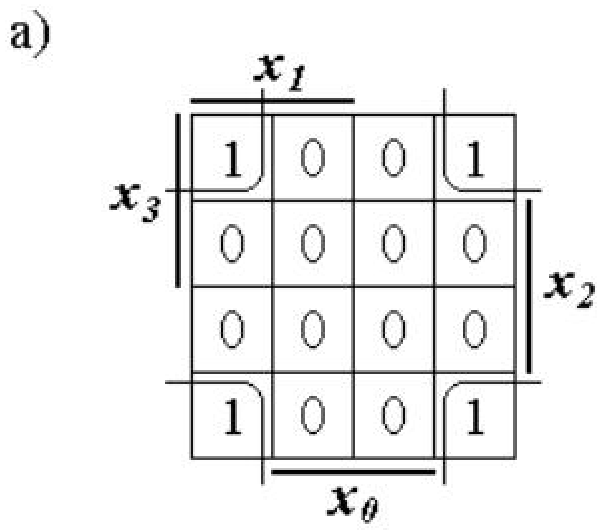
x_2 x_1

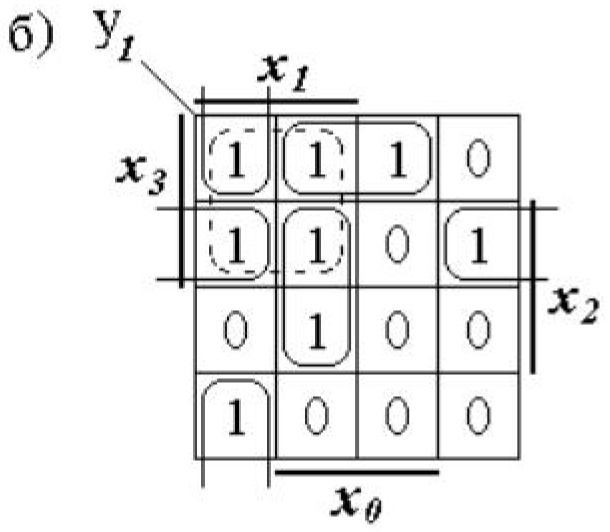
36	37	39	38	34	35	33	32
44	45	47	46	42	43	41	40
60	61	63	62	58	59	57	56
52	53	55	54	50	51	49	48
20	21	23	22	18	19	17	16
28	29	31	30	26	27	25	24
12	13	15	14	10	11	9	8
4	5	7	6	2	3	1	0

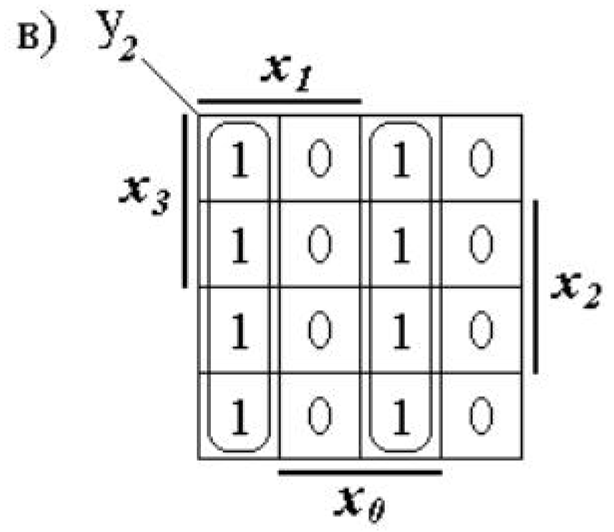
x_5 x_4 x_3 x_3

x_0 x_0

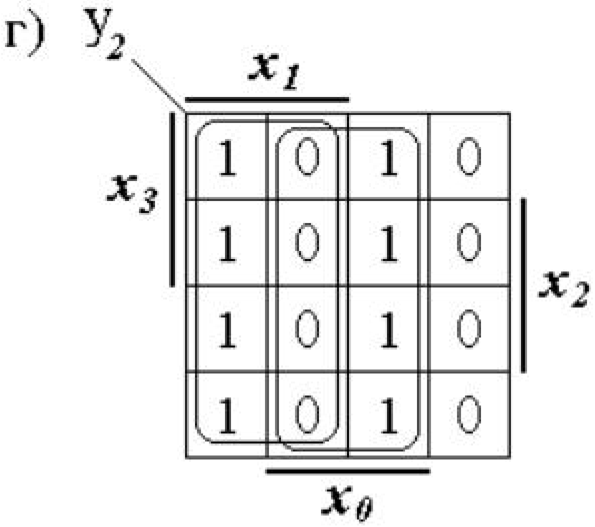
Минимизация на картах Карно для n= 4

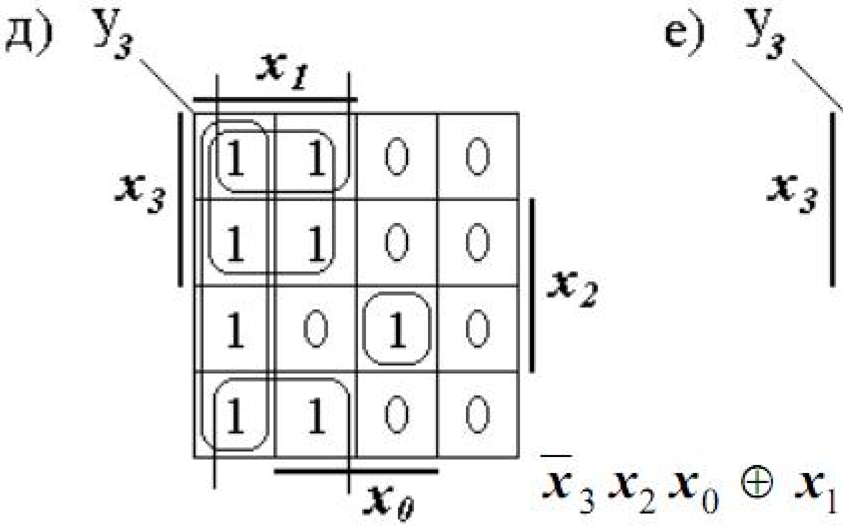


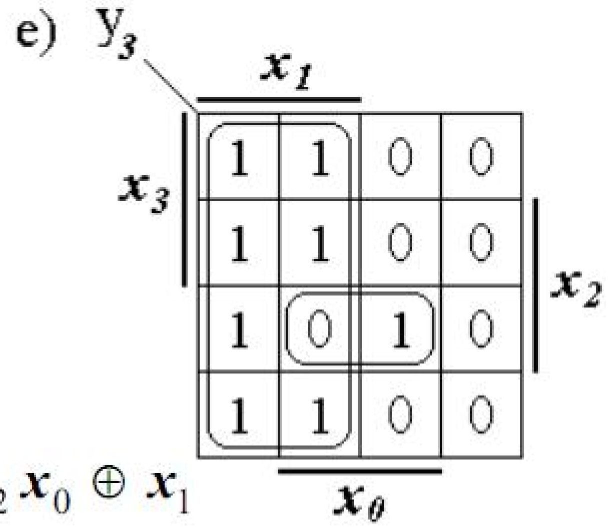




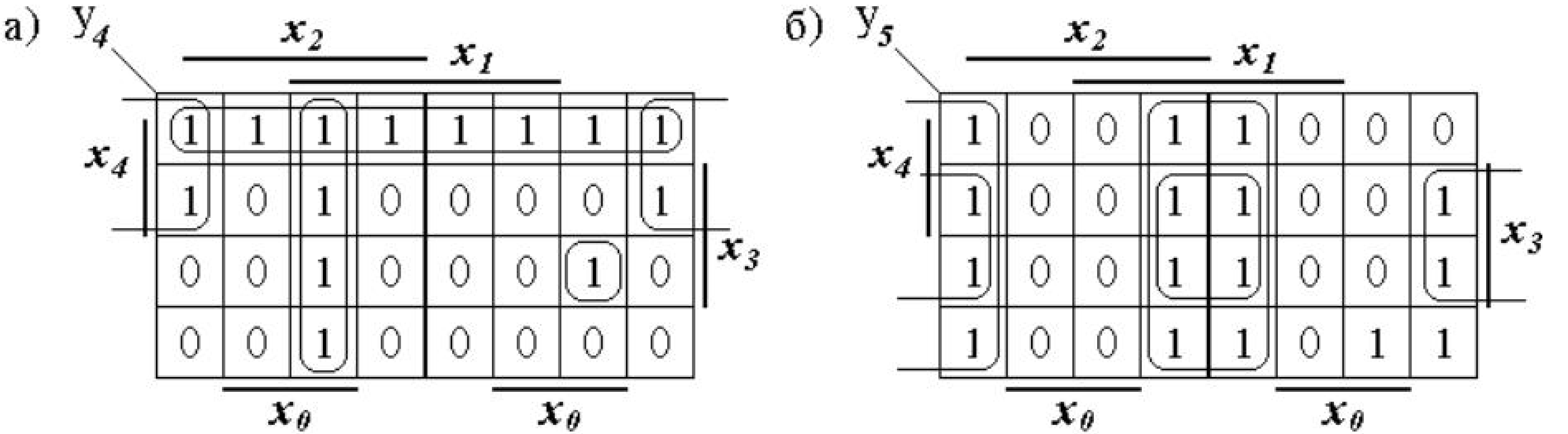
$y_0 = \bar{x}_2 \bar{x}_0$ $\quad y_1 = x_3 x_2 \bar{x}_0 + x_3 \bar{x}_2 x_0 + x_2 x_1 x_0 + \bar{x}_2 x_1 \bar{x}_0$ $\quad y_2 = x_1 \bar{x}_0 + \bar{x}_1 x_0 = x_1 \oplus x_0$







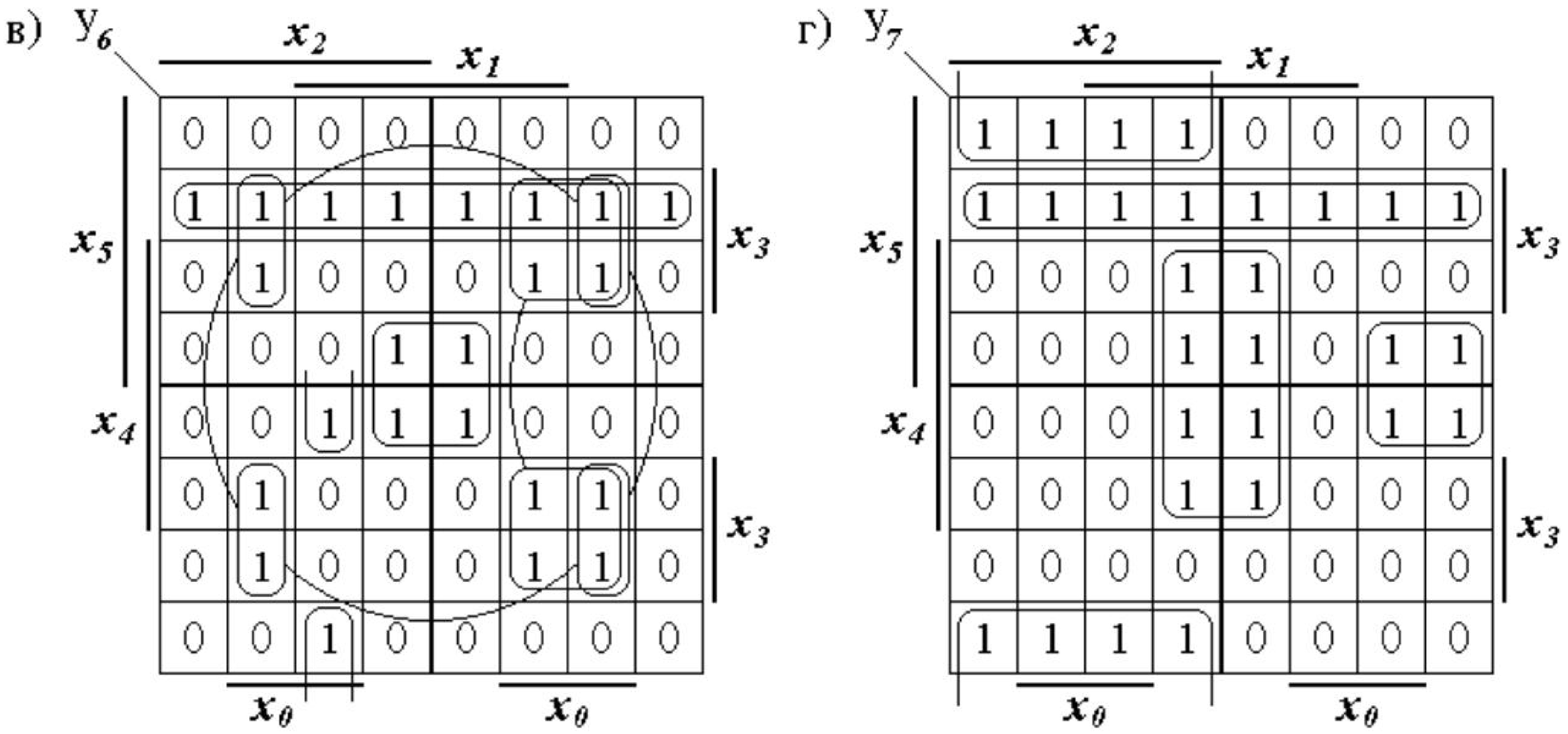
Минимизация на картах Карно для n= 5



$$y_4 = x_4\bar{x}_3 + x_4\bar{x}_1\bar{x}_0 + x_2 x_1 x_0 + \bar{x}_4 x_3 \bar{x}_2 \bar{x}_1 x_0$$

$$y_5 = x_3\bar{x}_0 + x_2\bar{x}_0 + x_1\bar{x}_0 + \bar{x}_4\bar{x}_3\bar{x}_2\bar{x}_1$$

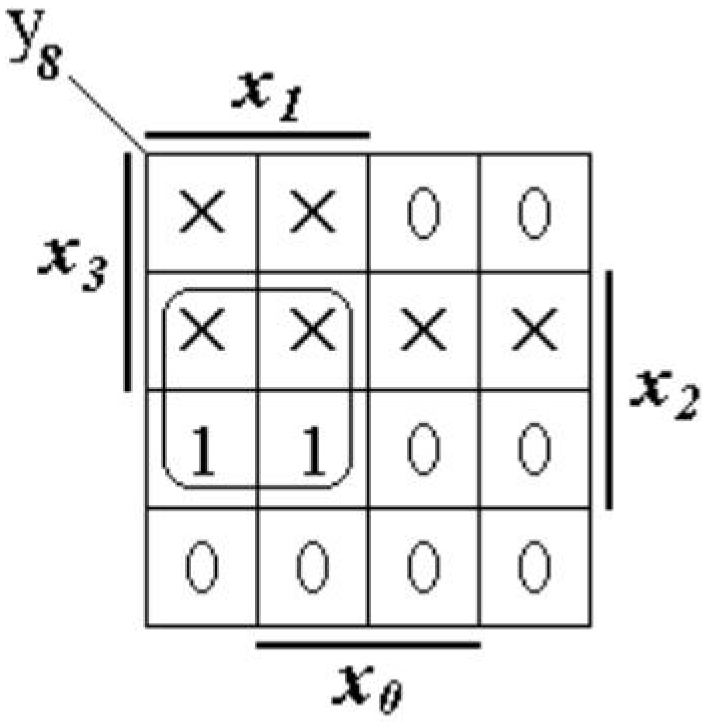
Минимизация на картах Карно для n= 6



$$y_6 = x_3\bar{x}_2 x_0 + x_3\bar{x}_1 x_0 + x_5\bar{x}_4 x_3 + x_4\bar{x}_3 x_1\bar{x}_0 + \bar{x}_5\bar{x}_3 x_2 x_1 x_0$$

$$y_7 = x_5\bar{x}_4 x_3 + x_4 x_1\bar{x}_0 + \bar{x}_4\bar{x}_3 x_2 + x_4\bar{x}_3\bar{x}_2\bar{x}_1$$

Минимизация неполностью определённой ФАЛ



$$y_8 = x_2\, x_1$$

Достоинства и недостатки табличного метода минимизации ФАЛ

Достоинства:

1. Основным достоинством применения карт Карно является компактность, простота и наглядность представления полностью и неполностью определенных функций.
2. Их применение оправдано для $n = 2 \div 6$, а при определенных навыках даже для $n = 7$ и 8, что соответствует большинству реально встречающихся инженерных задач.
3. Карты Карно можно использовать для минимизации ФАЛ, заданных как в СДНФ, так и в СКНФ.
4. Удобно минимизировать системы булевых функций, так как на картах Карно легко выделять общие части реализуемой системы ФАЛ.
5. Легко находятся минимальные комбинации контуров по их виду на карте Карно.
6. Для построения карты Карно не обязательно задавать её в СДНФ или СКНФ (можно подставить значения наборов в любой вид ФАЛ и заносить значения ФАЛ на этом наборе в соответствующую клетку карты Карно).
7. Карты Карно сразу позволяют реализовать первые два этапа минимизации (склеивание и выявление лишних импликант).

Недостатки:

1. Затруднительно использовать карты Карно при $n > 6$.
2. Метод не является алгоритмически систематическим, многое зависит от навыков разработчика. Удобство обращения и экономия времени во многом зависит от его способности распознавать оптимальные конфигурации покрытия карт Карно.

Метод Квайна-Мак'Класски

Метод состоит из последовательного выполнения этапов:

1. **Нахождение первичных импликант**;
2. **Расстановка меток**;
3. **Нахождение существенных импликант**;
4. **Вычеркивание лишних столбцов**;
5. **Вычеркивание лишних первичных импликант**;
6. **Выбор минимального покрытия максимальными интервалами**.

Метод Квайна-Мак'Класски

Пусть минимизируемая функция задана в СДНФ.

Элементарная коньюнкция ранга n = ***минитерм*** *ранга n*.

$$y = f\,(0011,\ 0100,\ 0101,\ 0111,\ 1101,\ 1110,\ 1111)$$

1. **Нахождение первичных импликант**;

Для всех минтермов функции определяют вес. Все минитермы функции, вес которых отличается на 1 попарно сравнивают. Записывают новый минитерм ранга n-1, на месте разряда с различными значениями записывают ~. Полученные минитермы ранга n-1 сравнивают попарно с учетом ~ и получают минитермы ранга n-2 и т.д. до тех пор пока это возможно. Минитермы для которых произошло склеивание отмечаются.

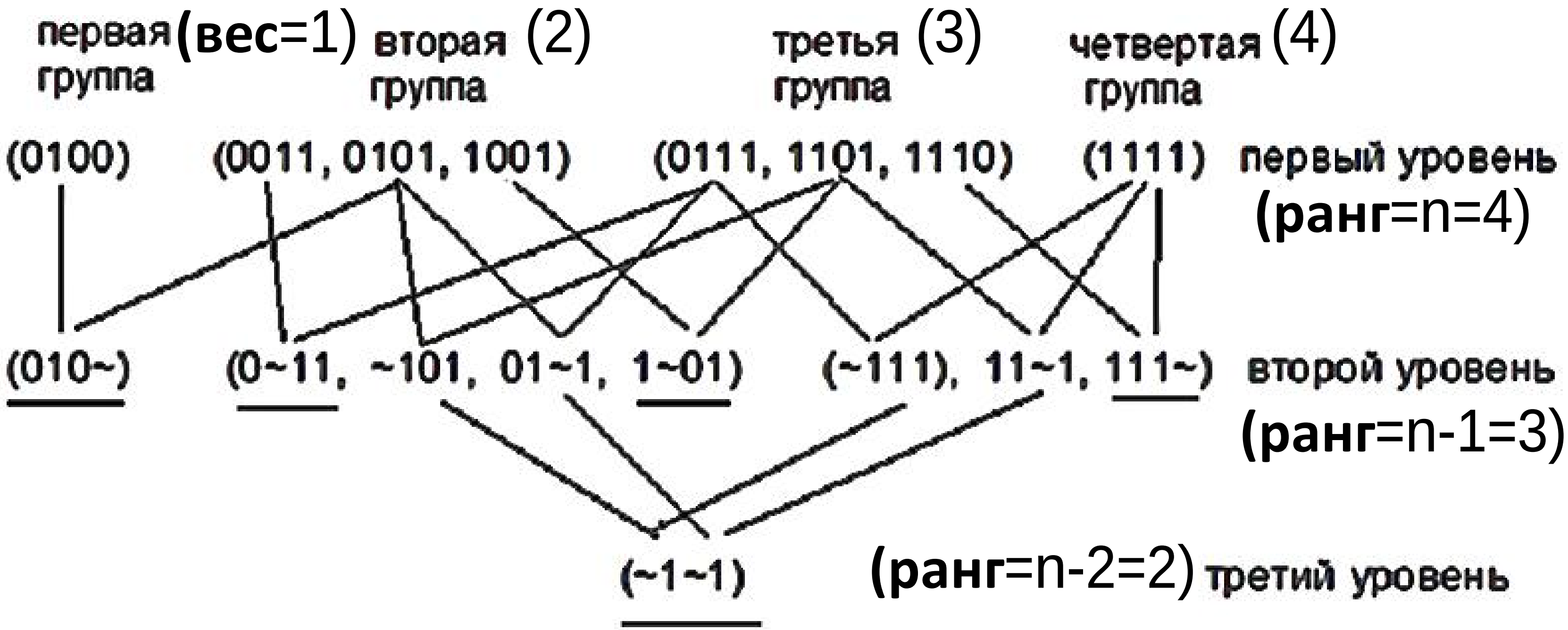
Все неотмеченные минитермы – ***первичные*** *или* ***простые импликанты****.*

Метод Квайна-Мак'Класски

Пусть минимизируемая функция задана в СДНФ.

$$y = V(0011, 0100, 0101, 0111, 1101, 1110, 1111)$$

1. **Нахождение первичных импликант**;



Первичные импликанты: **010~, 0~11, 1~01, 111~, ~1~1**

Метод Квайна-Мак'Класски

Пусть минимизируемая функция задана в СДНФ.

y = V**(0011, 0100, 0101, 0111, 1101, 1110, 1111)**

2. **Расстановка меток**;

Для данной функции $f(x_1, x_2, x_3)$ $= VM_i$,

где M_i *– простые импликанты полученные на первом этапе.*

Для нахождения МДНФ нужно найти минимальное подмножество Mi, покрывающее коньюнкции исходной СДНФ.

Составляется таблица, строки - первичные импликанты минимизируемой функции, столбцы - исходные минитермы.

На пересечении ставится отметка, если первичная импликанта входит в соответствующий минитерм.

Метод Квайна-Мак'Класски

Пусть минимизируемая функция задана в СДНФ.

y = V**(0011, 0100, 0101, 0111, 1101, 1110, 1111)**

2. **Расстановка меток**;

	0011	0100	0101	0111	1001	1101	1110	1111
010~		v	v					
0~11	v			v				
1~01					v	v		
111~							v	v
~1~1			v	v		v		v

Метод Квайна-Мак'Класски

Пусть минимизируемая функция задана в СДНФ.

$y = V(0011, 0100, 0101, 0111, 1101, 1110, 1111)$

3. **Нахождение существенных импликант**;

*Если в столбце только одна метка, то первичная соответствующая импликанта – **существенная**.*

Существенная импликанта не может быть исключена из результата, т.к. без нее не будет полного покрытия исходных минитермов.

Поэтому для сокращение размерности таблицы впоследствии вычеркивают строки с существенными импликантами и вычеркивают столбцы, которые они покрывают.

Существенные импликанты: **0~11, 010~, 1~01, 111~**

Метод Квайна-Мак'Класски

Пусть минимизируемая функция задана в СДНФ.

$y = V(0011, 0100, 0101, 0111, 1101, 1110, 1111)$

3. **Нахождение существенных импликант**;

	0011	0100	0101	0111	1001	1101	1110	1111
010~		v	v					
0~11	v			v				
1~01					v	v		
111~							v	v
~1~1			v	v		v		v

Существенные импликанты: **0~11, 010~, 1~01, 111~**

Метод Квайна-Мак’Класски

Пусть минимизируемая функция задана в СДНФ.

y = V(0011, 0100, 0101, 0111, 1101, 1110, 1111)

4. **Вычеркивание лишних столбцов**;

	0011 v	0100 v	0101 v	0111 v	1001 v	1101 v	1110 v	v 1111
010~		v	v					
0~11	v			v				
1~01					v	v		
111~							v	v
$\cancel{\sim 1 \sim 1}$			v	v		v		v

Метод Квайна-Мак'Класски

Пусть минимизируемая функция задана в СДНФ.

$$y = V(0011, 0100, 0101, 0111, 1101, 1110, 1111)$$

4. **Вычеркивание лишних первичных импликант**;

	0011	0100	0101	0111	1001	1101	1110	1111
010~		v	v					
0~11	v			v				
1~01					v	v		
111~							v	v
~1~1			v	v		v		v

Метод Квайна-Мак'Класски

Пусть минимизируемая функция задана в СДНФ.

$y = \vee(0011, 0100, 0101, 0111, 1101, 1110, 1111)$

5. **Выбор минимального покрытия максимальными интервалами**;

Выбирается такая совокупность первичных импликант, которая включает метки во всех столбцах (как минимум, по одной в каждом столбце).

При нескольких возможных вариантах, предпочтение отдается варианту с минимальным суммарным числом переменных в простых импликантах, образующих покрытие.

!!! В приведенном примере самая короткая коньюнкция (~1~1), покрывающая наибольшее число минитермов, в решение не вошла.

Метод Квайна-Мак'Класски

Пусть минимизируемая функция задана в СДНФ.

$$y = V(0011, 0100, 0101, 0111, 1101, 1110, 1111)$$

5. **Выбор минимального покрытия максимальными интервалами**;

	0011	0100	0101	0111	1001	1101	1110	1111
010~		v	v					
0~11	v			v				
1~01					v	v		
111~							v	v

Тогда МДНФ: $y = \underline{x}_3 x_2 \underline{x}_1 + \underline{x}_3 x_1 x_0 + x_3 \underline{x}_1 x_0 + x_3 x_2 x_1$

Метод Неопределенных коэффициентов

Метод состоит из последовательного выполнения этапов:

1. **Представляем функцию** $f(x_1,x_2,x_3)$ **в виде ДНФ с неопределенными коэффициентами**;

2. **Задаем все возможные значения аргументов** $<x_1, x_2, x_3>$ **и приравниваем к полученному значению функции** 0 **или** 1;

3. **Составляем систему уравнений для коэффициентов и приравниваем**, **соответственно**, **к** 0 **или** 1 **значению функции**;

4. **Все коэффициенты в уравнениях с** 0 **значением функции также равны** 0;

5. **Вычеркиваем из уравнений с** 1 **значением функции все коэффициенты равные** 0, **из уравнений с** 0 **значениями** $f(x_1,x_2,x_3)$

6. **В полученных уравнениях оставляем коэффициенты с минимальным количеством индексов**, **которые присутствуют в максимальном количестве строк**;

7. **Выбираем минимальное покрытие в котором присутствуют коэффициенты с разными индексами**, **отсюда получаем МДНФ**.

Метод Неопределенных коэффициентов

Представление функции в СДНФ с неопределенными коэффициентами:

$$f(x_1,x_2,x_3)=K_1^1x_1\vee K_1^0\,\overline{x_1}\vee K_2^1x_2\vee K_2^0\,\overline{x_2}\vee K_3^1x_3\vee K_3^0\,\overline{x_3}\vee$$
$$\vee K_{12}^{11}x_1x_2\vee K_{12}^{10}x_1\,\overline{x_2}\vee K_{12}^{01}\,\overline{x_1}x_2\vee K_{12}^{00}\,\overline{x_1}\,\overline{x_2}\vee$$
$$\vee K_{13}^{11}x_1x_3\vee K_{13}^{10}x_1\,\overline{x_3}\vee K_{13}^{01}\,\overline{x_1}x_3\vee K_{13}^{00}\,\overline{x_1}\,\overline{x_3}\vee$$
$$\vee K_{23}^{11}x_2x_3\vee K_{23}^{10}x_2\,\overline{x_3}\vee\vee K_{23}^{01}\,\overline{x_2}x_3\vee K_{23}^{00}\,\overline{x_2}\,\overline{x_3}\vee$$
$$\vee K_{123}^{111}x_1x_2x_3\vee K_{123}^{110}x_1x_2\,\overline{x_3}\vee K_{123}^{101}x_1\,\overline{x_2}x_3\vee$$
$$\vee K_{123}^{100}x_1\,\overline{x_2}\,\overline{x_3}\vee K_{123}^{011}\,\overline{x_1}x_2x_3\vee K_{123}^{010}x_1\,\overline{x_2}x_3\vee$$
$$\vee K_{123}^{001}\,\overline{x_1}\,\overline{x_2}x_3\vee K_{123}^{000}\,\overline{x_1}\,\overline{x_2}\,\overline{x_3}$$

Здесь представлены все возможные коньюнкции, **которые могут входить в ДНФ функции** $f(x_1,x_2,x_3)$

Метод Неопределенных коэффициентов

Система уравнений для определения значений коэффициентов на различных наборах $f(x_1,x_2,x_3)$:

$$\left.\begin{aligned}
K_1^1 \vee K_2^1 \vee K_3^1 \vee K_{12}^{11} \vee K_{13}^{11} \vee K_{23}^{11} \vee K_{123}^{111} &= f(1,1,1)\\
K_1^1 \vee K_2^1 \vee K_3^0 \vee K_{12}^{11} \vee K_{13}^{10} \vee K_{23}^{10} \vee K_{123}^{110} &= f(1,1,0)\\
K_1^1 \vee K_2^0 \vee K_3^1 \vee K_{12}^{10} \vee K_{13}^{11} \vee K_{23}^{01} \vee K_{123}^{101} &= f(1,0,1)\\
K_1^1 \vee K_2^0 \vee K_3^0 \vee K_{12}^{10} \vee K_{13}^{10} \vee K_{23}^{00} \vee K_{123}^{100} &= f(1,0,0)\\
K_1^0 \vee K_2^1 \vee K_3^1 \vee K_{12}^{01} \vee K_{13}^{01} \vee K_{23}^{11} \vee K_{123}^{011} &= f(0,1,1)\\
K_1^0 \vee K_2^1 \vee K_3^0 \vee K_{12}^{01} \vee K_{13}^{00} \vee K_{23}^{10} \vee K_{123}^{010} &= f(0,1,0)\\
K_1^0 \vee K_2^0 \vee K_3^1 \vee K_{12}^{00} \vee K_{13}^{01} \vee K_{23}^{01} \vee K_{123}^{001} &= f(0,0,1)\\
K_1^0 \vee K_2^0 \vee K_3^0 \vee K_{12}^{00} \vee K_{13}^{00} \vee K_{23}^{00} \vee K_{123}^{000} &= f(0,0,0)
\end{aligned}\right\}$$

Метод Неопределенных коэффициентов

Пример: $f(x_1,x_2,x_3)=x_1x_2x_3 \vee x_1x_2\overline{x}_3 \vee x_1\overline{x}_2x_3 \vee x_1\overline{x}_2\overline{x}_3 \vee \overline{x}_1\overline{x}_2\overline{x}_3$

Составляем систему:

$$\left.\begin{array}{l}
K_1^1 \vee K_2^1 \vee K_3^1 \vee K_{12}^{11} \vee K_{13}^{11} \vee K_{23}^{11} \vee K_{123}^{111} = 1 \\
K_1^1 \vee K_2^1 \vee K_3^0 \vee K_{12}^{11} \vee K_{13}^{10} \vee K_{23}^{10} \vee K_{123}^{110} = 1 \\
K_1^1 \vee K_2^0 \vee K_3^1 \vee K_{12}^{10} \vee K_{13}^{11} \vee K_{23}^{01} \vee K_{123}^{101} = 1 \\
K_1^1 \vee K_2^0 \vee K_3^0 \vee K_{12}^{10} \vee K_{13}^{10} \vee K_{23}^{00} \vee K_{123}^{100} = 1 \\
K_1^0 \vee K_2^1 \vee K_3^1 \vee K_{12}^{01} \vee K_{13}^{01} \vee K_{23}^{11} \vee K_{123}^{011} = 0 \\
K_1^0 \vee K_2^1 \vee K_3^0 \vee K_{12}^{01} \vee K_{13}^{00} \vee K_{23}^{10} \vee K_{123}^{010} = 0 \\
K_1^0 \vee K_2^0 \vee K_3^1 \vee K_{12}^{00} \vee K_{13}^{01} \vee K_{23}^{01} \vee K_{123}^{001} = 0 \\
K_1^0 \vee K_2^0 \vee K_3^0 \vee K_{12}^{00} \vee K_{13}^{00} \vee K_{23}^{00} \vee K_{123}^{000} = 1
\end{array}\right\}$$

Метод Неопределенных коэффициентов

Из уравнений с 0 значениями $f(x_1,x_2,x_3)$ получаем:

$K_1^0=K_2^0=K_2^1=K_3^0=K_3^1=K_{12}^{00}=K_{12}^{01}=K_{13}^{00}=K_{13}^{01}=K_{23}^{01}=K_{23}^{10}=K_{23}^{11}=K_{123}^{001}=$
$=K_{123}^{010}=K_{123}^{011}=0.$

$$\left.\begin{aligned}
&K_1^1 \vee K_{12}^{11} \vee K_{13}^{11} \vee K_{123}^{111} = 1\\
&K_1^1 \vee K_{12}^{11} \vee K_{13}^{10} \vee K_{123}^{110} = 1\\
&K_1^1 \vee K_{12}^{10} \vee K_{13}^{11} \vee K_{123}^{101} = 1\\
&K_1^1 \vee K_{12}^{10} \vee K_{13}^{10} \vee K_{23}^{00} \vee K_{123}^{100} = 1\\
&K_{23}^{00} \vee K_{123}^{000} = 1
\end{aligned}\right\}
\qquad
\left.\begin{aligned}
&K_1^1 = 1\\
&K_1^1 = 1\\
&K_1^1 = 1\\
&K_1^1 \vee K_{23}^{00} = 1\\
&K_{23}^{00} = 1
\end{aligned}\right\} *$$

* $K_{12}^{11}=K_{12}^{01}=K_{13}^{11}=K_{13}^{10}=K_{123}^{111}=K_{123}^{110}=K_{123}^{101}=K_{123}^{100}=K_{123}^{000}=0$

Отсюда получаем МДНФ: $f(x_1,x_2,x_3)=x_1 \vee \overline{x_2}\ \overline{x_3}$

Достоинства и недостатки МКМК и МНК

***Достоинства**:*

*1. Основным достоинством применения указанных методов это возможность их используются при большом числе переменных $n = 16$ **и более** в профессиональных разработках, они ориентированы на использование в САПР с применением ЭВМ для минимизации полностью и не полностью определенных функций.*

2. Методы КМК и МНК можно использовать для минимизации ФАЛ, заданных как в СДНФ, так и в СКНФ.

4. Удобно минимизировать системы булевых функций, так как легко выделять общие части реализуемой системы ФАЛ.

5. Методы являются алгоритмически систематическим, легко формализуются и легко алгоритмизируются, не зависят от навыков разработчика.

6. Методы позволяют последовательно реализовать все этапы минимизации (склеивание и выявление лишних импликант, получение минимальных покрытий).

***Недостатки**:*

1. Затруднительно использовать методы для числа переменных ≥ 6 для ручной минимизации.

*2. Методы не является алгоритмически инвариантными - время работы метода растёт экспоненциально с увеличением входных данных. Поэтому для функций с очень большим количеством переменных используют **эвристические** алгоритмы.*